

**JNTUHCOLLEGE OF ENGINEERING HYDERABAD (Autonomous)**  
**JNTUH, Kukatpally, Hyderabad – 500 085 Telangana, India**

**ACADEMIC REGULATIONS**  
**COURSE STRUCTURE AND SYLLABUS**

**Computer Science and Engineering**  
with Masters Specialization in  
**Software Engineering**

*For*

**FIVE YEAR INTEGRATED DOUBLE DEGREE MASTERS PROGRAM (IDDMP)**  
**Leading to B.Tech., M.Tech. at JNTUH and**  
**M. Sc. at Blekinge Institute of Technology, Sweden**

*(Applicable for the Batches admitted from 2013-2014)*

**JNTUH**



**BTH, Sweden**



**JNTUHCOLLEGE OF ENGINEERING HYDERABAD (Autonomous)**  
**JNTUH, Kukatpally, Hyderabad – 500 085**  
**Telangana, India**  
**2013**

**JNTUHCOLLEGE OF ENGINEERING HYDERABAD (Autonomous)**  
**JNTUH, Kukatpally, Hyderabad – 500 085 Telangana, India**



**JNTUH**



**BTH, Sweden**

**ACADEMIC REGULATIONS COURSE STRUCTURE AND SYLLABI**

*For*

**FIVE YEAR INTEGRATED DOUBLE DEGREE MASTERS PROGRAM (IDDMP)**  
**Leading to B. Tech., M. Tech. at JNTUH and M. Sc. at Blekinge Institute of Technology,**  
**Sweden**

*(Applicable for the Batches admitted from 2013-2014)*

**In**

**Computer Science and Engineering**

with Masters Specialization in

**Software Engineering**

**1) Five Year Integrated Double Degree Masters Program:**

JNTUH offers Five Years (10 Semesters) Integrated Double Degree Masters Program (IDDMP) under MOU with Blekinge Institute of Technology, (BTH), Sweden in the following Branches of Engineering, at its Constituent Autonomous College - JNTUH College of Engineering, Hyderabad, with effect from the Academic Year 2013 – 14 onwards.

S. No.	UG Program	PG Program	
		<i>M.Sc. at BTH, Sweden (Specialization)*</i>	<i>M. Tech. at JNTUH, India (Specialization)*</i>
1.	B. Tech. in Electronics & Communication Engineering (ECE)	M.Sc. ( Electrical Engineering with emphasis on Telecommunication Systems)	M. Tech. (Telecommunication Systems)
2.	B. Tech. in Electronics & Communication Engineering (ECE)	M.Sc. ( Electrical Engineering with emphasis on Signal Processing)	M. Tech. (Signal Processing)
3.	B. Tech. in Electronics & Communication Engineering (ECE)	M.Sc. ( Electrical Engineering with emphasis on Radio Communications)	M. Tech. (Radio Communications)
4.	B. Tech. in Computer Science & Engineering	M.Sc. (Computer Science)	M. Tech. (Computer Science & Engineering)

5.	B. Tech. in Computer Science & Engineering	M.Sc. (Software Engineering)	M. Tech. (Software Engineering)
6.	B. Tech. in Mechanical Engineering	M.Sc. (Structural Mechanics)	M. Tech. (Structural Mechanics)

*(\* A Minimum of 50% of intake/sanctioned students strength is necessary for any specialization to be offered.)*

A student would be conferred the B. Tech., M. Tech. and M. Sc. Degrees in this IDDMP, after the successful completion of all the requirements for the 10 Semesters of study and earning the appropriate credits.

## 2) Eligibility of Admission:

- 2.1 Admission to the IDDMP shall be made either on the basis of JEE (Main) rank or the merit rank obtained by the qualifying examination at an Entrance Test conducted by the AP State Government (EAMCET), OR the University, OR on the basis of any other order of merit approved by the University.
- 2.2 Students opting for the 5 Year IDDMP must specify their choice for M. Tech. and M. Sc.(with the specialization given above), after choosing the appropriate Branch of Engineering, at the time of Admissions only. Option thus exercised is final, and cannot be changed during the study period.
- 2.3 Students opting for 5 Year IDDMP have to study for the specified period, to earn the relevant credits for the award of the B. Tech, M. Tech & M. Sc. Degrees, and they will not be permitted to have a choice for B. Tech. Degree alone after 4 years study.

## 3) IDDMP Structure:

- 3.1 The Integrated Double Degree Masters Program comprises of two parts – B. Tech. or UG Program and M. Tech. & M. Sc. or PG Program. The UG and PG Programs have the following groups or categories or components, which may include theory subjects / Laboratory courses / Design / Practicals / Major projects etc. as well.

S. No.	UG/PG Program	Group/Category/Component	Description
1)	UG	BS – Basic Sciences	Includes - Mathematics, Physics and Chemistry Subjects
2)	UG	EAS - Engineering Arts and Sciences	Include fundamental engineering subjects
3)	UG	HSS – Humanities and Social Sciences	Includes subjects related to Humanities, Social Sciences and Management
4)	UG	DE – Departmental Electives	Includes Elective subjects related to the parent discipline, department or branch of engineering
5)	UG	DC – Departmental Core	Includes core subjects related to

			the parent discipline, department or branch of engineering
6)	UG	OE – Open Electives	Elective subjects which include inter-disciplinary subjects or subjects in an area outside the parent discipline, department or branch of engineering
7)	UG	Project	B. Tech. Project or UG Project or UG Major Project
8)	PG	PGC	PG Core Subjects related to the M. Tech. & M. Sc. specialization
9)	PG	PGE	PG Elective Subjects related to the M. Tech. & M. Sc. Specialization
10)	PG	Thesis / Dissertation / Project	PG Project / Thesis / Dissertation in M. Tech. & M. Sc. Specialization
11)	PG	Comprehensive Viva	Comprehensive Viva based on UG & PG Subjects

**Note :** The PG subjects / Lab / Electives / Thesis / Dissertation / Comprehensive viva as indicated above are tentative. The actual details of courses / Labs / Project / Majors / minors / Seminars etc. will be as per the norms and procedures of BTH, Sweden corresponding PG specialization.

- 3.2 In the IDDMP, each Subject, Lab., Project, Industrial Training / Seminar / Comprehensive Viva etc. - has specified credits, as indicated in the Course Structure. The credit requirements for IDDMP are: (i) at UG Level: 174 credits at B. Tech. level, plus (ii) at PG Level: 80 credits at M. Tech. & M. Sc.. level.
- 3.3 The minimum instruction days for each Semester shall be 90 working days. In a Semester, one lecture hour per week is rated as one credit, and two tutorial or two practical hours per week may be rated as one credit in general.
- 3.4 There shall be no branch transfers at UG Level, and no changes of specializations at PG Level, after the completion of the First Admission Process.
- 3.5 The Course Structure and Curriculum for the first 3 years (6 Semesters) would be same as that for earning the Regular 4 Year B. Tech. degree in the respective Branch of Engineering. The PG Project Work for the M. Tech. degree shall commence at the beginning of V Year I Semester, and shall be carried out up to the end of the V Year II Semester at BTH, Sweden.

#### 4) Course Work:

- 4.1 A student after securing admission must pursue the 5 Year Integrated Double Degree Master Program of study for a duration of 10 Semesters (or 5 years). Each Semester shall be of 22 weeks duration (inclusive of examinations), with 16 weeks of instruction days at JNTUH up to 7 Semesters and the remaining Semesters will be as per the regulations of BTH.

- 4.2 Course work up to and inclusive of IV Year I Semester shall be conducted at JNTUH, IV year II semester and V Year I & II semesters shall be at Blekinge Institute of Technology (BTH), Sweden.
- 4.3 The student must secure a total of 254 credits for the IDDMP - 174 credits for the B. Tech. degree Program, plus 80 credits for the M. Tech. & M. Sc. Program, under different categories as indicated in Item 3.1 and 3.2.
- 4.4 The student should complete the IDDMP within a period equal to twice the prescribed duration of the Program, from the Date of Admission. Students, who fails to fulfill all the academic requirements for the award of the Double Degrees within 10 academic years from the Date of Admission, shall forfeit their seat in both B. Tech., M. Tech. & M. Sc. Courses.

**5) Attendance Requirements at JNTUH:**

- 5.1 The student shall be eligible to appear for the Semester End Examinations, if he acquires a minimum of 75% attendance in aggregate of all the subjects put together up to IV Year - I Semester in each Semester.
- 5.2 Condonation of shortage of attendance in aggregate up to 10% (net attendance of 65% and above, and below 75%) in each Semester may be granted by the College Academic Committee. Such condonation shall be granted only on genuine and valid reasons, on representation by the candidate with supporting evidence, and on payment of the stipulated condonation fee.
- 5.3 Shortage of attendance below 65% in aggregate shall NOT be condoned.
- 5.4 Students, whose shortage of attendance is not condoned in any Semester, are not eligible to take their End Examinations of that Semester, and their registration for that Semester shall stand cancelled.
- 5.5 A student shall not be promoted to the next Semester, unless he satisfies the attendance requirement of the present Semester. In such cases, the student may seek re-admission for that Semester, as and when offered.

**6) Academic Requirements:**

The following academic requirements have to be satisfied, in addition to the attendance requirements specified in Item 5.

- 6.1(a) **U.G. Part:** A student shall be deemed to have satisfied the minimum academic requirements and earned the credits allotted to each theory subject / practical subject / design / drawing subject / UG Project, if he secures not less than 35% of marks in the end examination (25 out of 70 marks, or 18 out of 50 marks as case may be), and a minimum of 40% marks in the sum total of the internal evaluation and end examination taken together.
- (b) **P.G. Part:** A student shall be deemed to have satisfied the minimum academic requirements and earned the credits allotted to each theory subject / practical subject / design / drawing / Comprehensive Viva-voice, if he secures a minimum of 40% of marks in the end examination, and a minimum of 50% marks in the sum total of the internal evaluation and end examination taken together.

- 6.2 A student shall be **promoted from II to III year** only if he fulfills the academic requirements of earning **40 credits from two regular and one supplementary examinations of I year I Semester, and One Regular & One Supplementary exam of I year II Semester, and one regular examination of II year I Semester** irrespective of whether the candidate takes the examination or not.
- 6.3 A student shall be **promoted from III year to IV year** only if he fulfills the academic requirements of earning total **67 credits from the following examinations**, whether the candidate takes the examinations or not.
- a. Three regular and two supplementary examinations of I B Tech – I Semester.
  - b. Two regular and two Supplementary examinations of I B Tech II Semester
  - c. Two regular and one supplementary examinations of II year – I Semester.
  - d. One regular and one supplementary examinations of II year II Semester.
  - e. One regular examination of III year I Semester
- 6.4 A student shall be eligible to proceed to BTH, Sweden for admission into IV Year II Semester level, if he completes all the subjects and earned all the credits up to III B. Tech, I Semester and UG (Bachelor) Project during IV B. Tech - I Semester at JNTUH.
- 6.5 The credits earned by each student at JNTUH (inclusive of UG & PG) shall be transferred to BTH, Sweden, only after the student successfully completes and earns all credits up to and inclusive of IV year I semester along with UG Project.
- 6.6 A student shall be eligible to appear for the end Semester examination in any Subject (Theory/ Lab.) or Seminar/ Comprehensive Viva/ Project etc., but absent at it or has failed in the end examination, may appear for the same at the supplementary examination or subsequent examination as and when offered.
- 6.7 (a) When a student is detained due to shortage of attendance in any Semester, he may be re-admitted into that Semester when it is next offered, **with the academic regulations of the batch into which he got readmitted.**
- (b) When a student is detained due to lack of credits in any year, he may be readmitted into the next year after fulfillment of the academic requirements, **with the academic regulations of the batch into which he got readmitted**
- 6.8 A student shall register for all the 254 credits as specified in the Course Structure and put up the minimum attendance requirements in all the Semesters, and earn all the 254 (174 at UG level + 80 at PG level) credits for the IDDMP. Marks obtained in all the specified 174 UG credits shall be considered for the calculation of percentage of marks for the B. Tech. Program, and the marks obtained in all the specified 80 PG credits shall be considered for the calculation of % of marks for the M. Tech. & M.Sc. Program. Evaluation of M. Sc. programme performance will be as per BTH norms in vogue.
- 6.9 Students, who fail to earn the 174 UG credits as indicated in the Course Structure, within 8 Academic Years from the Date of Admission, shall forfeit their seat in the

IDDMP, and their admission for the entire Double Degree Masters Program shall stand cancelled.

- 6.10 Students, having secured the 174 UG credits, but fail to earn all the specified PG credits as indicated in the Course Structure, within 10 Academic Years from the Date of Admission, shall forfeit their seat in the IDDMP, and their registration/continuation for the PG Degree Program shall stand cancelled.

7) **Evaluation Procedure:**

The performance of a student shall be evaluated in each Semester Subject wise as follows:

**7.1 For UG Part of the IDDMP**

- i) The performance of a student shall be evaluated in each Semester Subject wise with a maximum of 100 marks for Theory and 75 marks for Practical/Laboratory subject. In addition the project work shall be evaluated for 200 marks.
- ii) For theory subjects, the distribution shall be **30 marks for Internal Evaluation and 70 marks for the End-Examination.**
- iii) For theory subjects, during the Semester there shall be **2 mid-term examinations.** Each mid-term examination consists of **one objective paper for 10 marks, one subjective paper for 15 marks with a duration of 110 minutes (20 minutes for objective and 90 minutes for subjective paper),** and one Assignment for 5 marks. Objective paper shall be set with multiple choice questions, true/false, fill-in the blanks, matching type questions, etc. for 10 marks. Subjective paper shall contain 5 questions, out of which the student has to answer 3 questions, each for 5 marks. The first mid-term examination shall be conducted for the first 50% of the syllabus, and second mid-term examination shall be conducted for the remaining 50% of the syllabus. First Assignment should be submitted before the conduct of the first mid examinations, and the second Assignment should be submitted before the conduct of the second mid examinations. The Assignment shall be as specified by the concerned subject teacher. The total marks secured by the student, in each mid-term examination, are evaluated for 30 marks and the better of the two mid-term examinations shall be taken as the final marks secured by each candidate.
- iv) For practical subjects there shall be a continuous evaluation during the Semester for **25 sessional marks and 50 end examination marks.** Out of the 25 marks for internal, **day-to-day work in the laboratory shall be evaluated for 15 marks,** and **two internal examinations for practical's each of 10 marks,** shall be conducted by the concerned laboratory teachers. The **better of two** internal exams shall be considered. The End Examination shall be conducted by the teacher concerned and another faculty member of the same Department, as suggested by the Head of Dept.
- v) For the Subject having design and / or drawing, (such as Engineering Graphics, Engineering Drawing, Machine Drawing and Estimation etc.), the distribution shall be **30 marks for internal evaluation (15 marks for day-to-day work and 15 marks for internal tests) and 70 marks for end examination.** There shall be two internal tests in a Semester and the **better of the two** shall be considered for the award of marks for internal tests.

- vi) **Open Electives:** Students are to choose One Open Elective (OE-I) during 3<sup>rd</sup> Year I-Semester and another Open Elective (OE-II) during 3<sup>rd</sup> Year II-Semester from the corresponding list of Open Electives given. However, students cannot opt for an Open Elective subjects offered by their own department, if it is already listed under core / elective subjects offered by that department in any Semester.
- vii) The UG Project shall be evaluated for 200 marks, out of which 60 marks shall be for Internal Evaluation and 140 marks for the End Semester Viva-Voce Examination. Out of 60 marks allocated for internal evaluation, 30 marks shall be awarded by the Project Supervisor (based on his continuous performance evaluation of the student), and the other 30 marks shall be awarded by Project Evaluation Committee (PEC) based on the presentation made by the student on the progress of the project at the time of IV Year I Semester I & II-Mid examinations. The PEC shall be constituted by the Head of the Department and shall consist of the Head of the Department, the Supervisor of UG Project and Senior Faculty Member of the Department.

## **7.2 For M. Tech. (PG) Part of the IDDMP at JNTUH**

- i) Theory Subjects are evaluated for 100 marks, and practicals/Laboratory Subjects are also Evaluated for 100 marks.
- ii) For theory subjects, the distribution shall be 40 marks for Internal Evaluation and 60 marks for the End-Examination. For the award of the 40 Internal (sessional) marks for theory subjects, there shall be 2 internal examinations during the Semester, one at the middle of the Semester and the other immediately after the completion of instruction; each of which shall be evaluated for 25 marks, and the better one out of these two internals shall be considered for awarding the 25 sessional marks. Out of the remaining 15 sessional marks, 5 marks are allocated for teacher's assessment (allotment is based on performance of the student in the concerned subject/class) and 10 marks will be awarded based on the student's performance in the Assignments.
- iii) For practical's/Laboratory subjects, there shall be a continuous evaluation during the Semester for 40 sessional marks and 60 End Examination marks. Of these 40 marks, 15 marks shall be awarded for day-to-day work and 25 marks to be awarded by conducting an internal laboratory test. The End Examination shall be conducted by the teacher concerned and another faculty member of the same Department, as suggested by the Head of Dept.
- 7.3** For M. Sc. (PG) part of the IDDMP at BTH, Sweden, the concerned subjects, regulations and evaluation procedure offered shall be as per the norms in vogue at BTH, Sweden.

## **7.4 Grading Procedure (Common for M. Tech. & M. Sc.)**

- i) Marks will be awarded to indicate the performance of each student in each Theory Subject or Practical or UG Project or comprehensive viva voce etc. at JNTUH as specified above in Items 7.1, 7.2, & 7.3 and a proportional letter grade shall be given.
- ii) As a measure of the student's performance, a Grading System using the following letter grades and corresponding percentage of marks shall be followed.

**\*For UG at JNTUH:**

<i>% of Marks Secured at JNTUH</i>	<i>Letter Grade at JNTUH</i>
------------------------------------	------------------------------



70% and above	A
Below 70% but not less than 60%	B
Below 60% but not less than 50%	C
Below 50% but not less than 40%	D
Below 40%	F

**\*For PG at JNTUH:**

<i>% of Marks Secured at JNTUH</i>	<i>Letter Grade at JNTUH</i>
70% and above	A
Below 70% but not less than 60%	B
Below 60% but not less than 50%	C
Below 50%	F

**\*For PG at BTH, Sweden**

<i>% of Marks Secured at JNTUH</i>	<i>Letter Grade Equivalent at BTH, Sweden</i>
Less than or equal to 100% but not less than 90%	A
Below 90% but not less than 80%	B
Below 80% but not less than 70%	C
Below 70% but not less than 60%	D
Below 60% but not less than 50%	E
Below 50%	F

**8) Award of Degree or Class:**

After a student satisfies all the requirements prescribed for the completion of the IDDMP and becomes eligible for the award of the respective Degree, he shall be placed in one of the following four classes ~

**For UG at JNTUH:**

<i>Class Awarded in UG Program</i>	<i>% of Marks Secured at JNTUH</i>	<i>Program Credits at JNTUH</i>
FIRST CLASS with DISTINCTION	70% and above	From the Aggregate secured for the 174 UG credits.
FIRST CLASS	Below 70% but not less than 60%	
SECOND CLASS	Below 60% but not less than 50%	
PASS CLASS	Below 50% but not less than 40%	

**For PG:**

Class Awarded in PG at JNTUH	% of Marks Secured JNTUH	Equivalence between BTH grade and JNTUH marks for the purpose of award of class <b>BTH Grade<sup>#</sup> = JNTUH Marks</b>
FIRST CLASS with DISTINCTION	70% and above	<b>A = 95%</b> <b>B = 85%</b> <b>C = 75%</b> <b>D = 65%</b> <b>E = 55%</b> <b>F &lt; 50%</b>
FIRST CLASS	Below 70% but not less than 60%	
SECOND CLASS	Below 60% but not less than 50%	
FAIL	Below 50%	

**# Note: If any unspecified symbol/ character is given by BTH, Sweden as the ECTS grade for any subject the corresponding ratification for the appropriate specified grade shall be obtained from BTH, Sweden, and corresponding percentage marks will be awarded at JNTUH.**

#### **JNTUH awards the**

- B. Tech. Degree with specialization in **Computer Science and Engineering** after securing for 174 UG credits at JNTUH
- M. Tech. Degree with specialization in **Software Engineering** after securing a total of 80 PG credits (JNTUH) equivalent to 120 ECTS at BTH.

#### **BTH awards the**

- M. Sc. Degree in Electrical Engineering with emphasis on **Software Engineering** specialization after securing a total of 120 ECTS (BTH) equivalent to 80 PG credits at JNTUH.

### **9)Transfer of Credits Policy between JNTUH and BTH, Sweden for awarding PG degrees:**

- JNTUH offers 20 credits for PG in IV Year I Semester for this Integrated Double Degree Masters program, which will be transferred to BTH as 30 credits of European Credit Transfer System (ECTS).
- The 30 ECTS offered in each Semester at BTH for PG in IV year II Semester, V year I and II Semesters are transferred to JNTUH, which will be equivalent to 20 credits at JNTUH.

### **10) Withholding of Results:**

If the student has not paid dues to University/College, or if any case of indiscipline is pending against him, the result of the candidate may be withheld and he will not be allowed to go into the next higher Semester. The award or issue of the Degree may also be withheld in such cases.

### **11) Transitory Regulations:**

Students who have discontinued or have been detained for want of attendance, or who have failed after having undergone the IDDMP, may be considered eligible for readmission to the same or equivalent subjects as and when they are offered, subject to Item 6.9 and 6.10.

**MALPRACTICES RULES**

	<b>Nature of Malpractices</b>	<b>Punishment</b>
	<b>If the candidate:</b>	
1 (a)	Possesses or keeps accessible in examination hall, any paper, note book, programmable calculators, Cell phones, pager, palm computers or any other form of material concerned with or related to the subject of the examination (theory or practical) in which he is appearing but has not made use of (material shall include any marks on the body of the candidate which can be used as an aid in the subject of the examination)	Expulsion from the examination hall and cancellation of the performance in that subject only.
1 (b)	Gives assistance or guidance or receives it from any other candidate orally or by any other body language methods or communicates through cell phones with any candidate or persons in or outside the exam hall in respect of any matter.	Expulsion from the examination hall and cancellation of the performance in that subject only of all the candidates involved. In case of an outsider, he will be handed over to the police and a case is registered against him.
2	Has copied in the examination hall from any paper, book, programmable calculators, palm computers or any other form of material relevant to the subject of the examination (theory or practical) in which the candidate is appearing.	Expulsion from the examination hall and cancellation of the performance in that subject and all other subjects the candidate has already appeared including practical examinations and project work and shall not be permitted to appear for the remaining examinations of the subjects of that Semester/year. The Hall Ticket of the candidate is to be cancelled.
3	Impersonates any other candidate in connection with the examination.	The candidate who has impersonated shall be expelled from examination hall. The candidate is also debarred and forfeits the seat. The performance of the original candidate who has been impersonated, shall be cancelled in all the subjects of the examination (including practicals and project work) already appeared and shall not be allowed to appear for examinations of the remaining subjects of that semester/year. The candidate is also debarred for two consecutive semesters from class work and all examinations. The continuation of the course by the candidate is subject to the academic regulations in connection with forfeiture of seat. If the imposter is an outsider, he will be handed over to the police and a case is registered against him.

4	Smuggles in the Answer book or additional sheet or takes out or arranges to send out the question paper during the examination or answer book or additional sheet, during or after the examination.	Expulsion from the examination hall and cancellation of performance in that subject and all the other subjects the candidate has already appeared including practical examinations and project work and shall not be permitted for the remaining examinations of the subjects of that semester/year. The candidate is also debarred for two consecutive semesters from class work and all examinations. The continuation of the course by the candidate is subject to the academic regulations in connection with forfeiture of seat.
5	Uses objectionable, abusive or offensive language in the answer paper or in letters to the examiners or writes to the examiner requesting him to award pass marks.	Cancellation of the performance in that subject.
6	Refuses to obey the orders of the Chief Superintendent / Assistant – Superintendent / any officer on duty or misbehaves or creates disturbance of any kind in and around the examination hall or organizes a walk out or instigates others to walk out, or threatens the officer-in charge or any person on duty in or outside the examination hall of any injury to his person or to any of his relations whether by words, either spoken or written or by signs or by visible representation, assaults the officer-in-charge, or any person on duty in or outside the examination hall or any of his relations, or indulges in any other act of misconduct or mischief which result in damage to or destruction of property in the examination hall or any part of the College campus or engages in any other act which in the opinion of the officer on duty amounts to use of unfair means or misconduct or has the tendency to disrupt the orderly conduct of the examination.	In case of students of the college, they shall be expelled from examination halls and cancellation of their performance in that subject and all other subjects the candidate(s) has (have) already appeared and shall not be permitted to appear for the remaining examinations of the subjects of that semester/year. The candidates also are debarred and forfeit their seats. In case of outsiders, they will be handed over to the police and a police case is registered against them.
7	Leaves the exam hall taking away answer script or intentionally tears of the script or any part thereof inside or outside the examination hall.	Expulsion from the examination hall and cancellation of performance in that subject and all the other subjects the candidate has already appeared including practical examinations and project work and shall not be permitted for the remaining examinations of the subjects of that semester/year. The candidate is also

		debarred for two consecutive semesters from class work and all examinations. The continuation of the course by the candidate is subject to the academic regulations in connection with forfeiture of seat.
8	Possess any lethal weapon or firearm in the examination hall.	Expulsion from the examination hall and cancellation of the performance in that subject and all other subjects the candidate has already appeared including practical examinations and project work and shall not be permitted for the remaining examinations of the subjects of that semester/year. The candidate is also debarred and forfeits the seat.
9	If student of the college, who is not a candidate for the particular examination or any person not connected with the college indulges in any malpractice or improper conduct mentioned in clause 6 to 8.	Student of the colleges expulsion from the examination hall and cancellation of the performance in that subject and all other subjects the candidate has already appeared including practical examinations and project work and shall not be permitted for the remaining examinations of the subjects of that semester/year. The candidate is also debarred and forfeits the seat. Person(s) who do not belong to the College will be handed over to police and, a police case will be registered against them.
10	Comes in a drunken condition to the examination hall.	Expulsion from the examination hall and cancellation of the performance in that subject and all other subjects the candidate has already appeared including practical examinations and project work and shall not be permitted for the remaining examinations of the subjects of that semester/year.
11	Copying detected on the basis of internal evidence, such as, during valuation or during special scrutiny.	Cancellation of the performance in that subject and all other subjects the candidate has appeared including practical examinations and project work of that semester / year examinations.
12	If any malpractice is detected which is not covered in the above clauses 1 to 11 shall be reported to the College / University for further action to award suitable punishment.	

**12) General:**

- The Academic Regulations should be read as a whole for the purpose of any interpretation.
- The University/College reserves the right of altering the Academic Regulations and/or Syllabus/Course Structure, as and when necessary. The modifications or

amendments may be applicable to all the candidates on rolls, as specified by the University/College.

- Wherever the words 'he' or 'him' or 'his' occur in the above regulations, they will also include 'she' or 'her' or 'hers'.
- Wherever the word 'Subject' occurs in the above regulations, it implies the 'Theory Subject' and 'Practical Subject' or 'Lab.'.
- In case of any ambiguity or doubt in the interpretations of the above regulations, the decision of the CAC / Academic Senate / Vice-Chancellor will be final.
- There shall be no branch transfers, no place transfers, no course transfers, and no transfers from 5 year IDDMP to 4 year B. Tech. (Regular) degree programme (Of same branch or any other branch) or Vice versa, after the completion of the admission process.
- The visa will be issued by the respective embassy, the student is required to fulfill the necessary norms. Students are responsible to show the required financial proofs to the migration authorities while applying for student permit (VISA). Either JNTUH or BTH will not be responsible for the visa rejections caused on the grounds of in-sufficient financial funds/statements or any other issues in front of Migration Board.

\* \* \* \* \*

**ACADEMIC REGULATIONS  
COURSE STRUCTURE AND SYLLABUS**

**Computer Science and Engineering**  
with Masters Specialization in  
**Software Engineering**

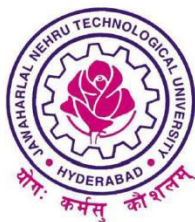
*For*

**FIVE YEAR INTEGRATED DOUBLE DEGREE MASTERS PROGRAM (IDDMP)**

**Leading to B.Tech., M.Tech. at JNTUH and  
M. Sc. at Blekinge Institute of Technology, Sweden**

*(Applicable for the Batches admitted from 2013-2014)*

**JNTUH**



**BTH, Sweden**



**JNTUHCOLLEGE OF ENGINEERING HYDERABAD (Autonomous)**

**JNTUH, Kukatpally, Hyderabad – 500 085**

**Telangana, India**

**2013**

**JNTUH COLLEGE OF ENGINEERING HYDERABAD  
(AUTONOMOUS)**

**COMPUTER SCIENCE & ENGINEERING**

**COURSE STRUCTURE**

(applicable from the batch admitted during 2013-14 and onwards)

**I YEAR**

**I SEMESTER**

S.No.	Group	Subject	L	T	P	Credits
1	BS	Mathematics- I	4	1	0	4
2	EAS	Computer Programming & Data Structures	4	1	0	4
3	HSS	English	4	0	0	4
4	EAS	Engineering Graphics	3	0	3	4
5	EAS	Engineering Mechanics	4	1	0	4
6	EAS	Computer Programming & Data Structures Lab	0	0	3	2
7	HSS	English Language Communication Skills Lab	0	0	3	2
8	EAS	Engineering Workshop	0	0	3	2
		NSS / NCC				
		<b>TOTAL</b>				<b>26</b>

**I YEAR**

**II SEMESTER**

S.No.	Group	Subject	L	T	P	Credits
1	BS	Mathematics – II	4	1	0	4
2	EAS	Basic Electrical & Electronics Engineering	4	1	0	4
3	BS	Engineering Chemistry	4	0	0	4
4	BS	Applied Physics	4	0	0	4
5	EAS	Environmental Science	4	0	0	4
6	BS	Computational Mathematics	2	0	0	2
7	EAS	Basic Electrical & Electronics Engineering Lab	0	0	3	2
8	BS	Applied Physics Lab	0	0	3	2
9	BS	Computational Mathematics Lab	0	0	3	2
		NSS/NCC				
		<b>TOTAL</b>				<b>28</b>



**JNTUH COLLEGE OF ENGINEERING HYDERABAD  
(AUTONOMOUS)**

**COMPUTER SCIENCE & ENGINEERING**

**COURSE STRUCTURE**

**II YEAR**

**I SEMESTER**

S.No.	Group	Subject	L	T	P	Credits
1	DC	Mathematical Foundations of Computer Science	4	1	0	4
2	DC	Digital Logic Design & Micro Processors	4	0	0	4
3	DC	Advanced Data Structures	4	1	0	4
4	DC	Object Oriented Programming Through C++	4	1	0	4
5	DC	Computer Organization & Architecture	4	1	0	4
6	DC	Digital Logic Design & Micro Processor Lab	0	0	3	2
7	DC	Advanced Data Structures Lab	0	0	3	2
8	DC	Object Oriented Programming Lab	0	0	3	2
		<b>TOTAL</b>				<b>26</b>

**II YEAR**

**II SEMESTER**

S.No.	Group	Subject	L	T	P	Credits
1	BS	Computer Oriented Statistical Methods	4	0	0	4
2	DC	Design and Analysis of Algorithms	4	1	0	4
3	DC	Formal Languages and Automata Theory	4	1	0	4
4	DC	Operating Systems	4	1	0	4
5	DC	Software Engineering	4	1	0	4
6	DC	File Structures Lab	0	0	3	2
7	DC	Operating Systems Lab	0	0	3	2
8	DC	Case Tools & Object Oriented Analysis and Design Lab	0	0	3	2
9	HSS	Human Values and Professional Ethics	2	0	0	2
		<b>TOTAL</b>				<b>28</b>

**JNTUH COLLEGE OF ENGINEERING HYDERABAD  
(AUTONOMOUS)**

**COMPUTER SCIENCE & ENGINEERING**

**COURSE STRUCTURE**

**III YEAR**

**I SEMESTER**

S.No.	Group	Subject	L	T	P	Credits
1	DC	Computer Networks	4	1	0	4
2	DC	Compiler Design	4	1	0	4
3	DC	Database Management Systems	4	1	0	4
4	OE-I	Open Elective – I	4	0	0	4
5	HSS	Managerial Economics and Financial Analysis	4	0	0	4
6	DC	Computer Networks and Compiler Design Lab	0	0	3	2
7	DC	Data Base Management Systems Lab	0	0	3	2
8	DC	JAVA Programming Lab	0	0	3	2
		<b>TOTAL</b>				<b>26</b>

**III YEAR**

**II SEMESTER**

S.No.	Group	Subject	L	T	P	Credits
1	DC	Data Warehousing and Data Mining	4	1	0	4
2	DC	Web Technologies	4	1	0	4
3	OE-II	Open Elective – II	4	0	0	4
4	DE-I	Department Elective – I	4	0	0	4
5	DE-II	Department Elective – II	4	0	0	4
6	HSS	Advanced English Language Communications Skills Lab	0	0	3	2
7	DC	Data Warehousing and Data Mining Lab	0	0	3	2
8	DC	Web Technologies Lab	0	0	3	2
9		Disaster Management	2	0	0	2
		<b>TOTAL</b>				<b>28</b>

**JNTUH COLLEGE OF ENGINEERING HYDERABAD.  
(AUTONOMOUS)**

**COMPUTER SCIENCE & ENGINEERING**

**COURSE STRUCTURE**

IV YEAR			I SEMESTER			
S.No.	Group	Subject	L	T	P	Credits
1	DC (UG)	UG Project	-	-	-	4*+8
2	PGC	Software Requirements and Estimation	4	1	0	4+1
3	PGE-I	PG Elective-I	4	1	0	4+1
4	PGE-II	PG Elective-II	4	0	0	4
5	PGC	PG Laboratory	0	0	6	4
6	PG	Comprehensive Viva voce	-	-	-	2
		<b>TOTAL</b>	<b>12</b>	<b>2</b>	<b>6</b>	<b>4*+28</b>

(\* 4 credits for the UG project carried out during the summer after 3<sup>rd</sup> Year –II Semester).

**IV-II AND V-I Semesters at BTH Sweden**

Each Semester in BTH-Sweden, is divided into two Groups ( LP1&LP2 in IV-II semester, LP3 & LP4 in V-I Semester and V-II Semester Project work). Students have to take only 2 subjects per Learning Period (LP) each subject with 7.5 ECTS. Total of 4x7.5=30 ECTS per Semester and 90 ECTS at BTH. The subjects will be notified at the time of entry to the semester.

**COURSES OFFERED AT  
BLEKINGE INSTITUTE OF TECHNOLOGY, SWEDEN**

(FROM IV –II SEMESTER ONWARDS)

IV Year		II Semester	III Study period	
Serial #	Course Slot	Course	Credits	BTH Internal ID
1	Mandatory 1	Research Methodology in Software Engineering and Computer Science	7,5	
2	Mandatory 2	Software Verification and Validation	7,5	

IV Year		II Semester	IV Study period	
Serial #	Course Slot	Course	Credits	BTH Internal ID
3	Mandatory 3	Software Metrics	7,5	
4	Elective 1		7,5	

V Year		I Semester	I Study period	
Serial #	Course Slot	Course	Credits	BTH Internal ID
5	Mandatory 4	Advanced Software Project Management	7,5	
6	Elective 2		7,5	

V Year		I Semester	II Study period	
Serial #	Course Slot	Course	Credits	BTH Internal ID
7	Mandatory 5	Applied Software Project Management	7,5	
8	Elective 3		7,5	

V Year		II Semester	III Study period	
Serial #	Course Slot	Course	Credits	BTH Internal ID
9	Mandatory 6	Master's Thesis (120 credits) in Software Engineering	15 (30)	

V Year		II Semester	IV Study period	
Serial #	Course Slot	Course	Credits	BTH Internal ID
9	Mandatory 6	Master's Thesis (120 credits) in Software Engineering	15 (30)	

**List of elective courses:<sup>1</sup>**

The students are expected to study 3 Elective Courses from the eligible courses provided in the list, corresponding to 22,5 ECTS credits.

- Server Architectures, 7,5
- Global Software Engineering, 7,5
- Software Architectures and Quality, 7,5
- Software Quality Management, 7,5
- Large Scale Requirements Engineering, 7,5
- Software Security, 7,5
- Performance Optimization, 7,5

**Open Elective-I**

S.No.	Subject	Offering Department
1.	GIS & Remote Sensing	Civil Engineering
2.	Non Conventional Power Generation	Electrical & Electronics Engineering
3.	Operations Research	Mechanical Engineering
4.	Electronic Measurements & Instrumentation	Electronics & Communication Engineering
5.	OOPS through JAVA	Computer Science & Engineering
6.	Data Structures and Analysis of Algorithms	Computer Science & Engineering
7.	Operating Systems	Computer Science & Engineering
8.	Material Science	Metallurgical Engineering
9.	Nano Technology	Physics
10	Engineering Management	Humanities & Social Sciences

**Open Elective-II**

S.No.	Subject	Offering Department
1.	Estimation, Quantity survey & Valuation	Civil Engineering
2.	Energy Storage Systems	Electrical & Electronics Engineering
3.	Mechatronics	Mechanical Engineering

<sup>1</sup>Eligible Courses will be offered to student as per the availability of resources.

4.	Principles of Communication Systems	Electronics & Communication Engineering
5.	E-Commerce	Computer Science & Engineering
6.	Computer Graphics	Computer Science & Engineering
7.	Database Management Systems	Computer Science & Engineering
8.	Nano Materials	Metallurgical Engineering
9.	Intellectual Property Rights	Humanities & Social Sciences
10.	Entrepreneurship	Humanities & Social Sciences

**Department Elective-I**

1. Artificial Intelligence
2. Computer Graphics
3. Software Project Management

**Department Elective-II**

1. Pattern Recognition
2. Multimedia Application Development
3. Software Testing Methodologies

**PG Elective – I**

1. Software Quality Assurance and Testing
2. Mobile Computing
3. Big Intelligence & Big Data

**PG Elective – II**

1. Software Architectures and Design Patterns
2. Software Design Engineering

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****I Year B.Tech. CSE I-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>4</b>	<b>1</b>	<b>0</b>	<b>4</b>

**MATHEMATICS-I**  
(Common for all Branches)

**UNIT – I: Differential calculus**

Rolle's Mean value Theorem – Lagrange's Mean Value Theorem – Cauchy's mean value Theorem – (all theorems without proof but with geometrical interpretations), verification of the Theorems and testing the applicability of these theorem to the given function. Radius, Center and circle of curvature – Evaluates and Envelopes, Curve tracing – Cartesian, polar and parametric curves.

Functions of several variables – Functional dependence- Jacobian- Maxima and Minima of functions of two variables with constraints and without constraints-Method of Lagrange multipliers.

**UNIT – II: Improper Integrals, Multiple Integration**

Gamma and Beta Functions –Relation between them, their properties – evaluation of improper integrals using Gamma / Beta functions

Multiple integrals – double and triple integrals – change of order of integration- change of variables (polar, cylindrical and spherical) Finding the area of a region using Double integration and volume of a region in space using triple integration.

**UNIT – III: Vector Calculus**

VectorCalculus: Scalar point function and vector point function, Gradient- Divergence- Curl and their related properties, - Laplacian operator, Line integral – Work done – Surfaceintegrals –Volume integral. Green's Theorem, Stoke's theorem and Gauss's Divergence Theorems (Statement & their Verification). Solenoidal and irrotational vectors, Finding potential function.

**UNIT – IV: First Order Ordinary Differential Equations**

Overview of differential equations- exact, linear and Bernoulli.

Applications of first order differential equations – Newton's Law of cooling, Law of natural growth and decay, orthogonal trajectories.

**UNIT V: Higher Order Ordinary Differential Equations**

Linear differential equations of second and higher order with constant coefficients, Non-homogeneous term of the type  $f(x) = e^{ax}$ ,  $\sin ax$ ,  $\cos ax$ , and  $x^n$ ,  $e^{ax} V(x)$ ,  $x^n V(x)$ , method of variation of parameters. Applications on bending of beams, Electrical circuits and simple harmonic motion.

**TEXT BOOKS:**

1. HIGHER ENGINEERING MATHEMATICS BY B S GREWAL.
2. ADVANCED ENGINEERING MATHEMATICS BY PETER V O'NEIL, CENGAGE LEARNING
3. ENGINEERING MATHEMATICS BY ERWIN KREYSZIG, 10<sup>TH</sup> EDITION WIELY PUBLICATIONS

**REFERENCES:**

1. MATHEMATICS FOR ENGINEERS BY K.B.DATTA AND M.A S.SRINIVAS,CENGAGE PUBLICATIONS
2. MATHEMATICS FOR ENGINEERS BY PROF.A R K PRASAD., WIELY INDIA.
3. ENGINEERING MATHEMATICS -1AND 2 BYT.K.V.IYENGAR & B.KRISHNA GANDHI etel

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****I Year B.Tech. CSE I-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>4</b>	<b>1</b>	<b>0</b>	<b>4</b>

**COMPUTER PROGRAMMING & DATA STRUCTURES****Objectives:**

- To understand the various steps in Program development.
- To understand the basic concepts in C Programming Language.
- To learn how to write modular and readable C Programs
- To understand the basic concepts such as Abstract Data Types, Linear and Non Linear Data structures.
- To understand the notations used to analyze the Performance of algorithms.
- To understand the behavior of data structures such as stacks, queues,
- To understand and analyze various searching and sorting algorithms.
- To write programs in C to solve problems using data structures such as arrays, linked lists,

**UNIT - I**

**Introduction to Computers** – Computer Systems, Computing Environments, Computer Languages, Creating and running programs, Software Development Method, Algorithms, Pseudo code, flow charts, applying the software development method.

**Introduction to C Language** – Background, Simple C programs, Identifiers, Basic data types, Variables, Constants, Input / Output, Operators. Expressions, Precedence and Associativity, Expression Evaluation, Type conversions, Bit wise operators, Statements, Simple C Programming examples.

**Selection Statements** – if and switch statements, Repetition statements – while, for, do-while statements, Loop examples, other statements related to looping – break, continue, go to, Simple C Programming examples.

**UNIT - II**

**Designing Structured Programs**- Functions, basics, user defined functions, inter function communication,

**Standard functions**-Scope, Storage classes-auto, register, static, extern, scope rules, type qualifiers, recursion- recursive functions, Preprocessor commands, example C programs

**Arrays** – Concepts, using arrays in C, inter function communication, array applications, two – dimensional arrays, multidimensional arrays, C program examples.

**UNIT - III**

**Pointers** – Introduction (Basic Concepts), Pointers for inter function communication, pointers to pointers, compatibility, memory allocation functions, array of pointers, programming applications, pointers to void, pointers to functions, command –line arguments.

**Strings** – Concepts, C Strings, String Input / Output functions, arrays of strings, string manipulation functions, string / data conversion, C program examples.

**UNIT - IV**

**Derived types** – Structures – Declaration, definition and initialization of structures, accessing structures, nested structures, arrays of structures, structures and functions, pointers to structures, self referential structures, unions, typedef, bit fields, enumerated types, C programming examples.



**Input and Output** – Concept of a file, streams, standard input / output functions, formatted input / output functions, text files and binary files, file input / output operations, file status functions (error handling), C program examples.

#### **UNIT – V**

**Sorting**- selection sort, bubble sort, insertion sort,

**Searching**-linear and binary search methods.

**Data Structures** – Introduction to Data Structures, abstract data types, Linear list – singly linked list implementation, insertion, deletion and searching operations on linear list, Stacks-Operations, array and linked representations of stacks, stack applications, Queues-operations, array and linked representations.

#### **TEXT BOOKS:**

1. C Programming & Data Structures, B.A.Forouzan and R.F. Gilberg, Third Edition, Cengage Learning.
2. Problem Solving and Program Design in C, J.R. Hanly and E.B. Koffman, Fifth Edition, Pearson Education.
3. The C Programming Language, B.W. Kernighan and Dennis M.Ritchie, PHI/Pearson Education

#### **REFERENCES:**

1. C& Data structures – P. Padmanabham, Third Edition, B.S. Publications.
2. C Programming with problem solving, J.A. Jones & K. Harrow, dreamtech Press
3. Programming in C – Stephen G. Kochan, III Edition, Pearson Eductaion.
4. C for Engineers and Scientists, H.Cheng, Mc.Graw-Hill International Edition
5. Data Structures using C – A.M.Tanenbaum, Y.Langsam, and M.J. Augenstein, Pearson Education / PHI
6. Education / PHI
7. C Programming & Data Structures,E.Balagurusamy,TMH.
8. C Programming & Data Structures, P. Dey, M Ghosh R Thereja, Oxford University Press
9. C& Data structures – E V Prasad and N B Venkateswarlu, S.Chand&Co.

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****I Year B.Tech. CSE I-Sem**

L	T	P	C
4	0	0	4

**ENGLISH****1. INTRODUCTION:**

In view of the growing importance of English as a tool for global communication and the consequent emphasis on training students to acquire communicative competence, the syllabus has been designed to develop linguistic and communicative competencies of Engineering students. The prescribed books and the exercises are meant to serve broadly as students' handbooks.

In the English classes, the focus should be on the skills of reading, writing, listening and speaking and for this the teachers should use the text prescribed for detailed study. For example, the students should be encouraged to read the texts/selected paragraphs silently. The teachers can ask comprehension questions to stimulate discussion and based on the discussions students can be made to write short paragraphs/essays etc.

The text for non-detailed study is for extensive reading/reading for pleasure. Hence, it is suggested that they read it on their own the topics selected for discussion in the class. The time should be utilized for working out the exercises given after each section, as also for supplementing the exercises with authentic materials of a similar kind for example, from newspaper articles, advertisements, promotional material etc.. *However, the stress in this syllabus is on skill development, fostering ideas and practice of language skills.*

**2. OBJECTIVES:**

- ✎ To improve the language proficiency of the students in English with emphasis on LSRW skills.
- ✎ To equip the students to study academic subjects more effectively using the theoretical and practical components of the English syllabus.
- ✎ To develop the study skills and communication skills in formal and informal situations.

**Learning Outcomes**

- 👍 Usage of correct English Language, written and spoken
- 👍 Enrichment of comprehension and fluency
- 👍 Gaining Confidence in using language in varied situations

**SYLLABUS:****Listening Skills:****Objectives**

1. To enable students to develop their listening skill so that they may appreciate its role in the LSRW skills approach to language and improve their pronunciation
2. To equip students with necessary training in listening so that they can comprehend the speech of people of different backgrounds and regions

*Students should be given practice in listening to the sounds of the language to be able to recognise them, to distinguish between them to mark stress and recognise and use the right intonation in sentences.*

- Listening for general content
- Listening to fill up information
- Intensive listening
- Listening for specific information

**Speaking Skills:****Objectives**

1. To make students aware of the role of speaking in English and its contribution to their success.
2. To enable students to express themselves fluently and appropriately in social and professional contexts.
  - Oral practice
  - Describing objects/situations/people
  - Role play – Individual/Group activities (Using exercises from all the **six** units of the prescribed text: ***Skills Annexe: Functional English for Success.***)
  - Just A Minute(JAM) Sessions.

**Reading Skills:****Objectives**

To develop an awareness in the students about the significance of silent reading and comprehension.

1. To develop the ability of students to guess the meanings of words from context and grasp the overall message of the text, draw inferences etc.
  - Skimming the text
  - Understanding the gist of an argument
  - Identifying the topic sentence
  - Inferring lexical and contextual meaning
  - Understanding discourse features
  - Scanning
  - Recognizing coherence/sequencing of sentences

**NOTE :** *The students will be trained in reading skills using the prescribed text for detailed study.*

*They will be examined in reading and answering questions using 'unseen' passages which may be taken from authentic texts, such as magazines/newspaper articles.*

**Writing Skills:****Objectives**

1. To develop an awareness in the students about writing as an exact and formal skill
2. To equip them with the components of different forms of writing, beginning with the lower order ones.

Writing sentences

- Use of appropriate vocabulary
- Paragraph writing
- Coherence and cohesiveness
- Narration / description
- Note Making
- Formal and informal letter writing
- Describing graphs using expressions of comparison

**4. TEXTBOOKS PRESCRIBED:**

In order to improve the proficiency of the student in the acquisition of the four skills mentioned above, the following texts are prescribed:

**For Detailed study**

1. First Text book entitled “*Skills Annexe -Functional English for Success*”, Published by Orient Black Swan, Hyderabad
2. The Second Textbook entitled “*Epitome of Wisdom*”, published by Maruthi Publications, Hyderabad.

The course content and study material is divided into **Five Units**.

**Unit –I:**

1. Chapter entitled ‘*Wit and Humour*’ from ‘Skills Annexe’ -Functional English to Success Published by Orient Black Swan, Hyderabad
2. Chapter entitled ‘*Mokshagundam Visvesvaraya*’ from “*Epitome of Wisdom*”, Published by Maruthi Publications, Hyderabad.

**and**

L-Listening For Sounds, Stress and Intonation

S-Greeting and Taking Leave, Introducing Oneself and Others (Formal and Informal Situations)

R- Reading for Subject/ Theme

W- Writing Paragraphs

G-Types of Nouns and Pronouns

V- Homonyms, homophones synonyms, antonyms

**Unit –II**

1. Chapter entitled “*Advances in Science and Technology*” from “*Skills Annexe - Functional English for Success*” Published by Orient Black Swan, Hyderabad.
2. Chapter entitled ‘*Three days To See*’ from “*Epitome of Wisdom*”, Published by Maruthi Publications, Hyderabad.

**and**

L – Listening for themes and facts

S – Apologizing, interrupting, requesting and making polite conversation

R- for theme and gist

W- Describing people, places, objects, events

G- Verb forms

V- Noun, verb, adjective and adverb

**Unit –III**

1. Chapter entitled ‘*Risk Management*’ from “*Skills Annexe -Functional English for Success*” Published by Orient Black Swan, Hyderabad
2. Chapter entitled ‘*Leela’s Friend*’ by R.K. Narayan from “*Epitome of Wisdom*”, Published by Maruthi Publications, Hyderabad

**And**

L – for main points and sub-points for note taking

S – Giving instructions and directions; Speaking of hypothetical situations

R – Reading for details

W – Note-making, information transfer, punctuation

G – Present tense

V – Synonyms and Antonyms

**Unit –IV**

1. Chapter entitled ‘*Human Values and Professional Ethics*’ from “*Skills Annexe -*

***Functional English for Success***” Published by Orient Black Swan, Hyderabad

2. Chapter entitled ***‘The Last Leaf’*** from ***“Epitome of Wisdom”***, Published by Maruthi Publications, Hyderabad

and

- L -Listening for specific details and information
- S- Narrating, expressing opinions and telephone interactions
- R -Reading for specific details and information
- W- Writing formal letters and CVs
- G- Past and future tenses
- V- Vocabulary - idioms and Phrasal verbs

#### Unit –V

1. Chapter entitled ***‘Sports and Health’*** from ***“Skills Annexe -Functional English for Success”*** Published by Orient Black Swan, Hyderabad
2. Chapter entitled ***‘The Convocation Speech’*** by N.R. Narayanmurthy’ from ***“Epitome of Wisdom”***, Published by Maruthi Publications, Hyderabad

and

- L- Critical Listening and Listening for speaker’s tone/ attitude
- S- Group discussion and Making presentations
- R- Critical reading, reading for reference
- W-Project proposals; Technical reports, Project Reports and Research Papers
- G- Adjectives, prepositions and concord
- V- Collocations and Technical vocabulary and using words appropriately

\* Exercises from the texts not prescribed shall also be used for classroom tasks.

#### SUGGESTED READING:

1. ***Contemporary English Grammar Structures and Composition***by David Green, MacMillan Publishers, New Delhi. 2010.
2. ***Innovate with English: A Course in English for Engineering Students***, edited by T Samson, Foundation Books.
3. English Grammar Practice, **Raj N Bakshi, Orient Longman.**
4. ***Technical Communication*** by Daniel Riordan. 2011. **Cengage Publications. New Delhi.**
5. ***Effective English***, edited by E Suresh Kumar, A RamaKrishna Rao, P Sreehari, Published by **Pearson**
6. Handbook of English Grammar& Usage, **Mark Lester and Larry Beason, Tata Mc Graw –Hill.**
7. Spoken English, **R.K. Bansal & JB Harrison, Orient Longman.**
8. Technical Communication, **Meenakshi Raman, Oxford University Press**
9. Objective English **Edgar Thorpe & Showick Thorpe, Pearson Education**
10. Grammar Games, **Renuvolcuri Mario, Cambridge University Press.**
11. Murphy’s English Grammar with CD, **Murphy, Cambridge University Press.**
12. Everyday Dialogues in English, **Robert J. Dixon, Prentice Hall India Pvt Ltd.,**
13. ABC of Common Errors **Nigel D Turton, Mac Millan Publishers.**
14. Basic Vocabulary **Edgar Thorpe & Showick Thorpe, Pearson Education**
15. Effective Technical Communication, **M Ashraf Rizvi, Tata Mc Graw –Hill.**
16. An Interactive Grammar of Modern English, **Shivendra K. Verma and Hemlatha Nagarajan , Frank Bros & CO**

17. A Communicative Grammar of English, **Geoffrey Leech, Jan Svartvik, Pearson Education**
18. Enrich your English, **Thakur K B P Sinha, Vijay Nicole Imprints Pvt Ltd.,**
19. A Grammar Book for You And I, **C. Edward Good, MacMillan Publishers.**

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****I Year B.Tech. CSE I-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>3</b>	<b>0</b>	<b>3</b>	<b>4</b>

**ENGINEERING GRAPHICS****Pre-requisite: Nil**

**Objective:** The objective of this subject is to provide the basic concepts about Engineering Drawing. Detailed concepts are given in projections, technical drawing, dimensioning and specifications.

**Codes / Tables: Nil****Question Paper Pattern:**

5 Questions to be answered out of 8 questions.

Each question should not have more than 3 bits.

**UNIT – I****INTRODUCTION TO ENGINEERING DRAWING :**

Principles of Engineering Graphics and their Significance, Conic Sections including the Rectangular Hyperbola – General method only. Cycloid, Epicycloid and Hypocycloid, Involute. Scales – Plain, Diagonal and Vernier Scales.

**UNIT- II****ORTHOGRAPHIC PROJECTIONS:**

Principles of Orthographic Projections – Conventions – Projections of Points and Lines  
Projections of Plane regular geometric figures.—Auxiliary Planes.

**UNIT – III**

Projections of Regular Solids – Auxiliary Views.

**UNIT – IV**

Sections or Sectional views of Right Regular Solids – Prism, Cylinder, Pyramid, Cone – Auxiliary views – Sections of Sphere.

Development of Surfaces of Right Regular Solids – Prism, Cylinder, Pyramid and Cone

**UNIT – V****ISOMETRIC PROJECTIONS :**

Principles of Isometric Projection – Isometric Scale – Isometric Views – Conventions – Isometric Views of Lines, Plane Figures, Simple and Compound Solids – Isometric Projection of objects having non- isometric lines. Isometric Projection of Spherical Parts. Conversion of Isometric Views to Orthographic Views and Vice-versa – Conventions

**TEXT BOOKS :**

1. Engineering Drawing N.D. Bhatt / Charotar
2. Engineering Drawing and Graphics Rane and Shah/ Pearson Edu.

**REFERENCE BOOKS:**

1. A Text Book of Engineering Drawing / Dhawan R K / S. Chand
2. Engineering Graphics With Auto CAD / James D Bethune / Pearson Edu.
3. Engineering Graphics / K R Mohan / Dhanpat Rai.
4. Text book on Engineering Drawing / KL Narayana/ P Kannaih / Scitech

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****I Year B.Tech. CSE I-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>4</b>	<b>1</b>	<b>0</b>	<b>4</b>

**ENGINEERING MECHANICS**

1. **Introduction to Mechanics** : Basic Concepts, system of Forces Coplanar Concurrent Forces -Components in Space -Resultant -Moment of Forces and its Application - Couples and Resultant of Force Systems. Equilibrium of system of Forces: Free body diagrams, Equations of Equilibrium of Coplanar Systems and Spatial Systems.
2. **Friction**: Types of friction -Limiting friction -Laws of Friction -static and Dynamic Frictions -Motion of Bodies -Wedge Screw, Screw-jack and differential screw –jack
3. **Centroid and Center of Gravity**: Introduction – Centroids of lines – Centroids of area - Centroids of Composite figures - Theorem of Pappus -Centre of Gravity of Bodies – Centroids of Volumes – Center of gravity of composite bodies.
4. **Area moments of Inertia**: Introduction – Definition of Moment of Inertia -Polar Moment of Inertia – Radius of gyration - Transfer Theorem for moment of inertia – Moments of inertia by integration - Moments of Inertia of Composite Figures, Product of Inertia, Transfer Formula for Product of Inertia.
5. **Mass Moment of Inertia**: Introduction - Moment of Inertia of Masses – Radius of gyration - Transfer Formula for Mass Moments of Inertia – Mass moments of inertia by integration - Mass moment of inertia of composite bodies.

**TEXT BOOKS :**

1. Singer's Engineering Mechanics Statics and Dynamics , K. Vijaya Kumar Reddy, J. Suresh Kumar, BS Publications, 3<sup>rd</sup> Edition(SI Units)Fifth impression 2013
2. Engg. Mechanics / Timoshenko & Young

**REFERENCES :**

1. Engg. Mechanics/ Irving Shames, G. Krishna Mohan Rao, Prentice Hall
2. Engg. Mechanics Umesh Regl / Tayal.
3. A text of Engineering Mechanics /YVD Rao/ K. Govinda Rajulu/ M. Manzoor Hussain, Academic Publishing Company
4. Text Book in Applied Mechanics / Malhotra, Subramanian, Gahlot and Rathore / New Age.
5. Engg. Mechanics / KL Kumar / Tata McGraw Hill.
6. Engg. Mechanics / M.V. Seshagiri Rao & D Rama Durgaiah.
7. Engg. Mechanics / S.S. Bhavikati & K.G. Rajasekharappa



**JNTUH COLLEGE OF ENGINEERING HYDERABAD****I Year B.Tech. CSE I-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>0</b>	<b>0</b>	<b>3</b>	<b>2</b>

**COMPUTER PROGRAMMING & DATA STRUCTURES LAB****Objectives:**

- To understand the various steps in Program development.
- To understand the basic concepts in C Programming Language.
- To learn how to write modular and readable C Programs
- To understand the basic concepts such as Abstract Data Types, Linear and Non Linear Data structures.
- To understand the notations used to analyze the Performance of algorithms.
- To understand the behavior of data structures such as stacks, queues,
- To understand and analyze various searching and sorting algorithms.
- To write programs in C to solve problems using data structures such as arrays, linked lists,

1. Write a C program to find the sum of individual digits of a positive integer.
2. Fibonacci sequence is defined as follows: the first and second terms in the sequence are 0 and 1.  
Subsequent terms are found by adding the preceding two terms in the sequence. Write a C program to generate the first n terms of the sequence.
3. Write a C program to generate all the prime numbers between 1 and n, where n is a value supplied by the user.
4. Write a C program to find the roots of a quadratic equation.
6. Write a C program to find the factorial of a given integer.
7. Write a C program to find the GCD (greatest common divisor) of two given integers.
8. Write a C program to solve Towers of Hanoi problem.
9. Write a C program, which takes two integer operands and one operator from the user, performs the operation and then prints the result. (Consider the operators +, -, \*, /, % and use Switch Statement)
10. Write a C program to find both the largest and smallest number in a list of integers.
11. Write a C program that uses functions to perform the following:
  - i) Addition of Two Matrices
  - ii) Multiplication of Two Matrices
12. Write a C program that uses functions to perform the following operations:
  - i) To insert a sub-string in to a given main string from a given position.
  - ii) To delete n Characters from a given position in a given string.

13. Write a C program to determine if the given string is a palindrome or not
14. Write a C program that displays the position or index in the string S where the string T begins, or – 1 if S doesn't contain T.
15. Write a C program to count the lines, words and characters in a given text.
16. Write a C program to generate Pascal's triangle.
17. Write a C program to construct a pyramid of numbers.
18. Write a C program that uses functions to perform the following operations:
  - i) Reading a complex number
  - ii) Writing a complex number
  - iii) Addition of two complex numbers
  - iv) Multiplication of two complex numbers(Note: represent complex number using a structure.)
19.
  - i) Write a C program which copies one file to another.
  - ii) Write a C program to reverse the first n characters in a file.  
(Note: The file name and n are specified on the command line.)
20.
  - i) Write a C program to display the contents of a file.
  - ii) Write a C program to merge two files into a third file ( i.e., the contents of the first file followed by those of the second are put in the third file)
21. Write a C program that uses functions to perform the following operations on singly linked list.:
  - i) Creation
  - ii) Insertion
  - iii) Deletion
  - iv) Traversal
22. Write C programs that implement stack (its operations) using
  - i) Arrays
  - ii) Pointers
23. Write C programs that implement Queue (its operations) using
  - i) Arrays
  - ii) Pointers
24. Write a C program that implements the following sorting methods to sort a given list of integers in ascending order
  - i) Bubble sort
  - ii) Selection sort
25. Write C programs that use both recursive and non recursive functions to perform the following searching operations for a Key value in a given list of integers:
  - i) Linear search
  - ii) Binary search

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****I Year B.Tech. CSE I-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>0</b>	<b>0</b>	<b>3</b>	<b>2</b>

**ENGLISH LANGUAGE COMMUNICATION SKILLS LAB**

The **Language Lab** focuses on the production and practice of sounds of language and familiarises the students with the use of English in everyday situations and contexts.

**Objectives**

- ✎ To facilitate computer-aided multi-media instruction enabling individualized and independent language learning
- ✎ To sensitise the students to the nuances of English speech sounds, word accent, intonation and rhythm
- ✎ To bring about a consistent accent and intelligibility in their pronunciation of English by providing an opportunity for practice in speaking
- ✎ To improve the fluency in spoken English and neutralize mother tongue influence
- ✎ To train students to use language appropriately for interviews, group discussion and public speaking

**Learning Outcomes**

- 👍 Better Understanding of nuances of language through audio- visual experience and group activities
- 👍 Neutralization of accent for intelligibility
- 👍 Speaking with clarity and confidence thereby enhancing employability skills of the students

**Syllabus:**

**English Language Communication Skills Lab shall have two parts:**

- a. Computer Assisted Language Learning (CALL) Lab**
- b. Interactive Communication Skills (ICS) Lab**

The following course content is prescribed for the **English Language Communication Skills Lab**

**Exercise – I**

**CALL Lab:** Introduction to Phonetics – Speech Sounds – Vowels and Consonants

**ICS Lab:** Ice-Breaking activity and JAM session

Articles, Prepositions, Word Formation- Prefixes & Suffixes, Synonyms & Antonyms

**Exercise – II**

**CALL Lab:** Structure of Syllables - Past Tense Marker and Plural Marker – Weak Forms and Strong Forms - Consonant Clusters.

**ICS Lab:** Situational Dialogues – Role-Play- Expressions in Various Situations – Self-introduction and Introducing Others – Greetings – Apologies – Requests – Social and Professional Etiquette - Telephone Etiquette.

Concord (Subject in agreement with verb) and Words often misspelt- confused/misused

**Exercise - III**

**CALL Lab:** Minimal Pairs- Word Accent and Stress Shifts- Listening Comprehension.

**ICS Lab:** Descriptions- Narrations- Giving Directions and Guidelines.

Sequence of Tenses, Question Tags and One Word Substitutes.

#### Exercise – IV

**CALL Lab:** Intonation and Common errors in Pronunciation.

**ICS Lab:** Extempore- Public Speaking

Active and Passive Voice, –Common Errors in English, Idioms and Phrases

#### Exercise – V

**CALL Lab:** Neutralization of Mother Tongue Influence and Conversation Practice

**ICS Lab:** Information Transfer- Oral Presentation Skills

Reading Comprehension and Job Application with Resume preparation.

#### Minimum Requirement of infrastructural facilities for ELCS Lab:

##### 1. Computer Assisted Language Learning (CALL) Lab:

**The Computer Aided Language Lab** for 40 students with 40 systems, one master console, LAN facility and English language software for self- study by learners.

##### System Requirement (Hardware component):

*Computer network with Lan with minimum 60 multimedia systems with the following specifications:*

- i) P – IV Processor
  - a) Speed – 2.8 GHZ
  - b) RAM – 512 MB Minimum
  - c) Hard Disk – 80 GB
- ii) Headphones of High quality

##### 2. Interactive Communication Skills (ICS) Lab :

**The Interactive Communication Skills Lab:** A Spacious room with movable chairs and audio-visual aids with a Public Address System, a T. V., a digital stereo –audio & video system and camcorder etc.

**Prescribed Lab Manual:** A Manual entitled “*English Language Communication Skills (ELCS) Lab Manual- cum- Work Book*”, published by Cengage Learning India Pvt. Ltd, New Delhi. 2013.

*In addition to the prescribed lab manual, all the listening and speaking activities mentioned in Text-1 and Text-2 can be conducted in the English Language Communication Skills Lab.*

##### Suggested Software:

- ❖ Cambridge Advanced Learners’ English Dictionary with CD.
- ❖ Grammar Made Easy by Darling Kindersley
- ❖ Punctuation Made Easy by Darling Kindersley
- ❖ Clarity Pronunciation Power – Part I
- ❖ Clarity Pronunciation Power – part II
- ❖ Oxford Advanced Learner’s Compass, 8<sup>th</sup> Edition
- ❖ *DELTA’s key to the Next Generation TOEFL Test: Advanced Skill Practice.*
- ❖ Lingua TOEFL CBT Insider, by Dreamtech

- ❖ TOEFL & GRE (KAPLAN, AARCO & BARRONS, USA, Cracking GRE by CLIFFS)
- ❖ **English in Mind (Series 1-4), Herbert Puchta and Jeff Stranks with Meredith Levy, Cambridge**
- ❖ **English Pronunciation in Use** (Elementary, Intermediate, Advanced) Cambridge University Press
- ❖ Raman, M & Sharma, S. 2011. Technical Communication, OUP
- ❖ Sanjay Kumar & Pushp Lata. 2011. Communication Skills, OUP

#### **SUGGESTED READING:**

1. Rama Krishna Rao, A. *et al. English Language Communication Skills – A Reader cum Lab Manual Course Content and Practice*. Chennai: Anuradha Publishers
2. Suresh Kumar, E. & Sreehari, P. 2009. *A Handbook for English Language Laboratories*. New Delhi: Foundation
3. *Speaking English Effectively* 2<sup>nd</sup> Edition by Krishna Mohan and N. P. Singh, 2011. Macmillan Publishers India Ltd. Delhi.
4. Sasi Kumar, V & Dhamija, P.V. *How to Prepare for Group Discussion and Interviews*. Tata McGraw Hill
5. Hancock, M. 2009. *English Pronunciation in Use. Intermediate*. Cambridge: CUP
6. Spoken English: A Manual of Speech and Phonetics by R. K. Bansal & J. B. Harrison. 2013. Orient Blackswan. Hyderabad.
7. Hewings, M. 2009. *English Pronunciation in Use. Advanced*. Cambridge: CUP
8. Marks, J. 2009. *English Pronunciation in Use. Elementary*. Cambridge: CUP
9. Nambiar, K.C. 2011. *Speaking Accurately. A Course in International Communication*. New Delhi : Foundation
10. Soundararaj, Francis. 2012. *Basics of Communication in English*. New Delhi: Macmillan
11. *Spoken English* (CIEFL) in 3 volumes with 6 cassettes, OUP.
12. *English Pronouncing Dictionary* Daniel Jones Current Edition with CD.
13. *A Textbook of English Phonetics for Indian Students* by T.Balasubramanian (Macmillan)

#### **DISTRIBUTION AND WEIGHTAGE OF MARKS**

##### ***English Language Laboratory Practical Examination:***

- 1) The practical examinations for the English Language Laboratory shall be conducted as per the University norms prescribed for the core engineering practical sessions.
- 2) For the Language lab sessions, there shall be a continuous evaluation during the year for 25 sessional marks and 50 year-end Examination marks. Of the 25 marks, 15 marks shall be awarded for day-to-day work and 10 marks to be awarded by conducting Internal Lab Test(s). The year- end Examination shall be conducted by the teacher concerned with the help of another member of the staff of the same department of the same institution.

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****I Year B.Tech. CSE I-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>0</b>	<b>0</b>	<b>3</b>	<b>2</b>

**ENGINEERING WORKSHOP****I. TRADES FOR EXERCISES :**(Any **six** trades from the following with minimum of **two** exercises in each trade)

1. Carpentry
2. Fitting
3. Tin-Smithy
4. Black Smithy
5. House-wiring
6. Foundry
7. Plumbing

**II. Trades for Demonstration & Exposure**

1. Demonstration of power tools & wiring
2. Welding
3. Machine Shop

**III. IT Workshop I:** Computer hardware, identification of parts, Disassembly, Assembly of computer to working condition, simple diagnostic exercises.**IT Workshop II:** Installation of operating system windows and linux simple diagnostic exercises.

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****I Year B.Tech. CSE II-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>4</b>	<b>1</b>	<b>0</b>	<b>4</b>

**MATHEMATICS-II****UNIT – I: Linear ODE with variable coefficients and series solutions (second order only)**

Equations reducible to constant coefficients-Cauchy's and Lagrange's differential equations  
 Motivation for series solutions, Ordinary point and Regular singular point of a differential equation, Transformation of non-zero singular point to zero singular point. Series solutions to differential equations around zero, Frobenius Method about zero.

**UNIT-II: Special Functions**

Legendre's Differential equation, General solution of Legendre's equation, Legendre polynomials Properties: Rodrigue's formula – Recurrence relations, Generating function of Legendre's polynomials – Orthogonality.

Bessel's Differential equation, Bessel functions properties: – Recurrence relations, Orthogonality, Generating function, Trigonometric expansions involving Bessel functions.

**UNIT-III: Partial Differential Equations**

Introduction and Formation of partial differential equation by elimination of arbitrary constants and arbitrary functions, solutions of first order linear (Lagrange) equation and non-linear equations (Charpit's method), Method of separation of variables for second order equations –Applications of Partial differential equations-Two dimensional wave equation., Heat equation.

**UNIT –IV: Laplace Transform**

Definition of Integral transform. Domain of the function and Kernel for the Laplace transforms, Laplace transform of standard functions, first shifting Theorem, Laplace transform of functions when they are multiplied or divided by "t". Laplace transforms of derivatives and integrals of functions. – Unit step function – second shifting theorem – Dirac's delta function, Periodic function – Inverse Laplace transform by Partial fractions(Heaviside method) Inverse Laplace transforms of functions when they are multiplied or divided by "s", Inverse Laplace Transforms of derivatives and integrals of functions, Convolution theorem-solving differential equations by Laplace transforms

**UNIT – V: Fourier Series and Fourier Transforms**

Definition of periodic function. Fourier expansion of periodic functions in a given interval of length,  $2\pi$ , Determination of Fourier coefficients – Fourier series of even and odd functions – Fourier series in an arbitrary interval – even and odd periodic continuation – Half-range Fourier sine and cosine expansions. Fourier integral theorem – Fourier sine and cosine integrals. Fourier transforms – Fourier sine and cosine transforms – properties – inverse transforms – Finite Fourier transforms.

**TEXT BOOKS:**

- 1) HIGHER ENGINEERING MATHEMATICS BY B S GREWAL.
- 2) ADVANCED ENGINEERING MATHEMATICS BY PETER V O'NEIL, CENGAGE LEARNING

- 3) ENGINEERING MATHEMATICS BY ERWIN KREYSZIG, 10<sup>TH</sup> EDITION WILEY PUBLICATIONS

**REFERENCES:**

- 1) MATHEMATICS FOR ENGINEERS SERIES- ADVANCED MATHEMATICS FOR ENGINEERS BY K.B.DATTA AND M.A S.SRINIVAS, CENGAGE PUBLICATIONS
- 2) ADVANCED ENGINEERING MATHEMATICS FOR ENGINEERS BY PROF.A R K PRASAD., WILEY INDIA
- 3) ADVANCED ENGINEERING MATHEMATICS BY SAHANAZ BATHUL, PHI PUBLICATION
- 4) ENGINEERING MATHEMATICS-3 BY T.K.V.IYENGAR &B.KRISHNA GANDHI ETC
- 5) COMPLEX VARIABLES PRINCIPLES AND PROBLEM SESSIONS BY A.K.KAPOOR, WORLD SCIENTIFIC PUBLISHERS
- 6) A TEXT BOOK OF ENGINEERING MATHEMATICS BY N P BALI, MANESH GOYAL



**JNTUH COLLEGE OF ENGINEERING HYDERABAD****I Year B.Tech. CSE II-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>4</b>	<b>1</b>	<b>0</b>	<b>4</b>

**BASIC ELECTRICAL AND ELECTRONICS ENGINEERING****UNIT- I ELECTRICAL and SINGLE PHASE AC CIRCUITS**

**Electrical Circuits** - R-L-C Parameters, Voltage and Current Independent and Dependent Sources, Source Transformation – V–I relationship for Passive elements, Kirchoff's Laws, Network reduction techniques – series, parallel, series parallel, star-to-delta, delta-to-star transformation, Nodal Analysis,

**Single Phase AC Circuits** - R.M.S. and Average values, Form Factor, steady state analysis of series, Parallel and Series parallel Combinations of R, L and C with Sinusoidal excitation, concept of reactance, Impedance, Susceptance and Admittance – phase and phase difference, Concept of Power Factor, j-notation, complex and Polar forms of representation.

**UNIT- II RESONANCE and NETWORK THEOREMS**

**Resonance** – Series resonance and Parallel resonance circuits, concept of bandwidth and Q factor, Locus Diagrams for RL, RC and RLC Combinations for Various Parameters.

**Network Theorems** - Thevenin's, Norton's, Maximum Power Transfer, Superposition, Reciprocity, Tellegen's, Millman's and Compensation theorems for DC and AC excitations.

**UNIT- III P-N JUNCTION DIODE & DIODE CIRCUITS**

**P-N Junction Diode** - Diode equation, Energy Band diagram, Volt-Ampere characteristic, Temperature dependence, Ideal versus practical, Static and dynamic resistances, Equivalent circuit, Load line analysis, Diffusion and Transition Capacitances.

**Rectifiers and Filters** - The P-N junction as a rectifier - A Half Wave Rectifier, Ripple Factor, Full Wave Rectifier, Bridge Rectifier, Harmonic components in Rectifier Circuits, Filters – Inductor Filters, Capacitor Filters, L- section Filters,  $\pi$ - section Filters.

**UNIT- IV BIPOLAR JUNCTION TRANSISTOR**

**Bipolar Junction Transistor (BJT)** - Construction, Principle of Operation, Symbol, Amplifying Action, Common Emitter, Common Base and Common Collector configurations.

**Transistor Biasing And Stabilization** - Operating point, DC & AC load lines, Biasing - Fixed Bias, Emitter Feedback Bias, Collector to Emitter feedback bias, Voltage divider bias, Bias stability, Stabilization against variations in  $V_{BE}$  and  $\beta$ , Bias Compensation using Diodes and Transistors.

**Transistor Configurations** - BJT modeling, Hybrid model, Determination of h-parameters from transistor characteristics, Analysis of CE, CB and CC configurations using h-parameters, Comparison of CE, CB and CC configurations.

**UNIT- V JUNCTION FIELD EFFECT TRANSISTOR & SPECIAL PURPOSE DEVICES:**

**Junction Field Effect Transistor** - Construction, Principle of Operation, Symbol, Pinch-Off Voltage, Volt-Ampere Characteristic, Comparison of BJT and FET, Small Signal Model, Biasing FET.

**Special Purpose Devices** - Breakdown Mechanisms in Semi Conductor Diodes, Zener diode characteristics, Use of Zener diode as simple regulator Principle of operation and Characteristics of Tunnel Diode (With help of Energy band diagram) and Varactor Diode, Principle of Operation of SCR.

**TEXT BOOKS:**

1. Electronic Devices and Circuits – R.L. Boylestad and Louis Nashelsky, PEI/PHI, 9<sup>th</sup> Ed, 2006.
2. Millman's Electronic Devices and Circuits – J.Millman and C.C.Halkias, Satyabratajit, TMH, 2/e, 1998.
3. Engineering circuit analysis- by William Hayt and Jack E. Kemmerly, Mc Graw Hill Company, 6<sup>th</sup> edition.

**REFERENCES:**

1. Introduction to Electronic Devices and Circuits-Rober T. Paynter, Pearson Education.
2. Electronic Devices and Circuits - K. Lal Kishore, B.S. Publications, 2<sup>nd</sup> Edition, 2005.
3. Electronic Devices and Circuits – Anil K. Maini, Varsha Agarwal –Wiley India Pvt. Ltd. 1/e 2009.
4. Linear circuit analysis (time domain phasor and Laplace transform approaches)- 2<sup>nd</sup> edition by Raymond A. DeCarlo and Pen-Min-Lin, Oxford University Press-2004.
5. Network Theory by N.C.Jagan & C.Lakshminarayana, B.S. Publications.
6. Network Theory by Sudhakar, Shyam Mohan Palli, TMH.

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****I Year B.Tech. CSE II-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>4</b>	<b>0</b>	<b>0</b>	<b>4</b>

**ENGINEERING CHEMISTRY**

**Unit-I: Water and its treatment** : Introduction – hardness of water – causes of hardness – types of hardness : temporary and permanent – expression and units of hardness – Estimation of hardness of water by complexometric method. Potable water and its specifications. Steps involved in treatment of potable water - Disinfection of potable water by chlorination and Ozonation. Boiler feed water and its treatment – Calgon conditioning – Phosphate conditioning - Colloidal conditioning – External treatment of water – Lime soda and ion-exchange processes. Desalination of water – Reverse osmosis. Numerical problems – Sewage water - COD, BOD and their determination. Treatment of sewage.

**Unit-II: Electrochemistry and corrosion :**

**Electrochemistry:** Conductance - Specific, equivalent and molar conductance . Ionic mobilities – Relationship between ionic conductance and ionic mobilities. Electro Chemical cells - electrode potential and its determination, standard electrode potential, types of electrodes – Standard hydrogen electrode, calomel and glass electrode. Nernst equation - electrochemical series and its applications.– Concept of concentration cell –Numerical problems.

**Corrosion-** Causes and effects of corrosion – theories of chemical and electrochemical corrosion - mechanism of electrochemical corrosion. Types of corrosion : Galvanic, water-line and pitting corrosion. Factors affecting rate of corrosion. Corrosion control methods – Cathodic protection - sacrificial anode and impressed current cathodic methods. Surface coatings – metallic coatings – methods of application of metallic coatings – Hot dipping , cementation, electroplating of copper - Organic coatings: Paints – their constituents and functions.

**Unit-III: High Polymers:** Definition – Classification of polymers with examples – Types of polymerisation – Chain growth (free radical addition mechanism), step growth polymerization, coordination polymerization. Plastics, fibres and elastomers - definition and characteristics. Plastics – thermoplastic and thermosetting plastics, constituents of plastics . Fibre reinforced plastics. Preparation, properties and applications of PVC, Teflon, Bakelite, Nylon 6:6 and terylene (Dacron); Rubber – Natural rubber , its processing and vulcanization. Elastomers: Preparation, properties and applications of Styrene butadiene, butyl and thiokol rubbers. Conducting polymers – Classification with examples; mechanism of conduction in trans-polyacetylene and applications of conducting polymers. Biodegradable polymers – concept and advantages - Polylactic acid and polyvinyl alcohol and their applications.

**Unit-IV: Chemistry of Energy sources**

**Fuels** :Classification of fuels - characteristics of a good fuel . Solid fuels: Coal – Analysis of coal by proximate and ultimate methods. Liquid fuels- Petroleum and its refining. Characteristics and uses of petrol, diesel and kerosene. Synthetic petrol- Fischer-Tropsch's process. Cracking – thermal cracking and catalytic cracking. Fluid bed catalytic cracking, Knocking - octane and cetane numbers. Gaseous fuels – Composition, properties and uses of Natural gas, LPG and CNG . Flue gas and its analysis by Orsat's apparatus.

**Combustion** – Definition, calorific value, HCV and LCV. Calculation of air quantity required for combustion of a fuel - Numerical problems.

**Alternate Energy sources** : Biodiesel - trans-esterification - advantages of biodiesel, fuel cells ( $H_2$ - $O_2$  and Methanol - $O_2$  fuel cell) – Photovoltaic cells.

**Unit-V : Batteries and Materials :**

**Batteries** : Cell and battery - Primary battery (dry cell, alkaline cell and Lithium cell). Secondary battery ( lead acid, Ni-Cd and lithium ion cell)

**Liquid crystal polymers** : classification, characteristics and applications.

**Insulators**- Characteristics and applications of thermal and electrical insulators.

**Nanomaterials** : Introduction. Preparation of nanomaterials by top down and bottom up approaches. Carbon nano fibres, nano gold particles and fullerenes - Applications of nanomaterials.

**Text Books:**

- 1) Engineering Chemistry by P.C.Jain & M.Jain ; Dhanpat Rai Publishing Company (P) Ltd., New Delhi, (15<sup>th</sup> Edition , 2005).
- 2) Engineering Chemistry by B.Rama Devi & Ch.Venkata Ramana Reddy ; Cengage Learning, 2012.

**Reference Books :**

- 1) A Text Book of Engineering Chemistry by Shashi Chawla, Dhanpat Rai & Co., New Delhi.(3<sup>rd</sup> Edition, 2003).
- 2) Engineering Chemistry by Y. Bharathi Kumari and C. Jyotsna, VGS Booklinks, 2012.
- 3) Text book of Engineering Chemistry by C P Murthy, C V Agarwal and A. Naidu; B.S. Publications, 2006.
- 4) Engineering Chemistry by M. Thirumala Chary and E. Lakshminarayana, Sci tech. Publications Pvt. Ltd., Chennai 2012.
- 5) Engineering Chemistry by B.Sivasankar, Tata McGraw-Hill Publishing Company Ltd., New Delhi 2008.
- 6) A Text Book of Engineering Chemistry by S.S. Dara, S.Chand Publications, (10<sup>th</sup> Edition ,2007).

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****I Year B.Tech. CSE II-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>4</b>	<b>0</b>	<b>0</b>	<b>4</b>

**APPLIED PHYSICS****UNIT-I**

**1.Principles of Quantum & Statistical Mechanics:** Waves and Particles, De Broglie Hypothesis, Matter Waves, Davisson and Germer's Experiment, G.P. Thomson Experiment, Heisenberg's Uncertainty principle, Schrodinger's Time -Independent Wave Equation, Physical Significance of the Wave Function, Particle in One Dimensional Potential Box. Maxwell-Boltzmann, Bose-Einstein and Fermi-Dirac statistics (Qualitative).

**2.Electron theory of Metals:** Introduction, Classical Free Electron Theory of metals, Root Mean Square (RMS )velocity, Mean Free Path, Mean collision Time, Drift Velocity, Relaxation Time, Electrical Resistivity, Draw backs of Classical Free Electron Theory, Density of States, Calculation of Fermi energy, Quantum Free Electron Theory, Electron in a periodic Potential, Kronig-Penny Model (Qualitative Treatment), Origin of Energy Band Formation in Solids, Classification of Materials into Conductors, Semiconductors and insulators, Concept of Effective Mass of an Electron.

**UNIT-II**

**3. Semiconductor Physics:** Position of Fermi Level, Estimation of Carrier concentration in Intrinsic and Extrinsic (p-type & n-type) Semiconductors, Equation of Continuity, Direct and Indirect Band gap Semiconductors, Hall Effect.

**4. Physics of Semiconductor Devices:** Formation of PN Junction, Energy band Diagram and I-V Characteristics of PN Junction Diode, Diode Equation, LED, LCD and Photo Diodes, Solar Cells.

**UNIT-III**

**5. Dielectric Properties:** Basic definitions, Electronic, Ionic (Quantitative) and Orientation Polarizations(Qualitative) and Calculation of Polarizabilities - Internal Fields in Solids, Clausius - Mossotti Equation, Piezo-electricity, Pyro- electricity and Ferro - electricity.

**6. Magnetic Properties:** Basic definitions , Origin of Magnetic Moment, Bohr Magneton, Classification of Dia, Para and Ferro Magnetic Materials on the basis of Magnetic Moment, Domain Theory of Ferro magnetism on the basis of Hysteresis Curve , Soft and Hard Magnetic Materials, Properties of Anti – Ferro and Ferri Magnetic Materials.

**7. Superconductivity:** Introduction to Superconductivity, Properties of Superconductors, Meissner Effect, BCS theory, Type-I and Type –II Superconductors, Magnetic Levitation and Applications of Superconductors.

**UNIT-IV**

**8. Lasers:** Characteristics of Lasers, Spontaneous and stimulated Emission of Radiation, Meta- Stable state, Population Inversion, Lasing Action, Einstein's Coefficients and Relation between them, Ruby Laser, Helium- Neon Laser, Semiconductor Diode Laser and Applications of Lasers.

**9. Fiber Optics:** Principle & construction (structure) of an Optical Fiber, Acceptance Angle, Numerical Aperture, Types of Optical Fibers, Losses in Optical Fibers and Applications of Optical Fibers in communication.

**UNIT-V**

**10. Nanotechnology:** Origin of Nanotechnology, Nano Scale, Surface to Volume Ratio, Quantum Confinement, Bottom-up Fabrication: Sol-Gel, Precipitation, Combustion Methods; Top-Down Fabrication: Chemical Vapor Deposition, Physical Vapor Deposition, Characterization Techniques(XRD, SEM &TEM) and Applications of Nanotechnology.

**Text books:**

1. Principles of Physics by Halliday, Resnick, Walker, Wiley India Pvt Ltd, 9<sup>th</sup> Edition.
2. Introduction to Solid State Physics by Charles Kittel, Wiley India Pvt Ltd, 7<sup>th</sup> Edition
3. Engineering Physics by R.K.GAUR & S.L.GUPTA, Dhanpat Rai Publications.
4. Solid State Physics by A J Dekker, MACMILLAN INDIA LTD.

**References:**

1. Modern Engineering Physics by Dr.K.Vijaya Kumar, Dr. S. Chandralingam, S.CHAND & COMPANY LTD
2. Applied Physics by P.K.Mittal, I K International Publishers
3. Applied Physics by P.K. Palanisamy :Scitech publishers
4. Introduction to Nanotechnology by Charles P.Poole, Jr.Frank J ownes, John Wiley & sons
5. Applied Physics for Engineers by P. Madusudana Rao, Academic Publishing Company
6. Engineering Physics by Sanjay D Jain, Girish G Sahasrbudha: University Press.

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****I Year B.Tech. CSE II-Sem**

L	T	P	C
4	0	0	4

**ENVIRONMENTAL SCIENCE****UNIT - I**

**MULTIDISCIPLINARY NATURE OF ENVIRONMENTAL STUDIES:** Definition, Scope and Importance – Need for Public Awareness.

**NATURAL RESOURCES :** Renewable and non-renewable resources – Natural resources and associated problems – Forest resources – Use and over – exploitation, deforestation, case studies – Timber extraction – Mining, dams and other effects on forest and tribal people – Water resources – Use and over utilization of surface and ground water – Floods, drought, conflicts over water, dams – benefits and problems - Mineral resources: Use and exploitation, environmental effects of extracting and using mineral resources, case studies. - Food resources: World food problems, changes caused by agriculture and overgrazing, effects of modern agriculture, fertilizer-pesticide problems, water logging, salinity, case studies. - Energy resources: Growing energy needs, renewable and non-renewable energy sources use of alternate energy sources. Case studies. Land resources: Land as a resource, land degradation, man induced landslides, soil erosion and desertification. Role of an individual in conservation of natural resources. Equitable use of resources for sustainable lifestyles.

**UNIT - II**

**ECOSYSTEMS :** Concept of an ecosystem. - Structure and function of an ecosystem. - Producers, consumers and decomposers. - Energy flow in the ecosystem - Ecological succession. - Food chains, food webs and ecological pyramids. - Introduction, types, characteristic features, structure and function of the following ecosystem:

- Forest ecosystem
- Grassland ecosystem
- Desert ecosystem
- Aquatic ecosystems (ponds, streams, lakes, rivers, oceans, estuaries)

**UNIT - III**

**BIODIVERSITY AND ITS CONSERVATION :** Introduction - Definition: genetic, species and ecosystem diversity. - Bio-geographical classification of India - Value of biodiversity: consumptive use, productive use, social, ethical, aesthetic and option values - . Biodiversity at global, National and local levels. - . India as a mega-diversity nation - Hot-spots of biodiversity - Threats to biodiversity: habitat loss, poaching of wildlife, man-wildlife conflicts. - Endangered and endemic species of India – Conservation of biodiversity: In-situ and Ex-situ conservation of biodiversity.

**UNIT - IV**

**ENVIRONMENTAL POLLUTION:** Definition, Cause, effects and control measures of :

- Air pollution
- Water pollution
- Soil pollution
- Marine pollution
- Noise pollution
- Thermal pollution
- Nuclear hazards

**SOLID WASTE MANAGEMENT:** Causes, effects and control measures of urban and industrial wastes. - Role of an individual in prevention of pollution. - Pollution case studies. - Disaster management: floods, earthquake, cyclone and landslides.

#### **UNIT - V**

**SOCIAL ISSUES AND THE ENVIRONMENT :** From Unsustainable to Sustainable development -Urban problems related to energy -Water conservation, rain water harvesting, watershed management -Resettlement and rehabilitation of people; its problems and concerns. Case Studies -Environmental ethics: Issues and possible solutions. - Climate change, global warming, acid rain, ozone layer depletion, nuclear accidents and holocaust. Case Studies. -Wasteland reclamation. -Consumerism and waste products. - Environment Protection Act. -Air (Prevention and Control of Pollution) Act. -Water (Prevention and control of Pollution) Act -Wildlife Protection Act -Forest Conservation Act - Issues involved in enforcement of environmental legislation. -Public awareness.

**HUMAN POPULATION AND THE ENVIRONMENT:** Population growth, variation among nations. Population explosion - Family Welfare Programme. -Environment and human health. -Human Rights. -Value Education. -HIV/AIDS. -Women and Child Welfare. -Role of information Technology in Environment and human health. -Case Studies.

**FIELD WORK :** Visit to a local area to document environmental assets River /forest grassland/hill/mountain -Visit to a local polluted site-Urban/Rural/industrial/ Agricultural Study of common plants, insects, birds. -Study of simple ecosystemspond, river, hill slopes, etc.

#### **TEXT BOOK:**

- 1 Textbook of Environmental Studies for Undergraduate Courses by Erach Bharucha for University Grants Commission.,UniversitiesPress
- 2 Environmental Studies by R. Rajagopalan, Oxford University Press.

#### **REFERENCE:**

1. Textbook of Environmental Sciences and Technology by M. Anji Reddy, BS Publication.



**JNTUH COLLEGE OF ENGINEERING HYDERABAD****I Year B.Tech. CSE II-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>2</b>	<b>0</b>	<b>0</b>	<b>2</b>

**COMPUTATIONAL MATHEMATICS****UNIT-I: Matrices and Linear Transformations:**

Real matrices – Symmetric, skew – symmetric, orthogonal. Complex matrices: Hermitian, Skew-Hermitian and Unitary Matrices. Idempotent matrix,

Finding rank of a matrix by reducing to Echelon and Normal forms .

Consistency of system of linear equations (homogeneous and non- homogeneous) using the rank of a matrix – Solving Linear System of equations, (number of equations and unknowns need not be same). Check the uniqueness of solutions.

Cayley-Hamilton Theorem (without Proof) – Verification. Finding inverse of a matrix and powers of a matrix by Cayley-Hamilton theorem, Linear dependence and Independence of Vectors. Linear Transformation – Orthogonal Transformation. Eigen values and Eigen vectors of a matrix. Finding linearly independent Eigen vectors of a matrix when the Eigen values of the matrix are repeated. Properties of Eigen values and Eigen vectors of matrices.

Diagonalization of matrix – Quadratic forms upto three variables- Reduction of quadratic form to canonical form, Rank – Positive definite, negative definite – semi definite – index – signature of quadratic form.

**UNIT – II: Interpolation and Curve fitting :**

**Interpolation:** Introduction- Errors in Polynomial Interpolation – Finite differences- Forward Differences- Backward differences –Central differences – Symbolic relations and separation of symbols- Difference Equations – Differences of a polynomial-Newton's formulae for interpolation – Central difference interpolation Formulae – Gauss Central Difference Formulae –Interpolation with unevenly spaced points-Lagrange's Interpolation formula.

**Curve fitting:** Fitting a straight line –Second degree curve-exponential curve-power curve by method of least squares.

**UNIT – III : Numerical techniques :****Solution of Algebraic and Transcendental Equations and Linear system of equations.**

Introduction – Graphical interpretation of solution of equations .The Bisection Method – The Method of False Position – The Iteration Method – Newton-Raphson Method .

Solving system of non-homogeneous equations by L-U Decomposition method(Crout's Method)Jacobi's and Gauss-Seidel Iteration method

**UNIT- IV: Numerical Differentiation, Integration, and Numerical solutions of First order differential equations:**

Numerical differentiation, Numerical integration – Trapezoidal rule, Simpson's  $1/3^{\text{rd}}$  and  $3/8$  Rule , Generalized Quadrature.

**UNIT – V:**

Numerical solution of Ordinary Differential equations: Solution by Taylor's series method – Picard's Method of successive Approximation- single step methods-Euler's Method-Euler's modified method, Runge-Kutta Methods.

**Text Books:**

- 1) INTRODUCTORY METHODS OF NUMERICAL ANALYSIS BY SS SASTRY
- 2) NUMERICAL AND STATISTICAL METHODS WITH PROGRAMMING IN C BY SUJATHA SINHA AND SUBHABRADA DINDA, SCITEC PUBLISHERS

**References:**

- 1) ADVANCED ENGINEERING MATHEMATICS BY ALAN JEFFERY
- 2) APPLIED NUMERICAL METHODS USING MATLAB BY RAO.V.DUKKIPATI, NEW AGE PUBLISHERS
- 3) NUMERICAL METHODS IN SCIENCE AND ENGINEERING – A PRACTICAL APPROACH BY S. RAJASEKHARAN, S. CHAND PUBLICATIONS

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****I Year B.Tech. CSE II-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>0</b>	<b>0</b>	<b>3</b>	<b>2</b>

**BASIC ELECTRICAL & ELECTRONICS ENGINEERING LAB****PART A: ELECTRONIC WORKSHOP PRACTICE (in 3 lab sessions):**

1. Identification, Specifications, Testing of R, L, C Components (Color Codes), Potentiometers, Switches (SPDT, DPDT, and DIP), Coils, Gang Condensers, Relays, Bread Boards, PCB's
2. Identification, Specifications and Testing of Active Devices, Diodes, BJT's, Low power JFET's, MOSFET's, Power Transistors, LED's, LCD's, SCR, UJT.
3. Study and operation of
  - Multimeters (Analog and Digital)
  - Function Generator
  - Regulated Power Supplies
  - CRO.

**PART B: (For Laboratory examination – Minimum of 12 experiments)**

1. PN Junction diode characteristics A) Forward bias B) Reverse bias.
2. Zener diode characteristics and Zener as voltage Regulator
3. Input & Output characteristics of Transistor in CB / CE configuration
4. Full Wave Rectifier with & without filters
5. Input and Output characteristics of FET in CS configuration
6. Measurement of h-parameters of transistor in CB, CE, CC configurations
7. SCR Characteristics.
8. Verification of KVL and KCL.
9. Serial and Parallel Resonance – Timing, Resonant frequency, Bandwidth and Q-factor determination for RLC network.
10. Verification of Superposition and Reciprocity theorems.
11. Verification of maximum power transfer theorem. Verification on DC, verification on AC with Resistive and Reactive loads.
12. Experimental determination of Thevenin's and Norton's equivalent circuits and verification by direct test.

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****I Year B.Tech. CSE II-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>0</b>	<b>0</b>	<b>3</b>	<b>2</b>

**APPLIED PHYSICS LAB****LIST OF EXPERIMENTS:**

1. Study of characteristics of LED and LASER sources.
2. Magnetic field along the axis of current carrying coil-Stewart and Gee's method.
3. Study of characteristics of p-i-n diode detectors.
4. Determination of frequency of A.C Mains-Sonometer.
5. Torsional pendulum.
6. Energy gap of material of PN- junction.
7. Bending Losses of Fibers & Evaluation of numerical aperture of given fiber.
8. L-C-R circuit.
9. Time constant of an R-C Circuit.
10. Characteristics of solar cell

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****I Year B.Tech. CSE II-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>0</b>	<b>0</b>	<b>3</b>	<b>2</b>

**COMPUTATIONAL MATHEMATICS LAB****Interpolation:****Programming Tasks:**

1. A) Write a C program to determine  $y$  for a given  $x$ , if two arrays of  $x$  and  $y$  of same size are given.(using Newton's interpolation both forward and backward)
- B) Write a C program to determine  $y$  for a given  $x$ , if two arrays of  $x$  and  $y$  of same size are given.(using Lagrange's interpolation)
- C) Write a C program to determine  $y$  for a given  $x$ , if two arrays of  $x$  and  $y$  of same size are given.(using Gauss interpolation)  
(Selection criteria of the interpolation formula are important.)

**Curve fitting:****Programming Tasks:**

2. A) Write a C program to find a line of best fit from the given two arrays of  $x$  and  $y$  of same size.
- B) Write a C program to find a curve of the form  $y = Ae^{Bx}$  from the given two arrays of  $x$  and  $y$  of same size.
- C) Write a C program to find a curve of the form  $y = Ax^B$  from the given two arrays of  $x$  and  $y$  of same size.
- D) Write a C program to find a curve of the form  $y = Ax^2 + Bx + C$  from the given two arrays of  $x$  and  $y$  of same size.

**Solution of Algebraic and Transcendental Equations****Programming Tasks:**

3. A) Write a C program to find the root of a given equation using bisection method.  
(Write this program such that the initial values given to the system are not usable, then the system should ask us to give new set of initial values)
- B) Write a C program to find the root of a given equation using method of false position(regula false position)
- C) Write a C program to find the root of a given equation using iteration method
- D) Write a C program to find the root of a given equation using Newton Rophson method

**Linear system of equations****Programming Tasks:**

4. A) Write a C program to find the solution of given system of linear equations using L- U decomposition method
- B) Write a C program to find the solution of given system of linear equations using jacobi's method
- C) Write a C program to find the solution of given system of equations using Gauss sidel iteration method
- D) Write a C program to find the solution of given system of equations using Gauss Jordan elimination method

**Numerical Differentiation, Integration, and Numerical solutions of First order differential equations:**

**Programming Tasks:**

5. A) Write a C program to evaluate definite integral using trapezoidal rule, Simpson's  $1/3^{\text{rd}}$  rule and  $3/8^{\text{th}}$  rule.  
B) Write a C program to solve a given differential equation using Taylor's series  
C) Write a C program to solve a given differential equation Euler's and modified Eulers method  
D) Write a C program to solve a given differential equation using Ruge-Kutta method.

**Text Books:**

- 1) INTRODUCTORY METHODS OF NUMERICAL ANALYSIS BY SS SASTRY
- 2) NUMERICAL AND STATISTICAL METHODS WITH PROGRAMMING IN C BY SUJATHA SINHA AND SUBHABRADA DINDA, SCITEC PUBLISHERS

**References:**

- 1) ADVANCED ENGINEERING MATHEMATICS BY ALAN JEFFERY
- 2) APPLIED NUMERICAL METHODS USING MATLAB BY RAO.V.DUKKIPATI, NEW AGE PUBLISHERS
- 3) NUMERICAL METHODS IN SCIENCE AND ENGINEERING – A PRACTICAL APPROACH BY S. RAJASEKHARAN, S. CHAND PUBLICATIONS

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****II Year B.Tech. CSE I-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>4</b>	<b>1</b>	<b>0</b>	<b>4</b>

**MATHEMATICAL FOUNDATIONS OF COMPUTER SCIENCE****Objectives:**

- To explain with examples the basic terminology of functions, relations, and sets.
- To perform the operations associated with sets, functions, and relations.
- To relate practical examples to the appropriate set, function, or relation model, and interpret the associated operations and terminology in context.
- To describe the importance and limitations of predicate logic.
- To relate the ideas of mathematical induction to recursion and recursively defined structures.
- To use Graph Theory for solving problems

**Outcomes:**

- Ability to Illustrate by examples the basic terminology of functions, relations, and sets and demonstrate knowledge of their associated operations.
- Ability to Demonstrate in practical applications the use of basic counting principles of permutations, combinations, inclusion / exclusion principle and the pigeonhole methodology.
- Ability to represent and Apply Graph theory in solving computer science problems

**UNIT-I****Fundamentals:**

Sets, Relations and Functions, Methods of Proof, Fundamentals of logic, Logical inferences  
Methods of Proof of an Implication, First Order Logic, Rules of Inference for Quantified Propositions, Mathematical Induction.

**UNIT-II****Elementary Combinatorics:**

Basics of Counting, Combinations and permutations, Enumeration - with repetitions, with constrained repetitions, Principle of Inclusion-Exclusion.

**UNIT-III****Recurrence Relations:**

Generating functions of Sequences, Coefficients of generating functions, Recurrence Relations, Methods of Solving Recurrence Relations, In homogeneous Recurrence Relations.

**UNIT-IV****Relations and Digraphs:**

Relations and Digraphs, Binary Relations, Equivalence Relations, Ordering Relations, Lattices, Paths and Closures, Directed Graphs and Adjacency Matrices.

**UNIT-V****Graphs:**

Graphs, Isomorphism, Trees, Spanning Trees, Binary Trees, Planar Graphs, Euler Circuits, Hamiltonian Graphs, Chromatic Numbers, Four-Colour Problem.

**Text Books:**

1. J. L. Mott, A. Kandel and T. P. Baker : Discrete Mathematics for Computer Scientists, Second Edition, Reston.

**References:**

1. Discrete and Combinational Mathematics- An Applied Introduction-5th Edition – Ralph. P. Grimaldi - Pearson Education.
2. Discrete Mathematical Structures – Thomas Koshy - Tata McGraw Hill Publishing Co.
3. Tremblay JP & Manohar P - Discrete Mathematical Structures with applications to computer science - Tata McGraw Hill Publishing Co.



**JNTUH COLLEGE OF ENGINEERING HYDERABAD****II Year B.Tech. CSE I-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>4</b>	<b>1</b>	<b>0</b>	<b>4</b>

**DIGITAL LOGIC DESIGN & MICRO PROCESSORS****UNIT 1**

**Introduction to number systems:** Binary codes, code conversion, Basic of Boolean algebra, Basic theorems and properties of Boolean algebra Canonical and standard form of Boolean function, all digital logic gates-map method of minimization of 4 and 5 variable functions, Don't-care map entries

**UNIT II**

**NAND and nor implementation:** Design of Binary adders Subtractors, comparator, decoder, encoder, multiplexers and Demultiplexers using gats/CS Introduction to sequential circuits, latches, Flip flops-SR,JK,JK master slave ,D and T type flip flops, Truth tables and excitation tables conversion of flip flops form

**UNIT III**

Concept of shift register, operation of shift register ,its configuration, operation of asynchronous connotes, Design of Synchronous modulo N- connotes, Design and operation of Ring and twisted Ring connotes

**UNIT IV**

**8086 Architecture:** functional diagram, Register organization, memory segmentation, programming model, memory addressing, Physical memory organization ,signal descriptions of 8086,Interrupts of 8086

**UNIT V**

**Instruction set and Assembly language programming of 8086:** instruction formats ,addressing models ,Instruction set ,simple programs involving logical, Branch all instructions ,Solving, String manipulations

**Text Books:**

1. Switching theory and logic design –A.Anand kumar PHI,2013
2. Advanced microprocessor & Pieperar-A.K.Ray and K.M.Bherchandavi ,TMH,2 ND EDITION.

**References:**

1. Switching and Finite Automatic theory-Zvi Kohavi ,Niraj K.Jha Cambridge ,3<sup>rd</sup> edition
2. Digital Design –Morris Mano,PHI, 3<sup>rd</sup> edition
3. Microprocessor and Interfacing –Douglas V.Hall,TMGH 2<sup>nd</sup> edition.

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****II Year B.Tech. CSE I-Sem**

L	T	P	C
4	1	0	4

**ADVANCED DATA STRUCTURES****Objectives:**

- To understand the basic concepts such as Abstract Data Types, Linear and Non Linear Data structures.
- To understand the notations used to analyze the Performance of algorithms.
- To understand the behavior of data structures such as stacks, queues, trees, hash tables, search trees, Graphs and their representations.
- To choose the appropriate data structure for a specified application.
- To understand and analyze various searching and sorting algorithms.
- To write programs in C to solve problems using data structures such as arrays, linked lists, stacks, queues, trees, graphs, hash tables, search trees.

**Unit I:**

Review of basic data structures- The list ADT, Stack ADT, Queue ADT, Implementation using template classes in C++.

**Dictionaries:** linear list representation, skip list representation, operations insertion, deletion and searching.

**Unit II:**

**Hash table representation-** hash functions, collision resolution-separate chaining, open addressing-linear probing, quadratic probing, double hashing, rehashing, extendible hashing, comparison of hashing and skip lists.

**Priority Queues** – Definition, ADT, Realizing a Priority Queue using Heaps, Definition, insertion, Deletion.

**Unit III:**

**Search Trees**-Binary Search Trees, Definition, ADT, Implementation, Operations- Searching, Insertion and Deletion, AVL Trees, Definition, Height of an AVL Tree, Operations – Insertion, Deletion and Searching, Introduction to Red –Black and Splay Trees, B-Tree of order m, height of a B-Tree, insertion, deletion and searching, Comparison of Search Trees.

**UNIT-IV:**

**Graphs-** Graph Implementation Methods. Graph Traversal Methods.

**Sortings**-Quick sort, Heap Sort, External Sorting- Model for external sorting, Merge Sort, Multiway merge, Polyphase merge.

**Unit V:**

**Pattern matching and Tries-** Pattern matching algorithms-Brute force, the Boyer –Moore algorithm, the Knuth-Morris-Pratt algorithm, Standard Tries, Compressed Tries, Suffix tries.

**Text Books:**

1. Data structures, Algorithms and Applications in C++, S.Sahni, University Press (India) Pvt.Ltd, 2nd edition, Universities Press Orient Longman Pvt. Ltd.
2. Data structures and Algorithms in C++, Michael T.Goodrich, R.Tamassia and .Mount, Wiley student edition, John Wiley and Sons.

**References:**

1. Data structures and Algorithm Analysis in C++, Mark Allen Weiss, Pearson Education. Ltd., Second Edition.
2. Data structures and algorithms in C++, 3rd Edition, Adam Drozdek, Thomson
3. Data structures using C and C++, Langsam, Augenstein and Tanenbaum, PHI.
4. Problem solving with C++, The OOP, Fourth edition, W.Savitch, Pearson education.

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****II Year B.Tech. CSE I-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>4</b>	<b>1</b>	<b>0</b>	<b>4</b>

**OBJECT ORIENTED PROGRAMMING THROUGH C++****UNIT I**

**C++ Basics:** Structure of a C++ program, Data types, Declaration of variables, Expressions, Operators, Operator Precedence, Evaluation of expressions, Type conversions, Pointers, Arrays, Pointers and Arrays, Strings, Structures, References.

Flow control statement-if, switch, while, for, do, break, continue, goto statements.

Functions- Scope of variables, Parameter passing, Default arguments, inline functions, Recursive functions, Pointers to functions.

Dynamic memory allocation and deallocation operators-new and delete, Preprocessor directives.

**UNIT-II**

Different paradigms for problem solving, need for OOP, differences between OOP and Procedure oriented programming, Abstraction, Overview of OOP principles, Encapsulation, Inheritance and Polymorphism.

**C++ Classes And Data Abstraction:** Class definition, Class structure, Class objects, Class scope, this pointer, Friends to a class, Static class members, Constant member functions, Constructors and Destructors, Dynamic creation and destruction of objects, Data abstraction, ADT and information hiding.

**UNIT-III**

**Inheritance:** Defining a class hierarchy, Different forms of inheritance, Defining the Base and Derived classes, Access to the base class members, Base and Derived class construction, Destructors, Virtual base class.

**Virtual Functions And Polymorphism:** Static and Dynamic bindings, Base and Derived class virtual functions, Dynamic binding through virtual functions, Virtual function call mechanism, Pure virtual functions, Abstract classes, Implications of polymorphic use of classes, Virtual destructors.

**UNIT-IV**

**C++ I/O:** I/O using C functions, Stream classes hierarchy, Stream I/O, File streams and String streams, Overloading << and >> operators, Error handling during file operations, Formatted I/O.

**UNIT-V**

**Exception Handling:** Benefits of exception handling, Throwing an exception, The try block, Catching an exception, Exception objects, Exception specifications, Stack unwinding, Rethrowing an exception, Catching all exceptions, Design issues in exception handling.

**Text Books:**

1. Problem solving with C++: The Object of Programming, 4<sup>th</sup> Edition, Walter Savitch, Pearson Education.
2. C++, the Complete Reference, 4<sup>th</sup> Edition, Herbert Schildt, TMH.

**References:**

1. C++ Primer, 3<sup>rd</sup> Edition, S.B.Lippman and J.Lajoie, Pearson Education.

2. The C++ Programming Language, 3<sup>rd</sup> Edition, B.Stroutstrup, Pearson Education.
3. OOP in C++, 3<sup>rd</sup> Edition, T.Gaddis, J.Walters and G.Muganda, Wiley DreamTech Press.
4. Object Oriented Programming in C++, 3<sup>rd</sup> Edition, R.Lafore, Galigotia Publications pvt ltd.
5. Computer Science, a Structured Programming Approach Using C++, B.A.Forouzan and R.F.Gilberg, Thomson.

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****II Year B.Tech. CSE I-Sem**

L	T	P	C
4	1	0	4

**COMPUTER ORGANIZATION AND ARCHITECTURE****Objectives:**

- Is to acquaint budding engineers with the basic principles of organization, operation and performance of modern-day computer systems.
- It covers all aspects of computer technology, from the underlying integrated circuit technology used to construct computer components, to the use of parallel organization concepts in combining those components.

**UNIT-I:**

**Basic Structure of Computers:** Computer Types, Functional unit, Basic OPERATIONAL concepts, Bus structures, Software, Performance, multiprocessors and multi computers. Data Representation. Fixed Point Representation, Floating Point Representation.( (IEEE 754)Error Detection codes.

**Register Transfer Language And Micro operations:** Register Transfer language. Register Transfer Bus and memory transfers, Arithmetic Micro operations, logic micro operations, shift micro operations, and Arithmetic logic shift unit.

**UNIT-II****Basic Computer Organization and Design:**

Instruction codes. Computer Registers Computer instructions, Timing and Control, Instruction cycle. Memory Reference Instructions, Input – Output and Interrupt, Complete Computer Description.

**Micro Programmed Control:** Control memory, Address sequencing, micro program example, design of control unit, micro Programmed sequencer, Hard wired control. Vs Micro programmed control

**UNIT-III:**

**Computer Processing Unit Organization:** General Register Organization ,STACK organization, Instruction Formats, Addressing modes, Data Transfer and Manipulation ,Program Control. Reduced Instruction set computer.

**Computer Arithmetic:** Addition and subtraction, multiplication Algorithms, Division Algorithms, Floating – point Arithmetic operations. BCD Adder

**UNIT-IV:**

**Input-Output Organization:** Peripheral Devices, Input-Output Interface, Asynchronous data transfer Modes of Transfer, Priority Interrupt Direct memory Access, Input –Output Processor (IOP)

**Pipeline And Vector Processing:** Parallel Processing, Pipelining, Arithmetic Pipeline, Instruction Pipeline, Dependencies, Vector Processing.

**UNIT-V:**

**Memory Organization:** Memory Hierarchy, Main Memory –RAM And ROM Chips, Memory Address map, Auxiliary memory-magnetic Disks, Magnetic tapes, Associate Memory,-Hardware Organization, Match Logic, Cache Memory –Associative Mapping , Direct Mapping, Set associative mapping ,Writing in to cache and cache Initialization , Cache

Coherence ,Virtual memory-Address Space and memory Space ,Address mapping using pages, Associative memory page table ,page Replacement .

**Multi Processors:** Characteristics or Multiprocessors, Interconnection Structures and Synchronization Cache Coherence. Shared Memory Multiprocessors.

**Text Books:**

1. Computer Systems Architecture – M.Moris Mano, IIIrd Edition, Pearson/PHI
2. Computer Organization – Car Hamacher, Zvonks Vranesic, SafeaZaky, Vth Edition, McGraw Hill.

**Reference:**

1. Computer Organization and Architecture – William Stallings Sixth Edition, Pearson/PHI
2. Structured Computer Organization – Andrew S. Tanenbaum, 4<sup>th</sup> Edition PHI/Pearson
3. Fundamentals or Computer Organization and Design, - Sivaraama Dandamudi Springer Int. Edition.
4. Computer Organization, Anjaneyulu, Himalaya Pub house.

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****II Year B.Tech. CSE I-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>0</b>	<b>0</b>	<b>3</b>	<b>2</b>

**DIGITAL LOGIC DESIGN AND MICROPROCESSOR LAB****Digital Logic Design Lab:**

1. Implement of Logic gates using NAND and NOR gates
2. Design Full adder using gates
3. Design and implement of 4:1 MUX, 8:1 MUX using gates /Ics.
4. Design and Implement of 3 to 8 decoder using gates
5. Design of 4 bit comparator using gates/IC
6. Design of Implement of 4 bit shift register using Flip flops
7. Design and Implement of Decode counter
8. Design and Implement of Asynchronous counter.

**Computer Organization & Micro Processor Lab**

Write assembly language programs for the following using MASAM.

1. Write assembly language programs to evaluate the expressions:
  - i)  $a = b + c - d * e$
  - ii)  $z = x * y + w - v + u / k$
  - a. Considering 8-bit, 16 bit and 32 bit binary numbers as b, c, d, e.
  - b. Considering 2 digit, 4digit and 8 digit BCD numbers.

Take the input in consecutive memory locations and results also Display the results by using "int xx" of 8086. Validate program for the boundary conditions.
2. Write an ALP of 8086 to take N numbers as input. And do the following operations on them.
  - a. Arrange in ascending and Descending order.
3. Find max and minimum
  - a. Find average

Considering 8-bit, 16 bit binary numbers and 2 digit, 4digit and 8 digit BCD numbers. Display the results by using "int xx" of 8086. Validate program for the boundary conditions.
4. Write an ALP of 8086 to take a string of as input (in 'C' format) and do the following Operations on it.
  - a. Find the length
  - b. Find it is Palindrome or n



5. Find whether given string substring or not.
  - a. Reverse a string
  - b. Concatenate by taking another sting  
Display the results by using “int xx” of 8086.
6. Write the ALP to implement the above operations as procedures and call from the main procedure.
7. Write an ALP of 8086 to find the factorial of a given number as a Procedure and call from the main program which display the result.

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****II Year B.Tech. CSE I-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>0</b>	<b>0</b>	<b>3</b>	<b>2</b>

**ADVANCED DATASTRUCTURES LAB****Objectives:**

- To make the student learn a object oriented way of solving problems.
  - To make the student write ADTS for all data structures.
1. C++ programs to implement the following using an array.  
a) Stack ADT b) Queue ADT
  2. Write C++ programs to implement the following using a singly linked list.  
a) Stack ADT b) Queue ADT
  3. Write C++ programs to implement the deque (double ended queue) ADT using a doubly linked list and an array.
  4. Write a C++ program to perform the following operations:  
a) Insert an element into a binary search tree.  
b) Delete an element from a binary search tree.  
c) Search for a key element in a binary search tree.
  5. Write C++ programs that use non-recursive functions to traverse the given binary tree in  
a) Preorder b) inorder and c) postorder.
  6. Write C++ programs for the implementation of bfs and dfs for a given graph.
  7. Write C++ programs for implementing the following sorting methods:  
a) Merge sort b) Heap sort
  8. Write a C++ program to perform the following operations  
a) Insertion into a B-tree b) Deletion from a B-tree
  9. Write a C++ program to perform the following operations  
a) Insertion into an AVL-tree b) Deletion from an AVL-tree
  10. Write a C++ program to implement all the functions of a dictionary (ADT) using hashing.
  11. Write a C++ program for implementing Knuth-Morris- Pratt pattern matching algorithm.
  12. Write a C++ program for implementing Boyer – Moore Patten matching algorithm

**Text Books :**

1. Data Structures and Algorithms in C++, Third Edition, Adam Drozdek, Thomson.
2. Data Structures using C++, D.S. Malik, Thomson
3. “Java Complete Reference” Dietel and Dietel

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****II Year B.Tech. CSE I-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>0</b>	<b>0</b>	<b>3</b>	<b>2</b>

**OBJECT ORIENTED PROGRAMMING LAB**

1. Write a C++ program to find the sum of individual digits of a positive integer.
2. A Fibonacci sequence is defined as follows: the first and second terms in the sequence are 0 and 1. Subsequent terms are found by adding the preceding two terms in the sequence. Write a C++ program to generate the first n terms of the sequence.
3. Write a C++ program to generate all the prime numbers between 1 and n, where n is a value supplied by the user.
4. Write C++ programs that use both recursive and non-recursive functions
  - a) To find the factorial of a given integer.
  - b) To find the GCD of two given integers.
  - c) To find the nth Fibonacci number.
5. Write a C++ program that uses a recursive function for solving Towers of Hanoi Problem.
6. Write a C++ program that uses functions
  - a) To swap two integers.
  - b) To swap two characters.
  - c) To swap two reals. Note: Use overloaded functions.
7. Write a C++ program to find both the largest and smallest number in a list of integers.
8. Write a C++ program to sort a list of numbers in ascending order.
9. Write a C++ program that uses function templates to solve problems-7&8.
10. Write a C++ program to sort a list of names in ascending order.
11. Write a C++ program to implement the matrix ADT using a class. The operations supported by this ADT are:
  - a) Reading a matrix.
  - b) Printing a matrix.
  - c) Addition of matrices.
  - d) Subtraction of matrices.
  - e) Multiplication of matrices.
12. Implement the matrix ADT presented in the problem-11 using overloaded operators (<<, >>, +, -, \*) and templates.
13. Implement the complex number ADT in C++ using a class. The complex ADT is used to represent complex numbers of the form  $c=a+ib$ , where a and b are real numbers. The operations supported by this ADT are:
  - a) Reading a complex number.
  - b) Writing a complex number.
  - c) Addition of Complex numbers.
  - d) Subtraction of complex numbers.
  - e) Multiplication of complex numbers.

- f) Division of complex numbers.
14. Write a C++ program that overloads the + operator and relational operators (suitable) to perform the following operations:
- Concatenation of two strings.
  - Comparison of two strings.
15. Implement the complex number ADT in C++ using a class. The complex ADT is used to represent complex numbers of the form  $c=a+ib$ , where a and b are real numbers. The operations supported by this ADT are:
- Reading a complex number.
  - Writing a complex number.
  - Addition of Complex numbers.
  - Subtraction of complex numbers.
  - Multiplication of complex numbers.
  - Division of complex numbers.
- Note: 1. overload << and >> operators in part a and part b.  
2. Overload +, -, \*, / operators in parts c, d, e and f.
16. Write a template based C++ program that determines if a particular value occurs in an array of Values.
17. Write a C++ program that uses functions to perform the following operations to:
- Insert a sub-string in to the given main string from a given position.
  - Delete n characters from a given position in a given string.
18. Write a C++ program that uses a function to reverse the given character string in place, without any duplication of characters.
19. Write a C++ program to make the frequency count of letters in a given text.
20. Write a C++ program to count the lines, words and characters in a given text.
21. Write a C++ program to determine if the given string is a palindrome or not.
22. Write a C++ program to make frequency count of words in a given text.
23. Write a C++ program that displays the position or index in the string S where the string t begins, or -1 if S doesn't contain t.
24. 2's complement of a number is obtained by scanning it from right to left and complementing all the bits after the first appearance of a 1. Thus 2's complement of 11100 is 00100. Write a C++ program to find the 2's complement of a binary number.
25. Write a C++ program that counts the number of 1 bit in a given integer.
26. Write a C++ program to generate Pascal's triangle.
27. Write a C++ program to construct of pyramid of numbers.
28. Write a C++ program to compute the Sine series.

29. Write a C++ program that converts Roman numeral into an Arabic integer.
30. Write a C++ program which converts a positive Arabic integer into its corresponding Roman numeral.
31. Write a C++ program to display the contents of a text file.
32. Write a C++ program which copies one file to another.
33. Write a C++ program to that counts the characters, lines and words in the text file.
34. Write a C++ program to change a specific character in a file.  
Note: Filename, number of the byte in the file to be changed and the new character are specified on the command line.
35. Write a C++ program to reverse the first n characters in a file.
36. Write a C++ program that uses a function to delete all duplicate characters in the given string.
37. Write a C++ program that uses a function ( i to a ) which converts a number to a character string.
38. Write a C++ program that uses a recursive function to find the binary equivalent of a given non-negative integer n.
39. Write a C++ program to generate prime numbers up to n using Sieve of Eratosthenes method.
40. Write a C++ program
  - a) To write an object to a file.
  - b) To read an object from the file.
41. Write C++ programs that illustrate how the following forms of inheritance are supported:
  - a) Single inheritance
  - b) Multiple inheritance
  - c) Multi level inheritance
  - d) Hierarchical inheritance
42. Write a C++ program that illustrates the order of execution of constructors and destructors when new class is derived from more than one base class.
43. Write a C++ program that illustrates how run time polymorphism is achieved using virtual functions.
44. Write a C++ program that illustrates the role of virtual base class in building class hierarchy.
45. Write a C++ program that illustrates the role of abstract class in building class hierarchy.

**Note:** Use the following in solving the above problems wherever they make sense:

- a) Constructors and destructors.
- b) Overloaded functions.
- c) Overloaded operators.
- d) Function and class templates.

- e) Exception handling mechanism.

**Suggested Books for lab:**

1. **C++ programming from Problem Analysis to Program Design**, 2<sup>nd</sup> Edition, D.S.Malik, Thomson.
2. **Object-Oriented Programming with C++**, M.P.Bhave, S.A.Patekar, Pearson Education.

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****II Year B.Tech. CSE II-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>4</b>	<b>1</b>	<b>0</b>	<b>4</b>

**COMPUTER ORIENTED STATISTICAL METHODS****UNIT-I: Probability**

Sample space and events – Probability – The axioms of probability – Some Elementary theorems – Conditional probability – Baye,s theorem, Random variables – Discrete and continuous.

**UNIT-II: Single Random variables and probability distributions.**

Random variables – Discrete and continuous. Probability distributions, mass function/ density function of a probability distribution . Mathematical Expectation, Moment about origin, Central moments Moment generating function of probability distribution. Binomial , Poisson & normal distributions and their properties . Moment generating functions of the above three distributions. and hence finding the mean and variance.

**UNIT-III: Multiple Random variables, Correlation & Regression**

Joint probability distributions- Joint probability mass / density function, Marginal probability mass / density functions, Covariance of two random variables, Correlation -Coefficient of correlation, The rank correlation.

Regression- Regression Coefficient, The lines of regression and multiple correlation & regression.

**UNIT-IV: Sampling Distributions and Testing of Hypothesis**

**Sampling:** Definitions of population, sampling, statistic, parameter. Types of sampling, Expected values of Sample mean and variance, sampling distribution, Standard error, Sampling distribution of means and sampling distribution of variance.

**Parameter estimations** – likelihood estimate, interval estimations .

**Testing of hypothesis:** Null hypothesis, Alternate hypothesis, type I, & type II errors – critical region, confidence interval, Level of significance. One sided test, Two sided test,

**Large sample tests:**

- (i) Test of Equality of means of two samples equality of sample mean and population mean (cases of known variance & unknown variance, equal and unequal variances)
- (ii) Tests of significance of difference between sample S.D and population S.D.
- (iii) Tests of significance difference between sample proportion and population proportion & difference between two sample proportions.

**Small sample tests:**

Student t-distribution, its properties; Test of significance difference between sample mean and population mean; difference between means of two small samples Snedecor's F- distribution and it's properties. Test of equality of two population variances

Chi-square distribution , it's properties, Chi-square test of goodness of fit

**UNIT- V: Queuing Theory & Stochastic Processes**

Arrival Theorem – Pure Birth process and Death process M/M/1 Model. Introduction to Stochastic Processes – Markov process classification of states – Examples of Markov Chains, Stochastic Matrix, limiting probabilities.

**Text Books:**

- 1) FUNDAMENTALS OF MATHEMATICAL STATISTICS BY S C GUPTA AND V.K.KAPOOR
- 2) PROBABILITY AND STATISTICS FOR ENGINEERS AND SCIENTISTS BY SHELDON M.ROSS,ACADEMIC PRESS
- 3) PROBABILITY AND STATISTICS FOR ENGINEERING AND THE SCIENCEC BY JAY L.DEVORE.

**References:**

- 1) MATHEMATICS FOR ENGINEERS SERIES –PROBABILITY STATISTICS AND STOCHASTIC PROCESS BY K.B.DATTA AND M.A S.SRINIVAS,CENGAGE PUBLICATIONS
- 2) PROBABILITY, STATISTICS AND STOCHASTIC PROCESS BY PROF.A R K PRASAD., WIELY INDIA
- 3) PROBABILITY AND STATISTICS BY T.K.V.IYENGAR &B.KRISHNA GANDHI etel
- 4) A TEXT BOOK OF PROBABILITY AND STATISTICS, SHAHNAZ BATHUL , CENGAGE LEARNING



**JNTUH COLLEGE OF ENGINEERING HYDERABAD****II Year B.Tech. CSE II-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>4</b>	<b>1</b>	<b>0</b>	<b>4</b>

**DESIGN AND ANALYSIS OF ALGORITHMS****Objectives:**

- To analyze performance of algorithms.
  - To choose the appropriate data structure and algorithm design method for a specified application
  - To understand how the choice of data structures and algorithm design methods impacts the performance of programs.
  - To solve problems using algorithm design methods such as the greedy method, divide and conquer, dynamic programming, backtracking and branch and bound.
- Prerequisites (Subjects) Data structures, Mathematical foundations of computer science

**UNIT I :**

**Introduction:** Algorithm, Psuedo code for expressing algorithms, Performance Analysis- Space complexity, Time complexity, Asymptotic Notation- Big oh notation, Omega notation, Theta notation and Little oh notation, Probabilistic analysis, Amortized analysis. Divide and conquer: General method , applications-Binary search, Quick sort, Merge sort, Strassen's matrix multiplication.

**UNIT II :**

**Disjoint Sets:** Disjoint set operations, union and find algorithms, spanning trees. Backtracking: General method, applications-n-queen problem, sum of subsets problem, graph coloring, Hamiltonian cycles. connected components and biconnected components.

**UNIT III:**

**Dynamic Programming:** General method, applications-Matrix chain multiplication, Optimal binary search trees, 0/1 knapsack problem, All pairs shortest path problem, Travelling sales person problem, Reliability design.

**UNIT IV:**

**Greedy method:** General method, applications-Job sequencing with dead lines, 0/1 knapsack problem, Minimum cost spanning trees, Single source shortest path problem.

**UNIT V:**

**Branch and Bound:** General method, applications - Travelling sales person problem, 0/1 knapsack problem- LC Branch and Bound solution, FIFO Branch and Bound solution.

**NP-Hard and NP-Complete problems:** Basic concepts, non deterministic algorithms, NP - Hard and NP-Complete classes, Cook's theorem.

**Text Books :**

1. Fundamentals of Computer Algorithms, Ellis Horowitz, Satraj Sahni and Rajasekharam, Galgotia publications pvt. Ltd.
2. Design and Analysis Algorithms- Parag Himanshu Dave, Himanshu Bhalchandra Dave Publisher: Pearson
3. Algorithm Design: Foundations, Analysis and Internet examples, M.T. Goodrich and R. Tomassia, John Wiley and sons.

**References:**

1. Introduction to Algorithms, second edition, T.H. Cormen, C.E. Leiserson, R.L. Rivest, and C. Stein, PHI Pvt. Ltd./ Pearson Education
2. Introduction to Design and Analysis of Algorithms A strategic approach, R.C.T. Lee, S.S. Tseng, R.C. Chang and T. Tsai, Mc Graw Hill.
3. Data structures and Algorithm Analysis in C++, Allen Weiss, Second edition, Pearson education.
4. Design and Analysis of algorithms, Aho, Ullman and Hopcroft, Pearson education.
5. Algorithms – Richard Johnson baugh and Marcus Schaefer, Pearson Education

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****II Year B.Tech. CSE II-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>4</b>	<b>1</b>	<b>0</b>	<b>4</b>

**FORMAL LANGUAGES AND AUTOMATA THEORY****Objectives:**

The purpose of this course is to acquaint the student with an overview of the theoretical foundations of computer science from the perspective of formal languages.

- Classify machines by their power to recognize languages.
- Employ finite state machines to solve problems in computing.
- Explain deterministic and non-deterministic machines.
- Comprehend the hierarchy of problems arising in the computer sciences.

**UNIT I :**

**Fundamentals :** Strings, Alphabet, Language, Operations, Finite state machine, definitions, finite automaton model, acceptance of strings, and languages, deterministic finite automaton and non deterministic finite automaton, transition diagrams and Language recognizers.

**Finite Automata :** NFA with  $\epsilon$  transitions - Significance, acceptance of languages. Conversions and Equivalence : Equivalence between NFA with and without  $\epsilon$  transitions, NFA to DFA conversion, minimisation of FSM, equivalence between two FSM's, Finite Automata with output- Moore and Melay machines.

**UNIT II:**

**Regular Languages :** Regular sets, regular expressions, identity rules, Constructing finite Automata for a given regular expressions, Conversion of Finite Automata to Regular expressions. Pumping lemma of regular sets, closure properties of regular sets (proofs not required).

**Grammar Formalism :** Regular grammars-right linear and left linear grammars, equivalence between regular linear grammar and FA, inter conversion, Context free grammar, derivation trees, sentential forms. Right most and leftmost derivation of strings.

**UNIT III:**

**Context Free Grammars:** Ambiguity in context free grammars. Minimization of Context Free Grammars. Chomsky normal form, Greibach normal form, Pumping Lemma for Context Free Languages. Enumeration of properties of CFL (proofs omitted).

**Push Down Automata :** Push down automata, definition, model, acceptance of CFL, Acceptance by final state and acceptance by empty state and its equivalence. Equivalence of CFL and PDA, interconversion. (Proofs not required). Introduction to DCFL and DPDA.

**UNIT IV :**

**Turing Machine :** Turing Machine, definition, model, design of TM, CSL, LBA Computable functions, recursively enumerable languages. Church's hypothesis, counter machine, types of Turing machines (proofs not required).

**UNIT V:**

**Computability Theory :** Chomsky hierarchy of languages, linear bounded automata and context sensitive language, LR(0) grammar, decidability of, problems, Universal Turing Machine, undecidability of posts. Correspondence problem, Turing reducibility, Definition of P and NP problems, NP complete and NP hard problems.

**Text Books :**

1. "Introduction to Automata Theory Languages and Computation". Hopcroft H.E. and Ullman J. D. Pearson Education
2. Introduction to Theory of Computation –Sipser 2nd edition Thomson
3. Theory of Computer Science – Automata languages and computation -Mishra and Chandrashekar, 2nd edition, PHI

**References :**

1. Introduction to Computer Theory, Daniel I.A. Cohen, John Wiley.
2. Introduction to languages and the Theory of Computation ,John C Martin, TMH
3. "Elements of Theory of Computation", Lewis H.P. & Papadimitriou C.H. Pearson /PHI.

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****II Year B.Tech. CSE II-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>4</b>	<b>1</b>	<b>0</b>	<b>4</b>

**OPERATING SYSTEMS****Objectives:**

- Provide an introduction to operating system concepts (i.e., processes, threads, scheduling, synchronization, deadlocks, memory management, file and I/O subsystems and protection) as fundamental principles by reference to real systems
- To give exposure to the professional responsibilities that are part of operating system design and development.
- To provide the student with the ability to write system oriented programs on Unix/Linux.

**UNIT I:**

**Operating System Introduction**, Structures - Simple Batch, Multi programmed, Time-shared, Personal Computer, Parallel, Distributed Systems, Real-Time Systems, System components, Operating-System services, System Calls, Virtual Machines, System Design and Implementation.

**UNIX/LINUX Utilities** - Introduction to Unix file system, vi editor, file handling utilities, security by file permissions, process utilities, disk utilities, networking commands, text processing utilities and backup utilities,

**Working with Bash shell:** what is a shell, shell responsibilities, pipes and input Redirection, output redirection, here documents, the shell as a programming language, shell meta characters, shell variables, shell commands, the environment, control structures, shell script examples.

**UNIT II:**

**Process and CPU Scheduling** - Process concepts and scheduling, Operation on processes, Cooperating Processes, Threads, and Interposes Communication Scheduling Criteria, Scheduling Algorithm, Multiple -Processor Scheduling, Real-Time Scheduling.

**System call interface for process management**-fork, exit, wait, waitpid, exec,

**Deadlocks** - System Model, Dead locks Characterization, Methods for Handling Dead locks Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, and Recovery from Deadlock.

**UNIT – III:**

**Process Management and Synchronization** - The Critical Section Problem, Synchronization Hardware, Semaphores, and Classical Problems of Synchronization, Critical Regions, Monitors.

**Interprocess Communication Mechanisms:** IPC between processes on a single computer system, IPC between processes on different systems, using pipes, fifos, message queues, shared memory, semaphores.

**UNIT IV**

**Memory Management and Virtual Memory** - Logical versus Physical Address Space, Swapping, Contiguous Allocation, Paging, Segmentation, Segmentation with Paging. Demand Paging, Performance of Demanding Paging, Page Replacement, Page Replacement Algorithm, Allocation of Frames, Thrashing.

**UNIT V:**

**File System Interface and Implementation** -Access methods, Directory Structure, Protection, File System Structure, Allocation methods, Free-space Management, Directory Management, Directory Implementation, Efficiency and Performance.

**Unix/LINUX Files:** File structure, directories, files and devices, System calls, library functions, low level file access, usage of open, creat, read, write, close, lseek, stat, ioctl.

**TEXT BOOKS:**

1. Operating System Principles- Abraham Silberchatz, Peter B. Galvin, Greg Gagne 7<sup>th</sup> Edition, John Wiley
2. Operating Systems – Internals and Design Principles Stallings, Fifth Edition–2005, Pearson Education/PHI
3. Unix the ultimate guide, Sumitabha Das, TMH.
4. Advanced programming in the Unix environment, W.R.Stevens, Pearson education.

**REFERENCE BOOKS:**

1. Operating System A Design Approach-Crowley,TMH.
2. Modern Operating Systems, Andrew S Tanenbaum 2<sup>nd</sup> edition Pearson/PHI
3. Operating Systems, Dhamdhere, TMH
4. Unix system programming using C++, T.Chan, PHI.
5. Unix programming environment, Kernighan and Pike, PHI. / Pearson Education
6. Unix Internals The New Frontiers, U.Vahalia, Pearson Education.
7. Unix for programmers and users, 3<sup>rd</sup> edition, Graham Glass, King Ables, Pearson Education

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****II Year B.Tech. CSE II-Sem**

L	T	P	C
4	1	0	4

**SOFTWARE ENGINEERING****Objectives:**

- To understand of software process models such as waterfall and evolutionary models.
- To understand of software requirements and SRS document.
- To understand of different software architectural styles.
- To understand of software testing approaches such as unit testing and integration testing.
- To understand on quality control and how to ensure good quality software.
- To Design and Draw (OOAD Objectives)

**UNIT-I:**

**Introduction to Software Engineering:** The evolving role of software, Changing Nature of Software, Software myths. **A Generic view of process:** Software engineering- A layered technology, a process framework, The Capability Maturity Model Integration (CMMI), Process patterns, process assessment, personal and team process models.

**Process models:** The waterfall model, Incremental process models, Evolutionary process models, The Unified process.

**UNIT-II:**

**Software Requirements:** Functional and non-functional requirements, User requirements, System requirements, Interface specification, the software requirements document.

**Requirements engineering process:** Feasibility studies, Requirements elicitation and analysis, Requirements validation, Requirements management.

**System models:** Context Models, Behavioral models, Data models, Object models, structured methods.

**UNIT-III:**

**Design Engineering:** Design process and Design quality, Design concepts, the design model.

**Creating an architectural design:** software architecture, Data design, Architectural styles and patterns, Architectural Design. Conceptual model of UML ,Basic structural Modeling ,Class Diagrams. Sequence Diagrams, Collaboration diagrams, use case diagrams, component diagrams.

**UNIT-IV:**

**Testing Strategies:** A strategic approach to software testing, test strategies for conventional software, Black-Box and White-Box testing, Validation testing, System testing, the art of Debugging.

**Product metrics:** Software Quality, Metrics for Analysis Model, Metrics for Design Model, Metrics for source code, Metrics for testing, Metrics for maintenance.

**UNIT-V:**

**Metrics for Process and Products:** Software Measurement, Metrics for software quality.

**Risk management:** Reactive vs Proactive Risk strategies, software risks, Risk identification, Risk projection, Risk refinement, RMMM, RMMM Plan.

**Quality Management:** Quality concepts, Software quality assurance, Software Reviews, Formal technical reviews, Statistical Software quality Assurance, Software reliability, The ISO 9000 quality standards.

**Text Books:**

1. Software Engineering, A practitioner's Approach- Roger S. Pressman, 6<sup>th</sup> edition. McGrawHill International Edition.
2. Software Engineering- Sommerville, 7<sup>th</sup> edition, Pearson education.
3. The unified Modeling Language user guide Grady Booch ,James Rumbaugh ,Ivar Jacobson:, Pearson Education.

**References:**

1. Software Engineering- K.K. Agarwal & Yogesh Singh, New Age International Publishers
2. Software Engineering, an Engineering approach- James F. Peters, Witold Pedrycz, John Wiley.
3. Systems Analysis and Design- Shely Cashman Rosenblatt, Thomson Publications.
4. Software Engineering principles and practice- Waman S Jawadekar, The McGraw-Hill Companies.
5. Fundamentals of object oriented design using UML Meiler page-Jones:, Pearson Education.
6. Object oriented analysis and Design Atul Kahate:, The MC Graw –Hill Companies.



**JNTUH COLLEGE OF ENGINEERING HYDERABAD****II Year B.Tech. CSE II-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>0</b>	<b>0</b>	<b>3</b>	<b>2</b>

**FILE STRUCTURES LAB**

1. Write a Program to create a file and store a string in it. Write another program to open the file and read the string.
2. Write a Program to read data from keyboard rather than a file ,and writes to a file ,rather than Monitor. Show how to execute the new version of the program in your programming environment ,given that the input is actually in a file .
3. Write a program to read a series of names ,one per line ,from standard input ,and write out those names spelled in reverse order to standard output. Use I/O redirection and pipes to do the following:
  - a. Input a series of names that are typed in from the keyboard ,and write them out ,reversed, to a file .
  - b. Read the names in from a file; then write them out ,re-reversed, to a file .
  - c. Read the a names in from second file , reverse them again, and then sort the resulting list of reversed words using **sort**.
4. Rewrite the insertion (<<)operator of a file .so that it uses the following **fields** representations:
  - a. Method 1 , Fixed length fields.
  - b. Method2, fields with length indicators.
  - c. Method3,fields delimited by “|”.
  - d. Method 4,fields with keyboard tags.
5. Rewrite the Extraction (>>) operator of a file .so that it uses the following **fields** representations:
  - a. Method 1 , Fixed length fields.
  - b. Method2, fields with length indicators.
  - c. Method 4,fields with keyboard tags.
6. Design And implement a class **KeywordBuffer** that pack buffers with keyword tags.
7. Write a **ReadByRRN** method for variable –length record files that finds a record on the basis of its position in the file .For example ,if requested to find the 547<sup>th</sup> record in a file ,it would read-through the first 546 records and then print the contents of 547<sup>th</sup> record .Implement skip sequential search to avoid reading the contents of unwanted records.
8. Write a driver for **findByLastName** that reads names from a separate transaction file that contains only the keys of the records to be extracted. Write the selected records to a separate output file .First ,assume that the records are in no particular order.Then assume that both the main file and the transaction file are sorted by key.In the latter case , make your program more efficient.

9. Implement the **BinarySearch** function for class Person using the canonical form of the combination of last name and the first name as the key . Write a driver program to test the function . Assume that the files are created with using class **RecordFile<person>** using a fixed length buffer.
10. Data Compression Techniques –Suppressing repeating Sequence
11. Reclaiming Space in Files.

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****II Year B.Tech. CSE II-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>0</b>	<b>0</b>	<b>3</b>	<b>2</b>

**OPERATING SYSTEMS LAB**  
**(Using UNIX/LINUX)**

**Objectives:**

At the end of the course, the students should be able to:

- To use Linux operating system for study of operating system concepts.
- To write the code to implement and modify various concepts in operating systems using Linux.

**Outcomes:**

- The course objectives ensure the development of students applied skills in operating systems related areas.
  - Students will gain knowledge in writing software routines, modules for implementing various concepts of operating systems.
1. Write Shell scripts for the following UNIX/LINUX commands:  
a. ls b.cat c.cp d.wc
  2. Write Shell scripts to illustrate the following constructs of UNIX/LINUX operating system:  
a. If b. if-else c. case d. while e. for f. until
  3. Write programs using the I/O system calls of UNIX/LINUX operating system:  
a. open, read, write, close, fcntl, seek, stat, opendir, readdir)
  4. Write C programs to illustrate the following system calls of UNIX/LINUX operating system:  
a.fork b. exec c.getpid, d.exit e. wait
  5. Write C programs to simulate the following CPU Scheduling algorithms:  
a. FCFC b.SJF c. RR, d. PRIORITY
  6. Write a C program to simulate Bankers Algorithm for Deadlock Avoidance and Prevention.
  7. Write C programs to illustrate the following IPC mechanisms:  
a.pipes, b.fifos c.message queues d.shared memory
  8. Write a C program to implement the Producer – Consumer problem using semaphores using UNIX/LINUX system calls.
  9. Write C programs to simulate the following file allocation strategies:  
a. Sequential b. Indexed c. Linked
  10. Write C programs to simulate the following file organization Techniques:  
a. Single level, b.Two level c.Hierarchical d. DAG

11. Write C programs to simulate the following memory management techniques:
  - a. Memory Variable Technique
  - b. Memory Fixed Technique
  - c. Paging, Segmentation
12. Write C programs to simulate the following Page Replacement Techniques:
  - a. FIFO
  - b. LRU
  - c. Optimal

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****II Year B.Tech. CSE II-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>0</b>	<b>0</b>	<b>3</b>	<b>2</b>

**CASE TOOLS & OBJECT ORIENTED ANALYSIS AND DESIGN LAB****USING RATIONAL ROSE - CASE TOOL****LIST OF EXPERIMENTS**

1. Prepare the following documents for two or three of the experiments listed below and develop the software engineering methodology.
2. Program Analysis and Project Planning. Thorough study of the problem - Identify project scope, Objectives, Infrastructure.
3. Software requirement Analysis. Describe the individual Phases / Modules of the project, Identify deliverables.
4. Data Modeling. Use work products - Data dictionary, Use diagrams and activity diagrams, build and test lass diagrams, Sequence diagrams and add interface to class diagrams.
5. Software Development and Debugging
6. Software Testing. Prepare test plan, perform validation testing, Coverage analysis, memory leaks, develop test case hierarchy, Site check and Site monitor.

**SUGGESTED LIST OF APPLICATIONS**

1. Student Marks Analyzing System
2. Quiz System
3. Online Ticket Reservation System
4. Payroll System
5. Course Registration System
6. Expert Systems
7. ATM Systems
8. Stock Maintenance
9. Real-Time Scheduler
10. Remote Procedure Call Implementation
11. Library Management System

**USING WINRUNNER-CASE TOOL**

1. Recording in context sensitive mode and analog mode
2.
  - a) GUI checkpoint for single property
  - b) GUI checkpoint for single object/window
  - c) GUI checkpoint for multiple objects
3.
  - a) Bitmap checkpoint for object/window
  - b) Bitmap checkpoint for screen area
4.
  - a) Database checkpoint for Default check

- b) Database checkpoint for custom check
- c) Database checkpoint for runtime record check
- 5.
  - a) Data driven test for dynamic test data submission
  - b) Data driven test through flat files
  - c) Data driven test through front grids
  - d) Data driven test through excel test
- 6.
  - a) Batch testing without parameter passing
  - b) Batch testing with parameter passing
  - c) Data driven batch
- 7. Silent mode test execution with out any interruption
- 8. Function generator
- 9. User defined functions
- 10. Executing a prepared query
- 11. Compiled module
- 12.
  - a) Synchronization for object/window property
  - b) Synchronization for object/window bitmap
  - c) Synchronization point for screen area bitmap
- 13. Pre-test mode
- 14. Pre-Learning
- 15.
  - a) Test case for calculator in windows application
  - b) Test case for flight Reservation System
- 16. Test plan for library management system

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****II Year B.Tech. CSE II-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>2</b>	<b>0</b>	<b>0</b>	<b>2</b>

**HUMAN VALUES AND PROFESSIONAL ETHICS**

**Unit 1 Human Values:** Morals, values, ethics – integrity – work ethics –service learning – civic virtue – respect for others- living peacefully - Caring –sharing –honesty – courage – valuing time – cooperation – commitment –empathy – self-confidence –spirituality – character- Mini-Cases

**Unit II Professional Ethics:** Profession- and professionalism - Two models of professionalism –Professional etiquette -Three types of Ethics or morality Responsibility in Engineering – Engineering standards –Engineering Ethics – Positive and Negative Faces. Professional Codes and Code of conduct (as given by ASME, ASCE, IEEE, IETE, Institute of Engineers as Guidelines for ethical conduct). Mini-cases.

**Unit III Professional Responsibilities:** Ethical standards Vs Professional Conduct – Zero Tolerance for Culpable Mistakes – Hazards and Risks-Risk benefit analysis– congeniality, collegiality and loyalty. Respect for authority – conflicts of interest – occupational crime — Mini-Cases.

**Unit IV Professional Rights:** professional rights and employee rights communicating risk and public policy – Whistle blowing - collective bargaining. Professionals /engineers as managers, advisors, experts, witnesses and consultants – moral leadership- Regulatory compliances, Monitoring and control- Mini-Cases

**Unit V Ethics in global context:** Global issues in MNCs- Problems of bribery, extortion, and grease payments – Problem of nepotism, excessive gifts – paternalism – different business practices – negotiating taxes. Mini-Cases.

**Mini-projects**

**Project 1:** The student of this course should invariably attend (or watch on internet/any TV channel/youtube/social media) two speeches of 30 minutes duration each dealing with spiritual discourse and submit a report on the contents of the lecture proceedings.

**Project 2:** Visit any organization (including shops/ hotels or shopping malls in your region) of your choice and observe how the professionals perform the given job with a focus on professional ethics and human values.

**References**

1. Aryasri, *Human Values and Professional Ethics*, Maruthi Publications.
2. S B George, *Human Values and Professional Ethics*, Vikas Publishing.
3. KR Govindan & Saenthil Kumar: *Professional Ethics and Human Values*, Anuradha Publications.
4. S K Chakraborty & D.Chakraborty: *Human Values and Ethics*, Himalaya.
5. M. Govindarajan, S. Natarajan, & V.S. Senthilkumar: *Engineering Ethics(Includes Human Values)*, HI Learning Pvt. Ltd., New Delhi – 110001

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****III Year B.Tech. CSE I-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>4</b>	<b>1</b>	<b>0</b>	<b>4</b>

**COMPUTER NETWORKS****Objectives:**

- To introduce the fundamental various types of computer networks.
- To demonstrate the TCP/IP and OSI models with merits and demerits.
- To explore the various layers of OSI Model.
- To introduce UDP and TCP Models.

**UNIT I**

ISO.OSI reference model,TCP/IP reference model, physical layer and media: Data and signals, Digital Transmission, Analog Transmission, Transmission Media, Multiplexing.

**UNIT II****Switching and Data Transmission**

Circuit-switched networks, Datagram networks, Virtual-circuit networks, Structure of a switch, Telephone network, Dial-up modems, Digital Subscriber line, Cable TV networks

**Data Link Layer**

**Error Detection and Correction:** Introduction, Block coding, Linear Block codes, Cyclic codes, Checksum.

**Data Link Control:** Framing, Flow and Error control, Protocols, Noiseless channels, Noisy channels, HDLC, Point-to-Point Protocol

**UNIT III**

**Multiple Access:** Random Access, Controlled Access, Channelization – **Connecting Devices:** Connecting LANs, Backbone Networks, Virtual LANs.

**High Speed Networks**

**Frame Relay:** Packet-Switching Networks, Frame Relay Networks – **Asynchronous Transfer Mode (ATM) :** ATM Protocol Architecture, ATM Logical Connections, ATM Cells, ATM Service Categories, ATM Adaptation Layer (AAL)- **High-Speed LANs :** The Emergence of High-Speed LANs, Ethernet, Fiber Channel, Wireless LANs.

**UNIT IV****Network Layer**

**Logical Addressing:** IPv4 Addresses, IPv6 Addresses, - **Internet Protocol:** Internetworking, IPv4, IPv6, Transition from IPv4 to IPv6 - **Network Delivery - Routing:** Forwarding, Unicast Routing Protocols, Multicast Routing Protocols

**Transport Layer**

Process-to-Process delivery, User Datagram Protocol (UDP), TCP, SCTP

**UNIT V**

**Congestion control:** Data traffic, Congestion, Congestion control, Quality of Service  
Application layer: DNS, SNMP, WWW and HTTP,Email

**Text Books:**

1. Data Communications and Networking, *Behrouz A. Forouzan*, Fourth Edition, Tata McGraw Hill
2. Computer Networks, *Andrew S. Tanenbaum*, Fourth Edition, Prentice Hall..



3. Computer Networks- A Systems Approach, Larry L. Peterson and Bruce S. Davie, Elsevier.

**References:**

1. Computer Networking-A Top-Down Approach Featuring the Internet-James F. Kurose, Keith W. Ross, Pearson Education.
2. An Engineering Approach to Computer Networking, *S. Keshav*, Pearson Education.

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****III Year B.Tech. CSE I-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>4</b>	<b>1</b>	<b>0</b>	<b>4</b>

**COMPILER DESIGN****Objectives:**

- To describe the steps and algorithms used by language translators.
- To discuss the effectiveness of optimization.
- To explain the machine dependent aspects of Compilation

**Outcomes:**

- Ability to understand the design of a compiler given features of the languages.
- Ability to implement practical aspects of automata theory.
- Gain Knowledge of powerful compiler generation tools.
- Ability to design a simple compiler

**UNIT – I:**

**Overview of Compilation:** Phases of Compilation – Lexical Analysis, Regular Grammar and regular expression for common programming language features, pass and Phases of translation, interpretation, bootstrapping, data structures in compilation – LEX lexical analyzer generator.

**UNIT – II:**

**Top down Parsing:** Context free grammars, Top down parsing – Backtracking, LL (1), recursive descent parsing, Predictive parsing, Non recursive predictive parsing.

**Bottom up parsing:** Shift Reduce parsing, Operator precedence parsing, LR parsers, Error recovery in parsing, handling ambiguous grammar, YACC – automatic parser generator.

**UNIT – III**

**Syntax – Directed Translation:** Syntax Directed Definitions, Construction of Syntax Trees, Bottom Evaluation of S-attributed Definitions, attributed definitions, Top down translation, Bottom-up evaluation of inherited attributes.

**Type Checking:** Type systems, Specification of a simple type checker, equivalence of type expressions.

**Storage Organization:** storage allocation strategies, parameter passing.

**Symbol table,** Language facilities for dynamic storage.

**UNIT IV**

**Intermediate Code generation:** Intermediate languages for code generation.

**Code Generation:** Issues in the design of a code generator, the target machine run time storage management, Next use information, DAG representation of basic blocks. Dynamic Programming code generation algorithm.

**UNIT V**

**Code Optimization:** Introduction, Principal source of optimization, local optimization, loop optimization, frequency reduction, folding.

**Data flow analysis:** Flow graph, data flow equation, global optimization, redundant sub expression elimination, Induction variable elements, Live variable analysis, Copy propagation.

**Text Books:**

1. Compilers: Principles, Techniques and Tools: Alfred V.Aho,Ravi Sethi, Jeffrey D. Ullman; Pearson Education
2. Modern Compiler Implementation in C- Andrew N. Appel, Cambridge University Press.

**References:**

1. lex &yacc – John R. Levine, Tony Mason, Doug Brown, O'reilly
2. Modern Compiler Design- Dick Grune, Henry E. Bal, Cariel T. H. Jacobs, Wiley dreamtech.
3. Engineering a Compiler-Cooper & Linda, Elsevier.
4. Compiler Construction, Loudon, Thomson..

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****III Year B.Tech. CSE I-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>4</b>	<b>1</b>	<b>0</b>	<b>4</b>

**DATABASE MANAGEMENT SYSTEMS****Objectives:**

- To understand the basic concepts and the applications of database systems.
- To master the basics of SQL and construct queries using SQL.
- To understand the relational database design principles.
- To become familiar with the basic issues of transaction processing and concurrency control.
- To become familiar with database storage structures and access techniques.

**UNIT I :**

**Data base System Applications:** data base System VS file System, View of Data, Data Abstraction ,Instances and Schemas, data Models ,the ER Model, Relational Model, Other Models, Database Languages, DDL, DML, database Access for applications Programs ,data base Users and Administrator ,Transaction Management, data base System Structure, Storage Manager, the Query Processor. History of Data base Systems. Data base design and ER diagrams, Beyond ER Design Entities, Attributes and Entity sets, Relationships and Relationship sets, Additional features of ER Model ,Concept Design with the ER Model, Conceptual Design for Large enterprises.

**UNIT II:**

**Introduction to the Relational Model:** Integrity Constraint Over relations ,Enforcing Integrity constraints , Querying relational data , Logical data base Design ,Introduction to Views, Destroying /altering Tables and Views. Form of Basic SQL Query, Examples of Basic SQL Queries, Introduction to Nested Queries ,Correlated Nested Queries Set, Comparison Operators, Aggregative Operators, NULL values ,Comparison using Null values ,Logical connectivity's, AND, OR and NOT, Impact on SQL Constructs Outer Joins, Disallowing NULL values ,Complex Integrity Constraints in SQL Triggers and Active Data bases, Oracle, SQL Server,DB2.

**UNIT III:**

**Relational Algebra :**Selection and projection set operations, renaming ,Joins ,Division, Examples of Algebra overviews, Relational calculus, Tuple relational Calculus, Domain relational calculus ,Expressive Power of Algebra and calculus.

**Schema refinement:** Problems Caused by redundancy, Decompositions, Problem related to decomposition, reasoning about FDS,FIRST, SECOND, THIRD Normal forms ,BCNF, Lossless join Decomposition ,Dependency preserving Decomposition, Schema refinement in Data base Design ,Multi valued Dependencies , FORTH Normal Form, FIFTH Normal Form.

**UNIT IV:**

Transaction Concept, Transaction State, Implementation of Atomicity and Durability, Concurrent Executions , Serializability, Recoverability ,Implementation of Isolation ,Testing for serializability,Lock Based Protocols , Timestamp Based Protocols, Validation- Based Protocols ,Multiple Granularity.

Recovery and Atomicity ,Log-Based Recovery ,Recovery with Concurrent Transactions, Buffer Management, Failure with loss of nonvolatile storage, Advance Recovery systems, Remote Backup systems.

**UNIT V:**

Data on External Storage, File Organization and Indexing, Cluster Indexes, Primary and Secondary Indexes, Index data Structures ,Hash Based Indexing ,Tree base Indexing ,Comparison of File Organizations ,Indexes and Performance Tuning, Intuitions for tree Indexes, Indexed Sequential Access Methods (ISAM), B+ Trees: A Dynamic Index Structure.

**Text Books :**

1. Data base Management Systems, Raghurama Krishnan, Johannes Gehrke, TATA McGrawHill 3rd Edition
2. Data base System Concepts, Silberschatz, Korth, McGraw hill, V edition.

**References :**

1. Data base Systems design, Implementation, and Management, Peter Rob & Carlos Coronel 7th Edition.
2. Fundamentals of Database Systems, Elmasri Navrate Pearson Education
3. Introduction to Database Systems, C.J.Date Pearson Education
4. Oracle for Professionals, The X Team, S.Shah and V.Shah, SPD.
5. Database Systems Using Oracle: A Simplified guide to SQL and PL/SQL, Shah, PHI.
6. Fundamentals of Database Management Systems, M.L.Gillenson, Wiley Student Edition.

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****III Year B.Tech. CSE I-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>4</b>	<b>0</b>	<b>0</b>	<b>4</b>

**OPEN ELECTIVE-I  
GIS & REMOTE SENSING**

**UNIT – I**

Introduction to Photogrammetry: Principle and types of aerial photographs, stereoscopy, Map Vs Mosaic, ground control, Parallax measurements for height, determinations.

Remote Sensing – I: Basic concepts and foundation of remote sensing – elements involved in remote sensing, electromagnetic spectrum, remote sensing terminology and units.

**UNIT - II**

Remote Sensing – II: Energy resources, energy interactions with earth surface features and atmosphere, resolution, sensors and satellite visual interpretation techniques, basic elements, converging evidence, interpretation for terrain evaluation, spectral properties of water bodies, introduction to digital data analysis.

**UNIT – III**

Geographic Information System: Introduction, GIS definition and terminology, GIS categories, components of GIS, fundamental operations of GIS, A theoretical framework for GIS.

Types of data representation: Data collection and input overview, data input and output. Keyboard entry and coordinate geometry procedure, manual digitizing and scanning, Raster GIS, Vector GIS – File management, Spatial data – Layer based GIS, Feature based GIS mapping.

**UNIT – IV**

GIS Spatial Analysis: Computational Analysis Methods (CAM), Visual Analysis Methods (VAM), Data storage-vector data storage, attribute data storage, overview of the data manipulation and analysis. Integrated analysis of the spatial and attribute data.

**UNIT – V**

Water Resources Applications-I: Land use/Land cover in water resources, Surface water mapping and inventory, Rainfall – Runoff relations and runoff potential indices of watersheds, Flood and Drought impact assessment and monitoring, Watershed management for sustainable development and Watershed characteristics.

Water Resources Applications – II: Reservoir sedimentation, Fluvial Geomorphology, water resources management and monitoring, Ground Water Targeting, Identification of sites for artificial Recharge structures, Drainage Morphometry, Inland water quality survey and management, water depth estimation and bathymetry.

**TEXT BOOKS:**

1. Remote Sensing and its applications by LRA Narayana University Press 1999.
2. Principals of Geo physical Information Systems – Peter A Burragh and Rachael A. Mc Donnell, Oxford Publishers 2004.

**REFERENCES:**

1. Concepts & Techniques of GIS by C.P.Lo Albert, K.W. Yonng, Prentice Hall (India) Publications.
2. Remote Sensing and Geographical Information systems by M.Anji Reddy JNTU Hyderabad 2001, B.S.Publications.
3. GIS by Kang – tsung chang, TMH Publications & Co.,
4. Basics of Remote sensing & GIS by S.Kumar, Laxmi Publications.
5. Fundamental of GIS by Mechanical designs John Wiley & Sons.

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****III Year B.Tech. CSE I-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>4</b>	<b>0</b>	<b>0</b>	<b>4</b>

**OPEN ELECTIVE-I**  
**NON CONVENTIONAL POWER GENERATION**

**UNIT - I**

Fundamentals of Solar Energy-Solar spectrum- Solar Radiation on Earth's surface-Solar radiation geometry-Solar radiation measurements- Solar radiation data- Solar radiation on horizontal and tilted surfaces. Solar Thermal conversion- Flat plate collectors- concentrated collectors- construction and thermal analysis- Solar applications- Solar ponds- Heliostat systems-water heater-air heater-solar still.

**UNIT - II**

Solar-Electric Power generation- Photovoltaic cells- Equivalent circuit- V-I Characteristics- Photovoltaic modules – constructional details- design considerations- Tracking- Maximum power point tracking – algorithms - PV solar system design with energy back up- Solar Thermo electric conversion.

**UNIT - III**

Wind Energy- Fundamentals of wind energy-power available in wind- Betz Limit- Aerodynamics of wind turbine- Wind turbines- Horizontal and vertical axis turbines –their configurations- Wind Energy conversion systems.

**UNIT - IV**

Energy from Bio Mass- Various fuels- Sources-Conversion technologies-Wet Processes – Dry Processes- Bio Gas generation – Aerobic and anaerobic digestion-Factors affecting generation of bio gas –Classification of bio gas plants-Different Indian digesters- Digester design considerations- Gasification process-Gasifiers – Applications. Geothermal Energy-sources-Hydrothermal convective- Geo-pressure resources- Petro-thermal systems(HDR)- Magma Resources-Prime Movers.

**UNIT - V**

OTEC Systems- Principle of operation-Open and closed cycles, Energy from Tides- Principle of Tidal Power- Components of tidal Power plants-Operation Methods-Estimation of Energy in Single and double basin systems- Energy and Power from Waves-Wave energy conversion devices- Fuel Cells-Design and Principle of operation-Types of Fuel Cells- Advantages and disadvantages-Types of Electrodes- Applications-Basics of Batteries – Constructional details of Lead acid batteries- Ni-Cd Batteries.

**Text Books**

1. John Twidell & Wier, Renewable Energy Resources, CRC Press, 2009.
2. G.D.Rai – Non Conventional Energy sources, Khanna publishers.

**References**

1. D.P .Kothari, Singal,Rakesh, Ranjan, Renewable Energy sources and Emerging Technologies, PHI, 2009.
2. F.C.Treble, Generating Electricity from Sun.



3. C.S.Solanki , Solar Photo volatics- Fundamentls- Principles and Applications, PHI 2009
4. S.P.Sukhatme , Solar Energy Principles and Application - TMH

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****III Year B.Tech. CSE I-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>4</b>	<b>0</b>	<b>0</b>	<b>4</b>

**OPEN ELECTIVE-I**  
**OPERATIONS RESEARCH**

**UNIT – I**

Development – Definition– Characteristics and Phases – Types of models – Operations Research models – applications.

**ALLOCATION:** Linear Programming Problem - Formulation – Graphical solution – Simplex method – Artificial variables techniques: Two-phase method, Big-M method; Duality Principle.

**UNIT – II**

**TRANSPORTATION PROBLEM** – Formulation – Optimal solution, unbalanced transportation problem – Degeneracy.

**Assignment problem** – Formulation – Optimal solution - Variants of Assignment Problem; Traveling Salesman problem.

**UNIT – III**

**SEQUENCING** – Introduction – Flow –Shop sequencing – n jobs through two machines – n jobs through three machines – Job shop sequencing – two jobs through ‘m’ machines

**REPLACEMENT:** Introduction – Replacement of items that deteriorate with time – when money value is not counted and counted – Replacement of items that fail completely- Group Replacement.

**UNIT – IV**

**THEORY OF GAMES:** Introduction –Terminology– Solution of games with saddle points and without saddle points- 2 x 2 games –m x 2 & 2 x n games - graphical method – m x n games - dominance principle.

**INVENTORY:** Introduction – Single item, Deterministic models – Types - Purchase inventory models with one price break and multiple price breaks –Stochastic models – demand discrete variable or continuous variable – Single Period model with no setup cost.

**UNIT – V**

**WAITING LINES:** Introduction – Terminology-Single Channel – Poisson arrivals and Exponential Service times – with infinite population and finite population models– Multichannel – Poisson arrivals and exponential service times with infinite population.

**DYNAMIC PROGRAMMING:**

Introduction – Terminology- Bellman’s Principle of Optimality – Applications of dynamic programming- shortest path problem – linear programming problem.

**TEXT BOOK :**

1. Operation Research /J.K.Sharma/MacMilan.
2. Introduction to O.R /Taha/PHI

**REFERENCE BOOKS :**

1. Operations Research: Methods and Problems / Maurice Saseini, Arthur Yaspan and Lawrence Friedman
2. Operations Research /A.M.Natarajan, P.Balasubramaniam, A. Tamilarasi/Pearson Edu.
3. Operations Research / Wagner/ PHI Publications.
4. Introduction to O.R/Hillier & Libermann (TMH).

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****III Year B.Tech. CSE I-Sem****L T P C**  
**4 0 0 4****OPEN ELECTIVE-I**  
**ELECTRONIC MEASUREMENTS AND INSTRUMENTATION****UNIT I:**

**Block Schematics of Measuring Systems:** Performance Characteristics, Static Characteristics, Accuracy, Precision, Resolution, Types of Errors, Gaussian Error, Root Sum Squares formula, Dynamic Characteristics, Repeatability, Reproducibility, Fidelity, Lag ;Measuring Instruments: DC Voltmeters, D' Arsonval Movement, DC Current Meters, AC Voltmeters and Current Meters, Ohmmeters, Multimeters, Meter Protection, Extension of Range, True RMS Responding Voltmeters, Specifications of Instruments.

**.UNIT II:**

**Signal Analyzers:** AF, HF Wave Analyzers, Harmonic Distortion, Heterodyne wave Analyzers, Spectrum Analyzers, Power Analyzers, Capacitance-Voltage Meters, Oscillators. **Signal Generators:** AF, RF Signal Generators, Sweep Frequency Generators, Pulse and Square wave Generators, Function Generators, Arbitrary Waveform Generator, Video Signal Generators, and Specifications

**UNIT III:**

**Oscilloscopes:** CRT, Block Schematic of CRO, Time Base Circuits, Lissajous Figures, CRO Probes, High Frequency CRO Considerations, Delay lines, Applications: Measurement of Time, Period and Frequency Specifications.

**Special Purpose Oscilloscopes:** Dual Trace, Dual Beam CROs, Sampling Oscilloscopes, Storage Oscilloscopes, Digital Storage CROs.

**UNIT IV:**

**Transducers:** Classification, Strain Gauges, Bounded, unbounded; Force and Displacement Transducers, Resistance Thermometers, Hotwire Anemometers, LVDT, Thermocouples, Synchros, Special Resistance Thermometers, Digital Temperature sensing system, Piezoelectric Transducers, Variable Capacitance Transducers, Magneto Strictive Transducers.

**UNIT V:**

**Bridges:** Wheat Stone Bridge, Kelvin Bridge, and Maxwell Bridge.

**Measurement of Physical Parameters:** Flow Measurement, Displacement Meters, Liquid level Measurement, Measurement of Humidity and Moisture, Velocity, Force, Pressure – High Pressure, Vacuum level, Temperature -Measurements, Data Acquisition Systems.

**TEXT BOOKS:**

1. Electronic Measurements and Instrumentation – K. Lal Kishore, Pearson Education 2010.
2. Electronic Instrumentation: H.S.Kalsi – TMH, 2<sup>nd</sup> Edition 2004.

**REFERENCES:**

1. Electronic Instrumentation and Measurements – David A. Bell, Oxford Univ. Press, 1997.
2. Modern Electronic Instrumentation and Measurement Techniques: A.D. Helbins, W.D. Cooper: PHI 5<sup>th</sup> Edition 2003.

3. Electronic Measurements and Instrumentation: B.M. Oliver, J.M. Cage TMH Reprint 2009.
4. Industrial Instrumentation: T.R. Padmanabham Springer 2009.

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****III Year B.Tech. CSE I-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>4</b>	<b>0</b>	<b>0</b>	<b>4</b>

**OPEN ELECTIVE-I**  
**OBJECT ORIENTED PROGRAMMING THROUGH JAVA**

**UNIT I :**

**Object oriented thinking and Java Basics-** Need for oop paradigm, summary of oop concepts, coping with complexity, abstraction mechanisms. A way of viewing world – Agents, responsibility, messages, methods, History of Java, Java buzzwords, data types, variables, scope and life time of variables, arrays, operators, expressions, control statements, type conversion and casting, simple java program, concepts of classes, objects, constructors, methods, access control, this keyword, garbage collection, overloading methods and constructors, method binding, inheritance, overriding and exceptions, parameter passing, recursion, nested and inner classes, exploring string class.

**UNIT II :**

**Inheritance, Packages and Interfaces** – Hierarchical abstractions, Base class object, subclass, subtype, substitutability, forms of inheritance- specialization, specification, construction, extension, limitation, combination, benefits of inheritance, costs of inheritance. Member access rules, super uses, using final with inheritance, polymorphism- method overriding, abstract classes, the Object class.

Defining, Creating and Accessing a Package, Understanding CLASSPATH, importing packages, differences between classes and interfaces, defining an interface, implementing interface, applying interfaces, variables in interface and extending interfaces.

Exploring java.io.

**UNIT III**

**Exception handling and Multithreading--** Concepts of exception handling, benefits of exception handling, Termination or resumptive models, exception hierarchy, usage of try, catch, throw, throws and finally, built in exceptions, creating own exception sub classes.

String handling, Exploring java.util. Differences between multi threading and multitasking, thread life cycle, creating threads, thread priorities, synchronizing threads, interthread communication, thread groups, daemon threads.

Enumerations, autoboxing, annotations, generics.

**UNIT IV :**

**Event Handling** : Events, Event sources, Event classes, Event Listeners, Delegation event model, handling mouse and keyboard events, Adapter classes.

The AWT class hierarchy, user interface components- labels, button, canvas, scrollbars, text components, check box, check box groups, choices, lists panels – scrollpane, dialogs, menubar, graphics, layout manager – layout manager types – border, grid, flow, card and grid bag.

**UNIT V :**

**Applets** – Concepts of Applets, differences between applets and applications, life cycle of an applet, types of applets, creating applets, passing parameters to applets.

**Swing** – Introduction, limitations of AWT, MVC architecture, components, containers, exploring swing- JApplet, JFrame and JComponent, Icons and Labels, text fields, buttons –

The JButton class, Check boxes, Radio buttons, Combo boxes, Tabbed Panes, Scroll Panes, Trees, and Tables.

**TEXT BOOKS :**

1. Java the complete reference, 7<sup>th</sup> edition, Herbert Schildt, TMH.
2. Understanding OOP with Java, updated edition, T. Budd, Pearson Education.

**REFERENCES :**

1. An Introduction to programming and OO design using Java, J.Nino and F.A. Hosch, John Wiley & Sons.
2. An Introduction to OOP, third edition, T. Budd, Pearson Education.
3. Introduction to Java programming, Y. Daniel Liang, Pearson Education.
4. An introduction to Java programming and object oriented application development, R.A. Johnson- Thomson.
5. Core Java 2, Vol 1, Fundamentals, Cay.S.Horstmann and Gary Cornell, eighth Edition, Pearson Education.
6. Core Java 2, Vol 2, Advanced Features, Cay.S.Horstmann and Gary Cornell, eighth Edition, Pearson Education
7. Object Oriented Programming with Java, R.Buyya, S.T.Selvi, X.Chu, TMH.
8. Java and Object Orientation, an introduction, John Hunt, second edition, Springer.
9. Maurach's Beginning Java2 JDK 5, SPD.
10. Programming and Problem Solving with Java, JM Slack, B S Publications.

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****III Year B.Tech. CSE I-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>4</b>	<b>0</b>	<b>0</b>	<b>4</b>

**OPEN ELECTIVE-I**  
**DATA STRUCTURES & ANALYSIS OF ALGORITHMS**

**Objectives:**

- To understand the basic concepts such as Abstract Data Types. Linear and Non Linear Data Structure.
- To understand the notations used to analyze the performance of algorithms
- To understand the behavior of data structures such as Trees, Graphs and their representation
- To choose the appropriate data structure for a specified application
- To analyze performance of algorithms
- To choose the appropriate data structure and algorithm design method for a specified application
- To understand how the choice of data structures and algorithms design methods impacts the performance of programs
- To solve problems using algorithms design methods such as the greedy method, divide and conquer, dynamic programming, Prerequisites Data structures, Mathematical foundations of computer science

**Unit I:**

**C++ Class Overview-** Class Definition, Objects, Class Members, Access Control, Class Scope, Constructors and destructors, parameter passing methods, Inline functions, static class members, this pointer, friend functions, dynamic memory allocation and deal location (new and delete), exception handling.

Function Over Loading, Operator Overloading, Generic Programming- Function and class templates, Inheritance basics, base and derived classes, inheritance types, base class access control, runtime polymorphism using virtual functions, abstract classes, streams I/O.

**Unit II:**

**Algorithms**, performance analysis- time complexity and space complexity. Review of basic data structures- The list ADT, Stack ADT, Queue ADT, Implementation using template classes in C++.

Dictionaries, linear list representation, skip list representation, operations insertion, deletion and searching.

**Unit III:**

**Trees:** Definition, ADT, Trees Implementation Methods. Trees Traversal Methods

**Graphs:** Definition, ADT, Graphs Implementation Methods. Graphs Traversal Methods.

**Priority Queues** – Definition, ADT, Realizing a Priority Queue using Heaps, Definition, insertion, Deletion.

**Unit IV:**

**Algorithms:** Pseudo code for expressing algorithms, Performance Analysis-Space complexity, Time complexity, Asymptotic Notation- Big oh notation, Omega notation, Theta notation and Little oh notation, Probabilistic analysis, Amortized analysis.

**Divide and conquer:** General method, applications-Binary search, Quick sort, Merge sort, Strassen's matrix multiplication.

#### **Unit V:**

**Dynamic Programming:** General method, applications-Matrix chain multiplication, Optimal binary search trees, 0/1 knapsack problem, All pairs shortest path problem, Travelling sales person problem, Reliability design.

#### **TEXT BOOKS:**

1. Data structures, Algorithms and Applications in C++, S.Sahni, University Press (India) Pvt. Ltd, 2nd edition, Universities Press Orient Longman Pvt. Ltd.
2. Data structures and Algorithms in C++, Michael T.Goodrich, R.Tamassia and .Mount, Wiley student edition, John Wiley and Sons.

#### **REFERENCES :**

1. Data structures and Algorithm Analysis in C++, Mark Allen Weiss, Pearson Education. Ltd., Second Edition.
2. Data structures and algorithms in C++, 3rd Edition, Adam Drozdek, Thomson
3. Data structures using C and C++, Langsam, Augenstein and Tanenbaum, PHI.
4. Problem solving with C++, The OOP, Fourth edition, W.Savitch, Pearson education



**JNTUH COLLEGE OF ENGINEERING HYDERABAD****III Year B.Tech. CSE I-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>4</b>	<b>0</b>	<b>0</b>	<b>4</b>

**OPEN ELECTIVE-I  
OPERATING SYSTEMS**

**Objectives:**

- Provide an introduction to operating system concepts (i.e., processes, threads, scheduling, synchronization, deadlocks, memory management, file and I/O subsystems and protection) as fundamental principles by reference to real systems
- To give exposure to the professional responsibilities that are part of operating system design and development.
- To provide the student with the ability to write system oriented programs on Unix/Linux.

**UNIT I:**

**Operating System Introduction**, Structures - Simple Batch, Multi programmed, Time-shared, Personal Computer, Parallel, Distributed Systems, Real-Time Systems, System components, Operating-System services, System Calls, Virtual Machines, System Design and Implementation.

**UNIX/LINUX Utilities** - Introduction to Unix file system, vi editor, file handling utilities, security by file permissions, process utilities, disk utilities, networking commands, text processing utilities and backup utilities,

**Working with Bash shell:** what is a shell, shell responsibilities, pipes and input Redirection, output redirection, here documents, the shell as a programming language, shell meta characters, shell variables, shell commands, the environment, control structures, shell script examples.

**UNIT II:**

**Process and CPU Scheduling** - Process concepts and scheduling, Operation on processes, Cooperating Processes, Threads, and Interposes Communication Scheduling Criteria, Scheduling Algorithm, Multiple -Processor Scheduling, Real-Time Scheduling.

**System call interface for process management**-fork, exit, wait, waitpid, exec,

**Deadlocks** - System Model, Dead locks Characterization, Methods for Handling Dead locks Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, and Recovery from Deadlock.

**UNIT – III:**

**Process Management and Synchronization** - The Critical Section Problem, Synchronization Hardware, Semaphores, and Classical Problems of Synchronization, Critical Regions, Monitors.

**Interprocess Communication Mechanisms:** IPC between processes on a single computer system, IPC between processes on different systems, using pipes, fifos, message queues, shared memory, semaphores.

**UNIT IV**

**Memory Management and Virtual Memory** - Logical versus Physical Address Space, Swapping, Contiguous Allocation, Paging, Segmentation, Segmentation with Paging. Demand Paging, Performance of Demanding Paging, Page Replacement, Page Replacement Algorithm, Allocation of Frames, Thrashing.

**UNIT V:**

**File System Interface and Implementation** -Access methods, Directory Structure, Protection, File System Structure, Allocation methods, Free-space Management, Directory Management, Directory Implementation, Efficiency and Performance.

**Unix/LINUX Files:** File structure, directories, files and devices, System calls, library functions, low level file access, usage of open, creat, read, write, close, lseek, stat, ioctl.

**TEXT BOOKS:**

1. Operating System Principles- Abraham Silberchatz, Peter B. Galvin, Greg Gagne 7<sup>th</sup> Edition, John Wiley
2. Operating Systems – Internals and Design Principles Stallings, Fifth Edition–2005, Pearson Education/PHI
3. Unix the ultimate guide, Sumitabha Das, TMH.
4. Advanced programming in the Unix environment, W.R.Stevens, Pearson education.

**REFERENCE BOOKS:**

1. Operating System A Design Approach-Crowley,TMH.
2. Modern Operating Systems, Andrew S Tanenbaum 2<sup>nd</sup> edition Pearson/PHI
3. Operating Systems, Dhamdhare, TMH
4. Unix system programming using C++, T.Chan, PHI.
5. Unix programming environment, Kernighan and Pike, PHI. / Pearson Education
6. Unix Internals The New Frontiers, U.Vahalia, Pearson Education.
7. Unix for programmers and users, 3<sup>rd</sup> edition, Graham Glass, King Ables, Pearson Education

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****III Year B.Tech. CSE I-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>4</b>	<b>0</b>	<b>0</b>	<b>4</b>

**OPEN ELECTIVE-I  
MATERIALS SCIENCE****UNIT – 1**

Engineering materials. Mechanical properties. Structure-property relation. Grains and grain boundaries. Slip. Determination of grain size. Microstructure and strength. Crystal structure and ductility.

**UNIT – 2**

Constitution of alloys. Necessity of alloying. Types of solid solutions. Hume-Rothery rules. Intermediate alloy phases. Intermetallic compounds. Electron compounds. Phase diagrams. Construction and interpretation. Isomorphous and eutectic systems. Cu-Ni and Pb-Sn phase diagrams. Lever rule. Phase rule.

**UNIT - 3**

The iron-carbon phase diagram. Polymorphism. Phases. Structure and properties of hypoeutectoid and hypereutectoid steels and cast irons ( white cast iron, malleable cast iron, gray cast iron, nodular cast iron).

**UNIT-4:**

Heat treatment of steels. TTT diagrams. Annealing, normalizing, hardening, tempering. Effect of alloying elements on the Fe-C diagram and TTT diagram. Hardenability. Heat treatment of nonferrous alloys. Precipitation hardening. Al-Cu phase diagram. Composite materials I. Particle-reinforced composites (Cu-Al<sub>2</sub>O<sub>3</sub>, WC-Co). Manufacturing techniques.

**UNIT-5:**

Ceramics. Crystalline ceramics. Classification: Clay products, Refractories, Abrasives. Applications. Glasses. Strain point, annealing point, softening point, working point, melting point. Composite materials II. Fiber-reinforced composites. Role of fibre phase and matrix phase. Polymer-matrix, Metal-matrix, and transformation-toughened ceramic matrix composites. Processing and Applications.

**TEXT BOOKS**

1. Materials Science and Engineering. An introduction, WD Callister, Jr., Adapted by R. Balasubramaniam, John Wiley & Sons, NY, Indian edition, 2007
2. Metallurgy for Engineers – Clark and Varney
3. Elements of Materials Science – V Raghavan

**REFERENCE BOOKS**

1. Foundations of Materials Science and Engineering – WF Smith
2. C. Suryanarayana, Experimental Techniques in Mechanics and Materials, JW, NJ, USA, 2006
3. Introduction to Physical Metallurgy , SH Avner, Tata McGraw-Hill edition, 1997

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****III Year B.Tech. CSE I-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>4</b>	<b>0</b>	<b>0</b>	<b>4</b>

**OPEN ELECTIVE-I  
NANOTECHNOLOGY**

**Unit 1: Background of Nanotechnology**

Scientific Revolutions, Nanotechnology and Nanomachines, The Periodic Table, Atomic Structure, Molecules and Phases, Energy, Molecular and Atomic size, Surfaces and Dimensional Space, Top down and Bottom up approach.

**Unit 2: Molecular Nanotechnology**

Atoms by inference, Electron Microscopes, Scanning electron microscope, Modern transmission electron microscope, Scanning probe microscope-atomic force microscope, Scanning tunneling microscope, Self Assembly.

**Unit 3: Nanopowders and Nanomaterials**

What are nanomaterials? Preparation, Plasma arcing, chemical vapor deposition, Sol-gels, Electrodeposition, Ball milling, using natural nanoparticles, Applications of nanomaterials.

**Unit 4: Nanoelectronics**

Approaches to nanoelectronics, Fabrication of integrated circuits, MEMS, NEMS, Nano circuits, Quantum wire, Quantum well, DNA-directed assembly and application in electronics.

**Unit 5: Applications**

MEMS, NEMS, Coatings, Optoelectronic Devices, Environmental Applications, Nanomedicine.

**Text Books**

1. Introduction to Nanoscience and Nanotechnology Gabor L. Hornyak, **NanoThread, Inc., Golden, Colorado, USA**; H.F. Tibbals, **University of Texas Southwestern Medical Center, Dallas, USA**; Joydeep Dutta, **Asian Institute of Technology, Pathumthani, Thailand**; John J. Moore, **Colorado School of Mines, Golden, USA**
2. Introduction to Nanotechnology by Charles P. Poole Jr and Frank J. Owens Wiley India Pvt Ltd.
3. Introduction to Nanoscience and Nanotechnology, Chatopadhyaya.K.K, and Banerjee A.N,
4. Introduction to nano tech by phani kumar
5. Introduction to Nano Technology by Charles P. Poole Jr and Frank J. Owens. Wiley India Pvt Ltd.
6. Introduction to Nanoscience and Nanotechnology, Chatopadhyaya.K.K, and Banerjee A.N,  
NANOTECHNOLOGY Basic Science and Emerging Technologies by Michael Wilson, Kamali Kannangara Geoff Smith, Michelle Simmons, Burkhard Raguse- CHAPMAN & HALL/CRC PRESS 2002.

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****III Year B.Tech. CSE I-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>4</b>	<b>0</b>	<b>0</b>	<b>4</b>

**OPEN ELECTIVE-I**  
**ENGINEERING MANAGEMENT**

**Course objectives:**

1. to sensitise and orient the future engineers about the challenges in managing engineering enterprises
2. to teach how to provide value through innovations, leadership in technology projects, and the application of emerging technologies through web-based tools

**Unit-I**

Introduction to Engineering Management - Management Challenges For Engineers – Planning –Organizing –Leading- Controlling-Value Engineering

**Unit-II**

Cost Accounting for Engineering Managers-Financial Accounting and Analysis for Engineering Managers- Managerial Finance for Engineering Managers

**Unit-III**

Project management – Total Quality Management –New product design – Production planning and control –Process planning – Maintenance Management – Marketing Management for Engineering Managers.

**Unit-IV**

Engineers as Managers/Leaders- Ethics In Engineering/Business Management. – Business Process Re-engineering-Ergonomics – Group Technology.

**Unit-V**

Advanced Manufacturing Technologies and systems -Web-Based Enablers For Engineering And Management- Globalization- Engineering Management In The New Millennium

**Text Book:**

1. C M Chang, Engineering Management: Challenges in the New Millennium, Pearson, 2013.

**References:**

1. Martand Telsang, Industrial Engineering and Production Management,S. Chand, Second edition.
2. A.K. Gupta, Engineering Management, S.Chand,2010.

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****III Year B.Tech. CSE I-Sem**

L	T	P	C
4	0	0	4

**MANAGERIAL ECONOMICS AND FINANCIAL ANALYSIS**

**Unit I Introduction & Demand Analysis:** Definition, Nature and Scope of Managerial Economics. Demand Analysis: Demand Determinants, Law of Demand and its exceptions. *Elasticity of Demand:* Definition, Types, Measurement and Significance of Elasticity of Demand. *Demand Forecasting,* Factors governing demand forecasting, methods of demand forecasting.

**Unit II Production & Cost Analysis:** *Production Function* – Isoquants and Isocosts, MRTS, Least Cost Combination of Inputs, Cobb-Douglas Production function, Laws of Returns, Internal and External Economies of Scale. *Cost Analysis:* Cost concepts. Break-even Analysis (BEA)-Determination of Break-Even Point (simple problems) - Managerial Significance.

**Unit III Markets & New Economic Environment:** Types of competition and Markets, Features of Perfect competition, Monopoly and Monopolistic Competition. Price-Output Determination in case of Perfect Competition and Monopoly. *Pricing:* Objectives and Policies of Pricing. Methods of Pricing. *Business:* Features and evaluation of different forms of Business Organisation: Sole Proprietorship, Partnership, Joint Stock Company, Public Enterprises and their types, *New Economic Environment:* Changing Business Environment in Post-liberalization scenario.

**Unit IV Capital Budgeting:** Capital and its significance, Types of Capital, Estimation of Fixed and Working capital requirements, Methods and sources of raising capital Trading Forecast Capital Budget, Cost Budget. Capital Budgeting: features of capital budgeting proposals, Methods of Capital Budgeting: Payback Method, Accounting Rate of Return (ARR) and Net Present Value Method (simple problems).

**Unit V Introduction to Financial Accounting & Financial Analysis:** Accounting concepts and Conventions Introduction IFRS Double-Entry Book Keeping, Journal, Ledger, Trial Balance- Final Accounts (Trading Account, Profit and Loss Account and Balance Sheet with simple adjustments). *Financial Analysis:* Analysis and Interpretation of Liquidity Ratios, Activity Ratios, and Capital structure Ratios and Profitability ratios. Du Pont Chart.

**TEXT BOOKS:**

1. Aryasri: Managerial Economics and Financial Analysis, TMH, 2012.
2. Vijay Kumar & Appa Rao Managerial Ecoeconomics & Financial Analysis, Cengage 2011.
3. J. V. Prabhakar Rao & P.V. Rao Managerial Ecoeconomics & Financial Analysis, Maruthi Publishers, 2011.

**REFERENCES:**

1. Ambrish Gupta, Financial Accounting for Management, Pearson Education, New Delhi.2012.
2. H. Craig Peterson & W. Cris Lewis, Managerial Economics, Pearson, 2012.
3. Lipsey & Chrystel, Economics, Oxford University Press, 2009
4. Domnick Salvatore: Managerial Economics In a Global Economy, Thomson, 2012.

5. Narayanaswamy: Financial Accounting—A Managerial Perspective, PHI, 2012.
6. S.N.Maheswari & S.K. Maheswari, Financial Accounting, Vikas, 2012.
7. Truet and Truet: Managerial Economics: Analysis, Problems and Cases, Wiley, 2012.
8. Dwivedi: Managerial Economics, Vikas, 2012.
9. Kasi Reddy Sraswathi, MEFA PHI Learning, 2012.
10. Shailaja & Usha : MEFA, University Press, 2012.

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****III Year B.Tech. CSE I-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>0</b>	<b>0</b>	<b>3</b>	<b>2</b>

**COMPUTER NETWORKS COMPILER DESIGN LAB**

1. Consider the following mini Language, a simple procedural high-level language, only operating on integer data, with a syntax looking vaguely like a simple C crossed with Pascal. The syntax of the language is defined by the following BNF grammar:

```

<program> ::= <block>
<block> ::= { <variabledefinition><slist> }
| { <slist> }
<variabledefinition> ::= int <vardeflist> ;
<vardeflist> ::= <vardec> | <vardec> , <vardeflist>
<vardec> ::= <identifier> | <identifier> [ <constant> ]
<slist> ::= <statement> | <statement> ; <slist>
<statement> ::= <assignment> | <ifstatement> | <whilestatement>
| <block> | <printstatement> | <empty>
<assignment> ::= <identifier> = <expression>
| <identifier> [ <expression> ] = <expression>
<ifstatement> ::= if <bexpression> then <slist> else <slist> endif
| if <bexpression> then <slist> endif
<whilestatement> ::= while <bexpression> do <slist> enddo
<printstatement> ::= print ( <expression> )
<expression> ::= <expression><addingop><term> | <term> | <addingop><term>
<bexpression> ::= <expression><relop><expression>
<relop> ::= < | <= | == | >= | > | !=
<addingop> ::= + | -
<term> ::= <term><multop><factor> | <factor>
<multop> ::= * | /
<factor> ::= <constant> | <identifier> | <identifier> [ <expression> ]
| ( <expression> )
<constant> ::= <digit> | <digit><constant>
<identifier> ::= <identifier><letterordigit> | <letter>
<letterordigit> ::= <letter> | <digit>
<letter> ::= a|b|c|d|e|f|g|h|i|j|k|l|m|n|o|p|q|r|s|t|u|v|w|x|y|z
<digit> ::= 0|1|2|3|4|5|6|7|8|9
<empty> has the obvious meaning

```

Comments (zero or more characters enclosed between the standard C/Java-style comment

brackets /\*...\*/) can be inserted. The language has rudimentary support for 1-dimensional arrays. The declaration `int a[3]` declares an array of three elements, referenced as `a[0]`, `a[1]` and `a[2]`. Note also that you should worry about the scoping of names.

A simple program written in this language is:

```

{ int a[3],t1,t2;
  t1=2;
  a[0]=1; a[1]=2; a[t1]=3;
  t2=-(a[2]+t1*6)/(a[2]-t1);
  if t2>5 then
    print(t2);
  else {

```



```

int t3;
t3=99;
t2=-25;
print(-t1+t2*t3); /* this is a comment
                  on 2 lines */
}
endif
}

```

1. Design a Lexical analyzer for the above language. The lexical analyzer should ignore redundant spaces, tabs and newlines. It should also ignore comments. Although the syntax specification states that identifiers can be arbitrarily long, you may restrict the length to some reasonable value.
2. Implement the lexical analyzer using JLex, flex or lex or other lexical analyzer generating tools.
3. Design Predictive parser for the given language
4. Design LALR bottom up parser for the above language.
5. Convert the BNF rules into Yacc form and write code to generate abstract syntax tree.
6. Write program to generate machine code from the abstract syntax tree generated by the parser. The following instruction set may be considered as target code.

The following is a simple register-based machine, supporting a total of 17 instructions. It has three distinct internal storage areas. The first is the set of 8 registers, used by the individual instructions as detailed below, the second is an area used for the storage of variables and the third is an area used for the storage of program. The instructions can be preceded by a label. This consists of an integer in the range 1 to 9999 and the label is followed by a colon to separate it from the rest of the instruction. The numerical label can be used as the argument to a jump instruction, as detailed below. In the description of the individual instructions below, instruction argument types are specified as follows:

**R**

specifies a register in the form R0, R1, R2, R3, R4, R5, R6 or R7 (or r0, r1, etc.).

**L**

specifies a numerical label (in the range 1 to 9999).

**V**

specifies a "variable location" (a variable number, or a variable location pointed to by a register - see below).

**A**

specifies a constant value, a variable location, a register or a variable location pointed to by a register (an indirect address). Constant values are specified as an integer value, optionally preceded by a minus sign, preceded by a # symbol. An indirect address is specified by an @ followed by a register. So, for example, an A-type argument could have the form 4 (variable number 4), #4 (the constant value 4), r4 (register 4) or @r4 (the contents of register 4 identifies the variable location to be accessed).

The instruction set is defined as follows:

**LOAD A,R**

loads the integer value specified by A into register R.

**STORE R,V**

stores the value in register R to variable V.

**OUT R**

outputs the value in register R.

**NEG R**

negates the value in register R.

**ADD A,R**

adds the value specified by A to register R, leaving the result in register R.

**SUB A,R**

subtracts the value specified by A from register R, leaving the result in register R.

**MUL A,R**

multiplies the value specified by A by register R, leaving the result in register R.

**DIV A,R**

divides register R by the value specified by A, leaving the result in register R.

**JMP L**

causes an unconditional jump to the instruction with the label L.

**JEQ R,L**

jumps to the instruction with the label L if the value in register R is zero.

**JNE R,L**

jumps to the instruction with the label L if the value in register R is not zero.

**JGE R,L**

jumps to the instruction with the label L if the value in register R is greater than or equal to zero.

**JGT R,L**

jumps to the instruction with the label L if the value in register R is greater than zero.

**JLE R,L**

jumps to the instruction with the label L if the value in register R is less than or equal to zero.

**JLT R,L**

jumps to the instruction with the label L if the value in register R is less than zero.

**NOP**

is an instruction with no effect. It can be tagged by a label.

**STOP**

stops execution of the machine. All programs should terminate by executing a STOP instruction.

**Computer Networks Lab**

1. Implement the data link layer framing methods such as character, character stuffing and bit stuffing.
2. Implement on a data set of characters the three CRC polynomials – CRC 12 , CRC 16 and CRC CCIP .
3. Implement Dijkstra 's algorithm to compute the Shortest path thru a graph.
4. Take an example subnet graph with weights indicating delay between nodes. Now obtain Routing table at each node using instance vector routing algorithm
5. Take an example subnet of hosts . Obtain broadcast tree for it.
6. Implement following with any simulation software.
  - i) Ethernet  
Objective: Study how the throughput of the network is affect by network load as well as size of the packets
  - ii) Switched LAN'S:

Objective: Study how the through put and collision of packets in a switched network are affected by the configuration of the network and types of switching devices that are used.

iii) Transmission Control Protocol:

Objective: Study TCP as end-to –end transmission protocol and analyze size of congestion window with different mechanisms

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****III Year B.Tech. CSE I-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>0</b>	<b>0</b>	<b>3</b>	<b>2</b>

**DATABASE MANAGEMENT SYSTEMS LAB****List of Experiments:-**

- 1) Concept design with E-R Model
- 2) Relational Model
- 3) Normalization
- 4) Practicing DDL commands
- 5) Practicing DML commands
- 6) Querying (using ANY, ALL, IN, Exists, NOT EXISTS, UNION, INTERSECT, Constraints etc.)
- 7) Queries using Aggregate functions, GROUP BY, HAVING and Creation and dropping of Views.
- 8) Triggers (Creation of insert trigger, delete trigger, update trigger. Practice triggers using the above database.)
- 9) Procedures
- 10) Usage of Cursors

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****III Year B.Tech. CSE I-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>0</b>	<b>0</b>	<b>3</b>	<b>2</b>

**JAVA PROGRAMMING LAB****Objectives:**

- To introduce java compiler and IDE platform
- To impart hand on experience with java programming

**Outcomes:**

The student is expected to have hands on experience with the following:

- Basics of java programming, multi-threaded programs and Exception handling
- The skills to apply OOP in Java programming in problem solving
- Ability to access data from a DB with Java programs
- Use of GUI components (Console and GUI based)

Use Eclipse or Netbean platform and acquaint with the various menus. Create a test project, add a test class and run it. See how you can use auto suggestions, auto fill. Try code formatter and code refactoring like renaming variables, methods and classes. Try debug step by step with a small program of about 10 to 15 lines which contains at least one if else condition and a for loop.

- 1) Write a Java program that creates a user interface to perform integer divisions. The user enters two numbers in the text fields, Num1 and Num2. The division of Num1 and Num2 is displayed in the Result field when the Divide button is clicked. If Num1 or Num2 were not an integer, the program would throw a Number Format Exception. If Num2 were Zero, the program would throw an Arithmetic Exception. Display the exception in a message dialog box.[Use JOption Pane –Input dialog ,Message dialog]
- 2) Write a Java program to create an abstract class named Shape that contains two integers and an empty method named printArea(). Provide three classes named Rectangle, Triangle and Circle such that each one of the classes extends the class Shape. Each one of the classes contains only the method printArea () that prints the area of the given shape.
- 3) Write a Java program that implements a multi-thread application that has three threads. First thread generates random integer every 1 second and if the value is even, second thread computes the square of the number and prints. If the value is odd, the third thread will print the value of cube of the number.
- 4) Write a Java program that connects to a database using JDBC and does add, delete, modify and retrieve operations.
- 5) Write a Java program that simulates a traffic light. The program lets the user select one of three lights: red, yellow, or green with radio buttons. On selecting a button, an appropriate message with “Stop” or “Ready” or “Go” should appear above the buttons in selected color. Initially, there is no message shown.
- 6) Write a Java program that works as a simple calculator. Use a grid layout to arrange buttons for the digits and for the +, -,\*, % operations. Add a text field to display the result.Handle any possible exceptions like divided by zero

- 7) a) Develop an applet in Java that displays a simple message.  
b) Develop an applet in Java that receives an integer in one text field, and computes its factorial Value and returns it in another text field, when the button named “Compute” is clicked.
- 8) Suppose that a table named Table.txt is stored in a text file. The first line in the file is the header, and the remaining lines correspond to rows in the table. The elements are separated by commas. Write a java program to display the table using Labels in Grid Layout.
- 9) Write a Java program that connects to a database using JDBC and does add, delete, modify and retrieve operations.
- 10) Write a Java program that loads names and phone numbers from a text file where the data is organized as one line per record and each field in a record are separated by a tab (\t). It takes a name or phone number as input and prints the corresponding other value from the hash table (hint: use hash tables).
- 11) Implement the above program with database instead of a text file.
- 12) Write a java program that prints the meta-data of a given table

**Text Books:**

1. Java Fundamentals – A comprehensive Introduction, Herbert Schildt and Dale Skrien, TMH.

**References:**

1. Java for Programmers, P.J.Deitel and H.M.Deitel, Pearson education (OR) Java: How to Program P.J.Deitel and H.M.Deitel, PHI.
2. Object Oriented Programming through Java, P.Radha Krishna, Universities Press.
3. Thinking in Java, Bruce Eckel, Pearson Education
4. Programming in Java, S.Malhotra and S.Choudhary, Oxford Univ. Press.

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****III Year B.Tech. CSE II-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>4</b>	<b>1</b>	<b>0</b>	<b>4</b>

**DATA WAREHOUSING AND DATAMINING****UNIT-I****DATA MINING**

Data-Types of Data-, Data Mining Functionalities- Interestingness Patterns-Classification of Data Mining systems- Data mining Task primitives -Integration of Data mining system with a Data warehouse-Major issues in Data Mining-Data Preprocessing.

**UNIT-II****DATA WAREHOUSE AND BUSINESS ANALYSIS**

Data Warehouse-Data Warehouse Architecture- Multidimensional Data Model-Data cube and OLAP Technology-Data Warehouse Implementation -DBMS schemas for Decision support - Efficient methods for Data cube computation.

**UNIT-III****ASSOCIATION RULE MINING AND CLASSIFICATION**

Mining Frequent Patterns-Associations and correlations- Mining Methods- Mining Various kinds of Association Rules- Correlation Analysis- Constraint based Association mining.- Classification and Prediction- Basic concepts-Decision tree induction-Bayesian classification, Rule-based classification - classification by Back propagation,-Support vector machines.-Associative Classification, Lazy learners-Other classification methods – Prediction.

**UNIT-IV****CLUSTERING AND APPLICATIONS**

Cluster analysis-Types of Data in Cluster Analysis-Categorization of Major Clustering Methods- Partitioning Methods,-Hierarchical Methods- Density-Based Methods,-Grid-Based Methods,-Model-Based Clustering Methods- Clustering high dimensional data-Constraint-Based cluster analysis-Outlier Analysis

**UNIT V****MINING DATA STREAMS, TIME-SERIES AND SEQUENCE DATA**

Basic concepts- Mining data streams-Mining Time-series data--Mining sequence patterns in Transactional databases-.Mining Object- Spatial- Multimedia-Text and Web data- Spatial Data mining- Multimedia Data mining--Text Mining- Mining the World Wide Web.

**Text Books:**

1. Data Mining – Concepts and Techniques - JIAWEI HAN & MICHELINE KAMBER , Elsevier.
2. Data Warehousing, Data Mining & OLAP- Alex Berson and Stephen J.Smith- Tata McGraw-Hill Edition ,Tenth reprint 2007

**References:**

1. Data Mining Techniques – ARUN K PUJARI, University Press
2. Building the Data Warehouse- W. H. Inmon, Wiley Dreamtech India Pvt. Ltd..
3. Data Warehousing Fundamentals – PAULRAJ PONNAIAH WILEY STUDENT EDITION

4. The Data Warehouse Life cycle Tool kit – RALPH KIMBALL WILEY STUDENT EDITION
5. Data Mining Introductory and advanced topics –MARGARET H DUNHAM, PEA.
6. DATA MINING – TAN, VIPIN KUMAR, PEA



**JNTUH COLLEGE OF ENGINEERING HYDERABAD****III Year B.Tech. CSE II-Sem**

L	T	P	C
4	1	0	4

**WEB TECHNOLOGIES****Objectives:**

- To introduce PHP language for server side scripting
- To introduce XML and processing of XML Data with Java
- To introduce Server side programming with Java Servlets and JSP
- To introduce Client side scripting with Javascript and AJAX.

**Outcomes:**

The Student is expected to

- gain knowledge of client side scripting, validation of forms and AJAX programming
- have understanding of server side scripting with PHP language
- have understanding of what is XML and how to parse and use XML Data with Java
- To introduce Server side programming with Java Servlets and JSP

**UNIT- I**

**Introduction to PHP:** Declaring variables, data types, arrays, strings, operators, expressions, control structures, functions, Reading data from web form controls like text boxes, radio buttons, lists etc., Handling File Uploads, Connecting to database (MySQL as reference), executing simple queries, handling results, Handling sessions and cookies

**File Handling in PHP:** File operations like opening, closing, reading, writing, appending, deleting etc. on text and binary files, listing directories

**UNIT- II**

**XML:** Introduction to XML, Defining XML tags, their attributes and values, Document Type Definition, XML Schemas, Document Object Model, XHTML

**Parsing XML Data** - DOM and SAX Parsers in java.

**UNIT- III**

**Introduction to Servlets:** Common Gateway Interface (CGI), Lifecycle of a Servlet, deploying a servlet, The Servlet API, Reading Servlet parameters, Reading Initialization parameters, Handling Http Request & Responses, Using Cookies and Sessions, connecting to a database using JDBC.

**UNIT- IV**

**Introduction to JSP:** The Anatomy of a JSP Page, JSP Processing, Declarations, Directives, Expressions, Code Snippets, implicit objects, Using Beans in JSP Pages, Using Cookies and session for session tracking, connecting to database in JSP.

**UNIT- V**

**Client side Scripting:** Introduction to Javascript: Javascript language - declaring variables, scope of variables, functions, event handlers (onclick, onsubmit etc.), Document Object Model, Form validation.

Simple AJAX application.

**Text Books:**

1. Web Technologies, Uttam K Roy, Oxford University Press
2. The Complete Reference PHP – Steven Holzner, Tata McGraw-Hill

**References:**

1. Web Programming, building internet applications, Chris Bates 2<sup>nd</sup> edition, Wiley Dreamtech
2. Java Server Pages –Hans Bergsten, SPD O'Reilly
3. Java Script, D.Flanagan, O'Reilly,SPD.
4. Beginning Web Programming-Jon Duckett WROX.
5. Programming world wide web, R.W.Sebesta, Fourth Edition, Pearson.
6. Internet and World Wide Web – How to program, Dietel and Nieto, Pearson.

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****III Year B.Tech. CSE II-Sem**

L	T	P	C
4	0	0	4

**OPEN ELECTIVE-II**  
**ESTIMATION, QUANTITY SURVEY & VALUATION**

**UNIT – I**

General items of work in Building – Standard Units Principles of working out quantities for detailed and abstract estimates – Approximate method of Estimating.

**UNIT – II**

Detailed Estimates of Buildings - Reinforcement bar bending and bar requirement schedules

**UNIT – III**

Earthwork for roads and canals.

**UNIT – IV**

Rate Analysis – Working out data for various items of work over head and contingent charges.

**UNIT-V**

Contracts – Types of contracts – Contract Documents – Conditions of contract, Valuation - Standard specifications for different items of building construction.

**NOTE : NUMBER OF EXERCISES PROPOSED :**

1. Three in flat Roof & one in Sloped Roof
2. Exercises on Data – three Nos.

**Text Books**

1. Estimating and Costing by B.N. Dutta, UBS publishers, 2000.
2. Estimating and Costing by G.S. Birdie

**Reference books :**

1. Standard Schedule of rates and standard data book by public works department.
2. I. S. 1200 ( Parts I to XXV – 1974/ method of measurement of building and Civil Engineering works – B.I.S.)
3. Estimation, Costing and Specifications by M. Chakraborti; Laxmi publications.

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****III Year B.Tech. CSE II-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>4</b>	<b>0</b>	<b>0</b>	<b>4</b>

**OPEN ELECTIVE-II**  
**ENERGY STORAGE SYSTEMS**

**Unit-1 Electrical Energy Storage Technologies**

Characteristics of electricity, Electricity and the roles of EES, High generation cost during peak-demand periods, Need for continuous and flexible supply, Long distance between generation and consumption, Congestion in power grids, Transmission by cable.

**Unit-2 Needs for Electrical Energy Storage**

Emerging needs for EES, More renewable energy, less fossil fuel, Smart Grid uses, The roles of electrical energy storage technologies, The roles from the viewpoint of a utility, The roles from the viewpoint of consumers, The roles from the viewpoint of generators of renewable energy.

**Unit-3 Features of Energy Storage Systems**

Classification of EES systems, Mechanical storage systems, Pumped hydro storage (PHS), Compressed air energy storage (CAES), Flywheel energy storage (FES), Electrochemical storage systems, Secondary batteries, Flow batteries, Chemical energy storage, Hydrogen (H<sub>2</sub>), Synthetic natural gas (SNG).

**Unit-4 Types of Electrical Energy Storage systems**

Electrical storage systems, Double-layer capacitors (DLC), Superconducting magnetic energy storage (SMES), Thermal storage systems, Standards for EES, Technical comparison of EES technologies.

**Unit-5 Applications**

Present status of applications, Utility use (conventional power generation, grid operation & service), Consumer use (uninterruptable power supply for large consumers), New trends in applications, Renewable energy generation, Smart Grid, Smart Micro grid, Smart House, Electric vehicles, Management and control hierarchy of storage systems, Internal configuration of battery storage systems, External connection of EES systems, Aggregating EES systems and distributed generation (Virtual Power Plant), Battery SCADA— aggregation of many dispersed batteries.

**TEXT BOOK:**

1. "Energy Storage Benefits and Market Analysis" by James M. Eyer, Joseph J. Iannucci and Garth P. Corey.
2. The Electrical Energy Storage by IEC Market Strategy Board.

**REFERENCE BOOKS:**

1. Jim Eyer, Garth Corey: *Energy Storage for the Electricity Grid: Benefits and Market Potential Assessment Guide*, Report, Sandia National Laboratories, Feb 2010.

\*\*\*

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****III Year B.Tech. CSE II-Sem**

L	T	P	C
4	0	0	4

**OPEN ELECTIVE-II  
MECHATRONICS****UNIT – I**

**INTRODUCTION:** Definition – Trends - Control Methods: Standalone , PC Based ( Real Time Operating Systems, Graphical User Interface , Simulation ) - Applications: identification of sensors and actuators in Washing machine, Automatic Camera, Engine Management, SPM, Robot, CNC, FMS, CIM.

**SIGNAL CONDITIONING :** Introduction – Hardware - Digital I/O , Analog input – ADC , resolution, Filtering Noise using passive components – Registers, capacitors - Amplifying signals using OP amps –Software - Digital Signal Processing – Low pass , high pass , notch filtering

**UNIT – II****PRECISION MECHANICAL SYSTEMS :**

Modern CNC Machines – Design aspects in machine structures, guideways, feed drives, spindle and spindle bearings, measuring systems, control software and operator interface, gauging and tool monitoring.

**Note: (text book: Mechatronics HMT – chapter 5)**

**ELECTRONIC INTERFACE SUBSYSTEMS :** TTL, CMOS interfacing - Sensor interfacing – Actuator interfacing – solenoids , motors Isolation schemes- opto coupling, buffer IC's - Protection schemes – circuit breakers , over current sensing , resetable fuses , thermal dissipation - Power Supply - Bipolar transistors / mosfets

**UNIT – III**

**ELECTROMECHANICAL DRIVES :** Relays and Solenoids - Stepper Motors - DC brushed motors – DC brushless motors - DC servo motors - 4-quadrant servo drives , PWM's - Pulse Width Modulation – Variable Frequency Drives, Vector Drives - Drive System load calculation.

**MICROCONTROLLERS OVERVIEW :** 8051 Microcontroller , micro processor structure – Digital Interfacing - Analog Interfacing - Digital to Analog Convertors - Analog to Digital Convertors - Applications. Programming –Assembly, C ( LED Blinking , Voltage measurement using ADC).

**UNIT – IV**

**PROGRAMMABLE LOGIC CONTROLLERS :** Basic Structure - Programming : Ladder diagram -Timers, Internal Relays and Counters - Shift Registers - Master and Jump Controls - Data Handling -Analog input / output - PLC Selection - Application.

**UNIT – V**

**PROGRAMMABLE MOTION CONTROLLERS :** Introduction - System Transfer Function – Laplace transform and its application in analysing differential equation of a control system - Feedback Devices : Position , Velocity Sensors - Optical Incremental encoders - Proximity Sensors : Inductive , Capacitive , Infrared - Continuous and discrete processes - Control System Performance & tuning - Digital Controllers - P , PI , PID Control

- Control modes – Position , Velocity and Torque - Velocity Profiles – Trapezoidal- S. Curve  
- Electronic Gearing - Controlled Velocity Profile - Multi axis Interpolation , PTP , Linear , Circular - Core functionalities – Home , Record position , GOTO Position - Applications : SPM, Robotics.

**TEXT BOOKS :**

1. Mechatronics Electronics Control Systems in Mechanical and Electrical Engineering by W Bolton, Pearson Education Press, 3rd edition, 2005.
2. Mechatronics/M.D.Singh/J.G.Joshi/PHI.

**REFERENCE:**

1. “Designing Intelligent Machines”. open University, London.
2. Michel B. Histan and David G. Alciatore,”
3. Introduction to Mechatronics and Measurement systems, “Tata MC Graw hill
4. I. C.W. Desi ha, “Control sensors and actuators,” Prentice Hall.
5. Mechatronics Source Book by Newton C Braga, Thomson Publications, Chennai.
6. Mechatronics – N. Shanmugam / Anuradha Agencies Publisers.
7. Mechatronics System Design / Devdas shetty/Richard/Thomson.

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****III Year B.Tech. CSE II-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>4</b>	<b>0</b>	<b>0</b>	<b>4</b>

**OPEN ELECTIVE-II**  
**PRINCIPLES OF COMMUNICATION SYSTEMS**

**Unit 1:**

**Introduction:** Communication Systems and types, modulation and multiplexing, Electromagnetic spectrum, Gain, Attenuation and decibels.

**Unit 2:**

**Simple description on Modulation:** Analog Modulation-AM, FM, Pulse Modulation-PAM, PWM, PCM, Digital Modulation Techniques-ASK, FSK, PSK, QPSK modulation and demodulation schemes.

**Unit 3:**

**Telecommunication Systems:** Telephones Telephone system, Paging systems, Telephony.

**Networking and Local Area Networks:** Network fundamentals, LAN hardware, Ethernet LANs, Token Ring LAN.

**Unit 4:**

**Satellite Communication:** Satellite Orbits, satellite communication systems, satellite subsystems, Ground Stations Satellite Applications, Global Positioning systems.

**Optical Communication:** Optical Principles, Optical Communication Systems, Fiber –Optic Cables, Optical Transmitters & Receivers, Wavelength Division Multiplexing.

**Unit 5:**

**Multiple Access Techniques:** FDMA, TDMA, CDMA, Packet Radio techniques-ALOHA, slotted ALOHA.

**Cellular and Mobile Communications:** Cellular telephone systems, AMPS, GSM, CDMA, WCDMA.

**Wireless Technologies:** Wireless LAN, PANs and Bluetooth, ZigBee and Mesh Wireless networks, Wimax and MANs, Infrared wireless, RFID communication, UWB.

**Text Books:**

1. Principles of Electronic Communication Systems, Louis E. Frenzel, 3e, McGraw Hill publications, 2008.
2. Wayne Tomasi, Introduction to data communications and networking, Pearson Education, 2005.

**Reference Books:**

1. Tarmo Anttalainen, Introduction to Telecommunications Network Engineering, Artech House Telecommunications Library.
2. Theodore Rappaport, Wireless Communications-Principles and practice, Printice Hall, 2002.
3. Roger L. Freeman, Fundamentals of Telecommunications, 2e, Wiley publications.
4. Kennady, Davis, Electronic Communications systems, 4e, TMH, 1999.

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****III Year B.Tech. CSE II-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>4</b>	<b>0</b>	<b>0</b>	<b>4</b>

**OPEN ELECTIVE-II**  
**E-COMMERCE**

**UNIT-I**

Electronic Commerce-Frame work, anatomy of E-Commerce applications, E-Commerce Consumer applications, E-Commerce organization applications.

**UNIT-II**

Consumer Oriented Electronic commerce - Mercantile Process models, Electronic payment systems - Digital Token-Based, Smart Cards, Credit Cards, Risks in Electronic Payment systems.

**UNIT-III**

Inter Organizational Commerce - EDI, EDI Implementation, Value added networks, Intra Organizational Commerce - work Flow, Automation Customization and internal Commerce, Supply chain Management.

**UNIT-IV**

Corporate Digital Library - Document Library, digital Document types, corporate Data Warehouses. Advertising and Marketing - Information based marketing, Advertising on Internet, on-line marketing process, market research.

**UNIT-V**

Consumer Search and Resource Discovery - Information search and Retrieval, Commerce Catalogues, Information Filtering, Multimedia - key multimedia concepts, Digital Video and electronic Commerce, Desktop video processings, Desktop video conferencing.

**TEXT BOOKS:**

1. Frontiers of electronic commerce – Kalakata, Whinston, Pearson.

**REFERENCE BOOKS:**

1. E-Commerce fundamentals and applications Hendry Chan, Raymond Lee, Tharam Dillon, Elizabeth Chang, John Wiley.
2. E-Commerce, S.Jaiswal – Galgotia.
3. E-Commerce, Efrain Turbon, Jae Lee, David King, H.Michael Chang.
4. Electronic Commerce – Gary P.Schneider – Thomson.
5. E-Commerce – Business, Technology, Society, Kenneth C.Taudon, Carol Guyerico Traver.



**JNTUH COLLEGE OF ENGINEERING HYDERABAD****III Year B.Tech. CSE II-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>4</b>	<b>0</b>	<b>0</b>	<b>4</b>

**OPEN ELECTIVE-II  
COMPUTER GRAPHICS****Objectives:**

- To make students understand about fundamentals of Graphics to enable them to design animated scenes for virtual object creations.
- To make the student present the content graphically.

**Outcomes:**

- Students can animate scenes entertainment.
- Will be able work in computer aided design for content presentation..
- Better analogy data with pictorial representation.

**UNIT-I:**

**Introduction:** Application areas of Computer Graphics, overview of graphics systems, video-display devices, raster-scan systems, random scan systems, graphics monitors and work stations and input devices

**Output primitives:** Points and lines, line drawing algorithms, mid-point circle and ellipse algorithms. Filled area primitives: Scan line polygon fill algorithm, boundary-fill and flood-fill algorithms

**UNIT-II:**

**2-D geometrical transforms:** Translation, scaling, rotation, reflection and shear transformations, matrix representations and homogeneous coordinates, composite transforms, transformations between coordinate systems

**2-D viewing :** The viewing pipeline, viewing coordinate reference frame, window to view-port coordinate transformation, viewing functions, Cohen-Sutherland and Cyrus-beck line clipping algorithms, Sutherland –Hodgeman polygon clipping algorithm

**UNIT-III:**

**3-D object representation :** Polygon surfaces, quadric surfaces, spline representation, Hermite curve, Bezier curve and B-Spline curves, Bezier and B-Spline surfaces. Basic illumination models, polygon rendering methods.

**3-D Geometric transformations:** Translation, rotation, scaling, reflection and shear transformations, composite transformations. **3-D viewing :** Viewing pipeline, viewing coordinates, view volume and general projection transforms and clipping.

**UNIT-IV:**

**Visible surface detection methods:** Classification, back-face detection, depth-buffer, scan-line, depth sorting, BSP-tree methods, area sub-division and octree methods

**UNIT-V:**

**Computer animation:** Design of animation sequence, general computer animation functions, raster animation, computer animation languages, key frame systems, motion specifications

**Text Books:**

1. “Computer Graphics *C version*”, Donald Hearn and M. Pauline Baker,

Pearson Education

2. “Computer Graphics Principles & practice”, second edition in C, Foley, VanDam, Feiner and Hughes, Pearson Education.

**References:**

1. Computer Graphics”, second Edition, Donald Hearn and M.Pauline Baker, PHI/Pearson Education.
2. Computer Graphics Second edition”, Zhigand xiang, Roy Plastock, Schaum’s outlines, Tata Mc-Graw hill edition.
3. rocedural elements for Computer Graphics, David F Rogers, Tata Mc Graw hill, 2<sup>nd</sup> edition.
4. Principles of Interactive Computer Graphics”, Neuman and Sproul, TMH.
5. Principles of Computer Graphics, Shalini Govil, Pai, 2005, Springer.
6. Computer Graphics, Steven Harrington, TMH

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****III Year B.Tech. CSE II-Sem**

L	T	P	C
4	0	0	4

**OPEN ELECTIVE-II**  
**DATABASE MANAGEMENT SYSTEMS**

**Objectives:**

- To understand the basic concepts and the applications of database systems.
- To master the basics of SQL and construct queries using SQL.
- To understand the relational database design principles.
- To become familiar with the basic issues of transaction processing and concurrency control.
- To become familiar with database storage structures and access techniques.

**UNIT I :**

**Data base System Applications:** data base System VS file System, View of Data, Data Abstraction ,Instances and Schemas, data Models ,the ER Model, Relational Model, Other Models, Database Languages, DDL, DML, database Access for applications Programs ,data base Users and Administrator ,Transaction Management, data base System Structure, Storage Manager, the Query Processor. History of Data base Systems. Data base design and ER diagrams, Beyond ER Design Entities, Attributes and Entity sets, Relationships and Relationship sets, Additional features of ER Model ,Concept Design with the ER Model, Conceptual Design for Large enterprises.

**UNIT II:**

**Introduction to the Relational Model:** Integrity Constraint Over relations ,Enforcing Integrity constraints , Querying relational data , Logical data base Design ,Introduction to Views, Destroying /altering Tables and Views. Form of Basic SQL Query, Examples of Basic SQL Queries, Introduction to Nested Queries ,Correlated Nested Queries Set, Comparison Operators, Aggregative Operators, NULL values ,Comparison using Null values ,Logical connectivity's, AND, OR and NOT, Impact on SQL Constructs Outer Joins, Disallowing NULL values ,Complex Integrity Constraints in SQL Triggers and Active Data bases, Oracle, SQL Server,DB2.

**UNIT III:**

**Relational Algebra :**Selection and projection set operations, renaming ,Joins ,Division, Examples of Algebra overviews, Relational calculus, Tuple relational Calculus, Domain relational calculus ,Expressive Power of Algebra and calculus.

**Schema refinement:** Problems Caused by redundancy, Decompositions, Problem related to decomposition, reasoning about FDS,FIRST, SECOND, THIRD Normal forms ,BCNF, Lossless join Decomposition ,Dependency preserving Decomposition, Schema refinement in Data base Design ,Multi valued Dependencies , FORTH Normal Form, FIFTH Normal Form.

**UNIT IV:**

Transaction Concept, Transaction State, Implementation of Atomicity and Durability, Concurrent Executions , Serializability, Recoverability ,Implementation of Isolation ,Testing for serializability,Lock Based Protocols , Timestamp Based Protocols, Validation- Based Protocols ,Multiple Granularity.

Recovery and Atomicity ,Log-Based Recovery ,Recovery with Concurrent Transactions, Buffer Management, Failure with loss of nonvolatile storage, Advance Recovery systems, Remote Backup systems.

**UNIT V:**

Data on External Storage, File Organization and Indexing, Cluster Indexes, Primary and Secondary Indexes, Index data Structures ,Hash Based Indexing ,Tree base Indexing ,Comparison of File Organizations ,Indexes and Performance Tuning, Intuitions for tree Indexes, Indexed Sequential Access Methods (ISAM), B+ Trees: A Dynamic Index Structure.

**Text Books :**

1. Data base Management Systems, Raghurama Krishnan, Johannes Gehrke, TATA McGrawHill 3rd Edition
2. Data base System Concepts, Silberschatz, Korth, McGraw hill, V edition.

**References :**

1. Data base Systems design, Implementation, and Management, Peter Rob & Carlos Coronel 7th Edition.
2. Fundamentals of Database Systems, Elmasri Navrate Pearson Education
3. Introduction to Database Systems, C.J.Date Pearson Education
4. Oracle for Professionals, The X Team, S.Shah and V.Shah, SPD.
5. Database Systems Using Oracle: A Simplified guide to SQL and PL/SQL, Shah, PHI.
6. Fundamentals of Database Management Systems, M.L.Gillenson, Wiley Student Edition.

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****III Year B.Tech. CSE II-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>4</b>	<b>0</b>	<b>0</b>	<b>4</b>

**OPEN ELECTIVE-II  
NANOMATERIALS****Unit – 1****Introduction**

What is Nano – Why Nano – Properties at Nano Scales, Advantages & Disadvantages, Applications in comparison with bulk materials (Nanostructure, nanowires, nanotubes, nanocomposites)

**Nano Particles**

Introduction – Synthesis procedures – wet chemical approach & physical vapor synthesis approach – size effect & shape change and their properties – examples of systems involved – characterization techniques – properties & their applications

**Unit – II****Nano Wires**

Introduction – various synthesis procedures (template assisted method, VLS method and other synthesis methods) – properties of nanowires – characterization procedures & principles involved. Applications of Nanowires.

**Nanotubes**

Introduction – Different systems involved in nanotubes – single walled, multi-walled, Carbon based, metal incorporated tubes. Synthesis procedures (Solid & gaseous carbon source based production techniques) Growth mechanism of carbon nanotubes – properties of carbon nanotubes – characterization – applications.

**Unit – III****Nano Composites**

Introduction, Synthesis procedures, various systems (metal-polymer, metal-ceramics and polymer-Ceramics). Characterization – procedures – Applications.

**Unit – IV****Micro/Nano Fabrication Techniques**

Introduction, Basic fabrication techniques (lithography, thin film deposition and doping) MEMS fabrication techniques, Nano fabrication techniques (E-Beam nano-imprint fabrication, Epitaxy and strain engineering. Scanned probe techniques)

**Unit – V****Materials of Nano Technology**

Introduction – Si-based Materials - Ge-based materials - Ferro electric materials – Polymer materials - GaAs & InP (III – V) Group materials, Nano tribology and materials - characterization using Scanning Probe Microscope, AFM, FFM.

**Nano Biomaterials**

Introduction, Biocompatibility; anti bacterial activity – principles involved – Applications.

**TEXT / Reference BOOKS**

1. Nano Materials: A. K. Bandyopadhyay, New age Publications
2. Nano Essentials: T. Pradeep, TMH
3. Springer Handbook of Nanotechnology
4. The Guest for new materials Auther S. T. Lakshmi Kumar, Published by Vigyan Prasar.

5. Nano – The Essentials: C – Pradeep (IITC Professor), McGraw Hill
6. Nano Materials Synthesis, Properties and applications, 1996, Edlstein and Cammarate

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****III Year B.Tech. CSE II-Sem**

L	T	P	C
4	0	0	4

**OPEN ELECTIVE-II**  
**INTELLECTUAL PROPERTY RIGHTS**

**Unit-I**

Introduction to Intellectual property law Basics-types of Intellectual property-Agencies responsible for intellectual property registration, internal organisations, and treaties the increasing importance of intellectual property rights. Foundation of trade mark law purpose types and function of trade mark A acquisition of trademarks rights. Selecting and evaluating the application drawing mark a mark .preparing the application drawing of marks.

**Unit-II**

Interparty proceeding, infringement, and dilution, inter parties proceedings infringement of trademarks dilution of trademarks related trade mark claims. New development in trademark law the internet protecting a domain names hyper linking and the first amendment other cyberspace trade mark issues. Applications in the United States based on foreign applications and registration.

**Unit-III**

Foundations of copyright law common law right and right under the 1976 copyright Act the united states copyright office-the subject matter of copyright, originality of material fixation of material work of authorship exclusion from copyright protection case study and activity. The right afforded by copyright law right of reproduction right to prepare derivative works copyright ownership, transfer and duration.

**Unit-IV**

Introduction foundations of patent law rights under federal law United States patent and trademark office design patents plant patents double patenting the orphan drug Act. Patent ownership and transfer sole and joint inventor's disputes over inventor ship. New developments and international patent law

**Unit-V**

*The law of trade secrets unfair competition determination of trade secret status liability for misappropriation of trade secrets employer-employee relationships protection for submissions defences to trade secret misappropriation remedies for misappropriation trade secret litigation trade secret protection programs. Intellectual property audits and due diligence reviews.*

**TEXT BOOKS:**

1. Deborah E. Bo choux : *intellectual property*, cengage learning , 2012.

**REFERENCES:**

1. P.Narayana: Intellectual property Law 3<sup>rd</sup> Edition. Eastern Law House 2001-2002.
2. Dr S.R.Myneni: law of intellectual property 2<sup>nd</sup> edition, Asian law house 2003

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****III Year B.Tech. CSE II-Sem**

L	T	P	C
4	0	0	4

**OPEN ELECTIVE-II  
ENTREPRENEURSHIP**

**Aim:** The aim of this subject is to inspire students to become entrepreneurs so that they will emerge as job providers rather than job seekers.

**Learning Outcome:** By the end of this course the students should be able to understand the mindset of the entrepreneurs, identify ventures for launching, develop an idea on the legal framework and also understand strategic perspectives in entrepreneurship.

1. Understanding Entrepreneurial Mindset- The revolution impact of entrepreneurship- The evolution of entrepreneurship- Approaches to entrepreneurship- Process approach- Twenty first century trends in entrepreneurship.  
Case1: Ready, Aim, Fire Fire (B. Janakiram, M.Rizwana, page 212),  
Case2: Henry Ford, (B. Janakiram, M. Rizwana, page 214)  
Case 3: From candle seller to CEO (Arya Kumar P.No. 48)
2. The individual entrepreneurial mind-set and Personality- The entrepreneurial journey- Stress and the entrepreneur- the entrepreneurial ego- Entrepreneurial motivations. Corporate Entrepreneurial Mindset- the nature of corporate entrepreneur- conceptualization of corporate entrepreneurship Strategy-sustaining corporate entrepreneurship.  
Case : Globalizing Local Talent, (B. Janakiram, M. Rizwana, page 228).
3. Launching Entrepreneurial Ventures- opportunities identification- entrepreneurial Imagination and Creativity- the nature of the creativity process-Innovation and entrepreneurship. Methods to initiate Ventures- Creating new ventures-Acquiring an Established entrepreneurial venture- Franchising-hybrid- disadvantage of Franchising.  
Case 1: Water, Water everywhere: but not a drop to drink, (Richard Blundel , Page 48).  
Case 2: Critical Incident, Mark Robinson: Strategy Mapping Business (Richard Blundel, Page 48).  
Case 3: Pets.com (Arya Kumar P.No. 88)  
Case 4: creativity in start-ups (Arya Kumar P.No. 166)  
Case 5: Opportunity – Earthmoving Industry (Arya Kumar P.No. 211)
4. Legal challenges of Entrepreneurship-Intellectual property protection-Patents, Copyrights-Trade marks and Trade secrets-Avoiding trademark pitfalls. Formulation of the entrepreneurial Plan- The challenges of new venture start-ups, Poor financial Understanding-Critical factors for new venture development-The Evaluation process- Feasibility criteria approach.  
Case 1: Victoria, Tomlinson; Network. (Richard Blundel, Page 99).  
Case 2: Tim Lockett, Knowing your Customers & Suppliers (Richard Blundel Page128).  
Case 3: Google (Arya Kumar P.No. 248)  
Case 4: Tata Motors – Nano (Arya Kumar P.No. 279)
5. Strategic perspectives in entrepreneurship- Strategic planning-Strategic actions- strategic positioning-Business stabilization- Building the adaptive firms-Understanding the growth stage-Unique managerial concern of growing ventures.  
Case 1: To Lease or Not: A Cash flow Question (David H.Holt, Page 452).  
Case 2:- Public Sector - address seed capital (David H.Holt, Page 453).



**READING****Text Book :**

1. D F Kuratko and T V Rao “Entrepreneurship- A South-Asian Perspective “Cengage Learning, 2012.

**Cases:**

1. Arya Kumar “**Entrepreneurship- creating and leading an entrepreneurial organization ” Pearson 2012.**
2. Richard Blundel” Exploring Entrepreneurship Practices and Perspectives,Oxford,2011.
3. David H Holt” Entrepreneurship:New Venture Creation” PHI,2013.

**Journal :**

1. **The Journal of Entrepreneurship**, Entrepreneurship Development Institute of India, Ahmedabad,
2. **Journal of Human Values** : IIM Calcutta.

**References:**

1. Vasant Desai “Small Scale industries and entrepreneurship” Himalaya publishing 2012.
2. Rajeev Roy “Entrepreneurship” 2e, Oxford, 2012.
3. B.Janakiram and M.Rizwana” Entrepreneurship Development :Text & Cases, Excel Books,2011.
4. Robert Hisrich et al “Entrepreneurship” 6<sup>th</sup> e, TMH, 2012.

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****III Year B.Tech. CSE II-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>4</b>	<b>0</b>	<b>0</b>	<b>4</b>

**Department Elective - I**  
**ARTIFICIAL INTELLIGENCE**

**Objectives:**

- To learn the difference between optimal reasoning vs human like reasoning
- To understand the notions of state space representation, exhaustive search, heuristic search along with the time and space complexities
- To learn different knowledge representation techniques
- To understand the applications of AI: namely Game Playing, Theorem Proving, Expert Systems, Machine Learning and Natural Language Processing

**Outcomes:**

- Possess the ability to formulate an efficient problem space for a problem expressed in English.
- Possess the ability to select a search algorithm for a problem and characterize its time and space complexities.
- Possess the skill for representing knowledge using the appropriate technique
- Possess the ability to apply AI techniques to solve problems of Game Playing, Expert Systems, Machine Learning and Natural Language Processing

**UNIT I****Introduction:**

AI problems, AI Technique, defining problem as a static space search production systems, problem characteristics, production system characteristics.

**Heuristic search Techniques:**

Generate – and – test, Hill climbing, best – first search, problem reduction, constraint satisfaction, means-ends analysis.

**UNIT II****Knowledge representation:**

Issues, predicate logic, resolution, representing Knowledge using rules, Forward versus Backward reasoning, Matching, control Knowledge, weak slot – and – filler structures, semantic nets, frames, strong slot – and – filler structures, conceptual dependency, scripts.

**UNIT III****Reasoning Techniques:**

Nomonotonic reasoning, Augmenting a problem solver, implementation of depth first search and Breadth first search, statistical reasoning, probability and Bayes theorem, certainty factors and rule-based systems, Bayesian networks.

**UNIT IV****Game Playing:**

Minimax search, alpha – beta cutoffs, planning system, Goal stack planning, hierarchical planning, understanding, understanding as constraint satisfaction, waltz algorithm, natural language processing, syntactic processing, Augmented transition Networks, Semantic analysis, case grammars.

**UNIT V****Learning:**

Rote learning, learning by taking advice, learning in problem solving, learning from examples, Winston's learning program, Decision trees, perception, vision, speech recognition, Navigation, manipulation, Robot architectures, Expert systems, shell, explanation, knowledge acquisition.

**Text Books:**

1. Artificial Intelligence" 2<sup>nd</sup> Edn. , E.Rich and K.Knight (TMH)

**References:**

1. Artificial Intelligence A modern Approach, second edition, Stuart Russell, peter Norrig , PHI/ Pearson Education.
2. Artificial Intelligence, and Expert systems – Patterson PHI

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****III Year B.Tech. CSE II-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>4</b>	<b>0</b>	<b>0</b>	<b>4</b>

**Department Elective -I**  
**COMPUTER GRAPHICS**

**Objectives:**

- To make students understand about fundamentals of Graphics to enable them to design animated scenes for virtual object creations.
- To make the student present the content graphically.

**Outcomes:**

- Students can animate scenes entertainment.
- Will be able work in computer aided design for content presentation..
- Better analogy data with pictorial representation.

**UNIT-I:**

**Introduction:** Application areas of Computer Graphics, overview of graphics systems, video-display devices, raster-scan systems, random scan systems, graphics monitors and work stations and input devices

**Output primitives:** Points and lines, line drawing algorithms, mid-point circle and ellipse algorithms. Filled area primitives: Scan line polygon fill algorithm, boundary-fill and flood-fill algorithms

**UNIT-II:**

**2-D geometrical transforms:** Translation, scaling, rotation, reflection and shear transformations, matrix representations and homogeneous coordinates, composite transforms, transformations between coordinate systems

**2-D viewing :** The viewing pipeline, viewing coordinate reference frame, window to view-port coordinate transformation, viewing functions, Cohen-Sutherland and Cyrus-beck line clipping algorithms, Sutherland –Hodgeman polygon clipping algorithm

**UNIT-III:**

**3-D object representation :** Polygon surfaces, quadric surfaces, spline representation, Hermite curve, Bezier curve and B-Spline curves, Bezier and B-Spline surfaces. Basic illumination models, polygon rendering methods.

**3-D Geometric transformations:** Translation, rotation, scaling, reflection and shear transformations, composite transformations. **3-D viewing :** Viewing pipeline, viewing coordinates, view volume and general projection transforms and clipping.

**UNIT-IV:**

**Visible surface detection methods:** Classification, back-face detection, depth-buffer, scan-line, depth sorting, BSP-tree methods, area sub-division and octree methods

**UNIT-V:**

**Computer animation:** Design of animation sequence, general computer animation functions, raster animation, computer animation languages, key frame systems, motion specifications

**Text Books:**

3. “Computer Graphics *C version*”, Donald Hearn and M. Pauline Baker,

Pearson Education

4. “Computer Graphics Principles & practice”, second edition in C, Foley, VanDam, Feiner and Hughes, Pearson Education.

**References:**

1. Computer Graphics”, second Edition, Donald Hearn and M.Pauline Baker, PHI/Pearson Education.
2. Computer Graphics Second edition”, Zhigand xiang, Roy Plastock, Schaum’s outlines, Tata Mc-Graw hill edition.
3. rocedural elements for Computer Graphics, David F Rogers, Tata Mc Graw hill, 2<sup>nd</sup> edition.
4. Principles of Interactive Computer Graphics”, Neuman and Sproul, TMH.
5. Principles of Computer Graphics, Shalini Govil, Pai, 2005, Springer.
6. Computer Graphics, Steven Harrington, TMH

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****III Year B.Tech. CSE II-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>4</b>	<b>0</b>	<b>0</b>	<b>4</b>

**Department Elective -I**  
**SOFTWARE PROJECT MANAGEMENT**

**UNIT I**

**Conventional Software Management:** The waterfall model, conventional software Management performance.

**Evolution of Software Economics:** Software Economics, pragmatic software cost estimation.

**UNIT II**

**Improving Software Economics:** Reducing Software product size, improving software processes, improving team effectiveness, improving automation, Achieving required quality, peer inspections. **The old way and the new:** The principles of conventional software Engineering, principles of modern software management, transitioning to an iterative process.

**UNIT III**

Life cycle phases: **Engineering and production stages, inception, Elaboration, construction, transition phases.**

**Artifacts of the process:** The artifact sets, Management artifacts, Engineering artifacts, programmatic artifacts. **Model based software architectures:** A Management perspective and technical perspective. **Work Flows of the process:** Software process workflows, Iteration workflows.

**UNIT IV**

**Checkpoints of the process:** Major mile stones, Minor Milestones, Periodic status assessments. **Iterative Process Planning:** work breakdown structures, planning guidelines, cost and schedule estimating, Iteration planning process, Pragmatic planning.

**Project Organizations and Responsibilities:** Line-of-Business Organizations, Project Organizations, evolution of Organizations. **Process Automation:** Automation Building blocks, The Project Environment.

**UNIT V**

**Project Control and Process instrumentation:** The seven core Metrics, Management indicators, quality indicators, life cycle expectations, pragmatic Software Metrics, Metrics automation. **Tailoring the Process:** Process discriminates.

**Future Software Project Management:** modern Project Profiles, Next generation Software economics, modern process transitions.

**Case Study:** The command Center Processing and Display system- Replacement (CCPDS-R).

**Text Books:**

1. Software Project Management, Walker Royce: Pearson Education, 2005.

**References:**

1. Software Project Management, Bob Hughes and Mike Cotterell: Tata McGraw-Hill Ed.
2. Software Project Management, Joel Henry, Pearson Education.
3. Software Project Management in practice, Pankaj Jalote, Pearson Education. 2005.

**JNTUH COLLEGE OF ENGINEERING HYDERABAD**

**III Year B.Tech. CSE II-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>4</b>	<b>0</b>	<b>0</b>	<b>4</b>

**Department Elective-II**  
**PATTERN RECOGNITION**

**UNIT I:****INTRODUCTION**

Pattern and features – Training and learning in pattern recognition systems – Pattern recognition approaches – Statistical pattern recognition – Syntactic pattern recognition – Neural pattern recognition – Reasoning driven pattern recognition – Discriminant functions – Linear and Fisher's discriminant functions.

**UNIT II****STATISTICAL PATTERN RECOGNITION**

Gaussian model – Supervised learning – Parametric estimation – Maximum likelihood estimation – Bayesian parameter estimation – Perceptron algorithm – LMSE algorithm – Problems with Bayes approach – Pattern classification by distance functions – Maximum distance pattern classifier.

**UNIT III****CLUSTER ANALYSIS**

Unsupervised learning – Clustering for unsupervised learning and classification – C-means algorithm – Hierarchical clustering procedures – Graph theoretic approach to pattern clustering – Validity of clustering solutions.

**UNIT IV****SYNTACTIC PATTERN RECOGNITION**

Elements of formal grammar – String generation as pattern description – Recognition of syntactic description – Parsing – Stochastic grammar and applications – Graph based structural representation.

**UNIT V****FEATURES EXTRACTION AND RECENT ADVANCES**

Entropy minimization – Karhunen – Loeve transformation – Neural network structures for pattern recognition – Unsupervised learning – Self organizing networks – Fuzzy pattern classifiers – Genetic algorithms – Application to pattern recognition.

**Text Books:**

1. Robert J, Schalkoff, "*Pattern Recognition: Statistical, Structural and Neural Approaches*", John Wiley & Sons Inc., New York, 1992.

**References:**

1. Duda R.O. and Hart P.E., "*Pattern Classification and Scene Analysis*", John Wiley, New York, 2001
2. Morton Nadler and Eric Smith P., "*Pattern Recognition Engineering*", John Wiley and Sons, New York, 1993.

3. Touand , Gonzalez R. "*Patten Recognition Principles*" Addison Wesley, 1974.
4. Earl Gose, Richard Johnsonbaugh, Steve Jost, "*Pattern Recognition and Image Analysis*", Prentice Hall of India Private Ltd., New Delhi – 110 001, 1999.
5. Duda R.O, Hart .P.E., D.G. Stal, "*Pattern Classification*", John Wiley, 2001
6. Sergious Theodoridis, Konstantinos Koutroumbus, "*Pattern Recognition*", Elsevier, 2006



**JNTUH COLLEGE OF ENGINEERING HYDERABAD****III Year B.Tech. CSE II-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>4</b>	<b>0</b>	<b>0</b>	<b>4</b>

**Department Elective -II**  
**MULTIMEDIA APPLICATION DEVELOPMENT**

**UNIT-I**

**Fundamental concepts in Text and Image:** Multimedia and hypermedia, world wide web, overview of multimedia software tools. Graphics and image data representation graphics/image data types, file formats, Color in image and video: color science, color models in images, color models in video.

**Fundamental concepts in video and digital audio:** Types of video signals, analog video, digital video, digitization of sound, MIDI, quantization and transmission of audio.

**UNIT-II**

**Action Script I:** ActionScript Features, Object-Oriented ActionScript, Datatypes and Type Checking, Classes, Authoring an ActionScript Class, Inheritance, Authoring an ActionScript 2.0 Subclass, Interfaces, Packages, Exceptions, An OOP Application Framework, Using Components with ActionScript MovieClip Subclasses.

**UNIT III**

**Programming:** Abstraction levels: Device Drivers, System software, frameworks and libraries, application generators. Requirements for programming languages: very large data volumes, real time requirements, synchronization, reusability, expandability, maintainability, robustness. Object-Oriented application development: Basic terms of the object model, object model properties. Object-oriented Frameworks and class libraries: data type modeling, modeling data streams, Distribution of Objects: Example: PMG and CORBA, Example: Digital audio video council(DAVIC), DSM-CC.

**UNIT IV**

**Design:** Design specific properties of images. Visualizations: objective visualization, Abstract visualization, producing visualization, Good and bad visualization. Symbols: logos, icons, and pictograms. Illustrations. Image Production Techniques. Typography: layout. Esthetics.

**UNIT-V**

**User Interfaces:** Example: Remote controlled video camera. Usability: Goals of Usability, Solving usability problems. Direct Manipulations: Characteristics of Direct Manipulation, Forms of Directness, Discussion of Direct manipulation. Guidelines for User-friendly user interfaces. GUI and the audio medium. Innovative forms of interaction: Virtual Reality, computer-augmented reality.

**Text Books:**

1. Fundamentals of Multimedia by Ze-Nian Li and Mark S. Drew PHI/Pearson Education
2. Essentials ActionScript 2.0, Colin Moock, SPD O'REILLY.
3. Multimedia Applications, Steinmetz, Nahrstedt, Springer.

**References:**

1. Digital Multimedia, Nigel Chapman and Jenny Chapman, Wiley-Dreamtech Macromedia Flash MX Professional 2004 Unleashed, Pearson.

2. Multimedia and communications Technology, Steve Heath, Elsevier(Focal Press)
3. Multimedia Basics by Weixel Thomson
4. Multimedia Technology and Applications, David Hilman , Galgotia

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****III Year B.Tech. CSE II-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>4</b>	<b>0</b>	<b>0</b>	<b>4</b>

**Department Elective -II**  
**SOFTWARE TESTING METHODOLOGIES**

**UNIT-I:**

**Introduction:-** Purpose of testing, Dichotomies, model for testing, consequences of bugs, taxonomy of bugs

**Flow graphs and Path testing:-** Basics concepts of path testing, predicates, path predicates and achievable paths, path sensitizing, path instrumentation, application of path testing.

**UNIT-II:**

**Transaction Flow Testing:-** transaction flows, transaction flow testing techniques. Dataflow testing:- Basics of dataflow testing, strategies in dataflow testing, application of dataflow testing. **Domain Testing:-** domains and paths, Nice & ugly domains, domain testing, domains and interfaces testing, domain and interface testing, domains and testability.

**UNIT-III:**

**Paths, Path products and Regular expressions:-** path products & path expression, reduction procedure, applications, regular expressions & flow anomaly detection.

**Logic Based Testing:-** overview, decision tables, path expressions, kv charts, specifications.

**UNIT-IV:**

**State, State Graphs and Transition testing:-** state graphs, good & bad state graphs, state testing, Testability tips.

**UNIT-V:**

**Graph Matrices and Application:-** Motivational overview, matrix of graph, relations, power of a matrix, node reduction algorithm, building tools. ( Student should be given an exposure to a tool like JMeter or Win-runner).

**Text Books:**

1. Software Testing techniques - Baris Beizer, Dreamtech, second edition.
2. Software Testing Tools – Dr.K.V.K.K.Prasad, Dreamtech.

**References:**

1. The craft of software testing - Brian Marick, Pearson Education.
2. Software Testing Techniques – SPD(Oreille)
3. Software Testing in the Real World – Edward Kit, Pearson.
4. Effective methods of Software Testing, Perry, John Wiley.
5. Art of Software Testing – Meyers, John Wiley.

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****III Year B.Tech. CSE II-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>0</b>	<b>0</b>	<b>3</b>	<b>2</b>

**ADVANCED ENGLISH LANGUAGE COMMUNICATION SKILLS LAB****1. Introduction**

The introduction of the Advanced Communication Skills Lab is considered essential at 3<sup>rd</sup> year level. At this stage, the students need to prepare themselves for their careers which may require them to listen to, read, speak and write in English both for their professional and interpersonal communication in the globalised context.

The proposed course should be a laboratory course to enable students to use 'good' English and perform the following:

- Gathering ideas and information to organise ideas relevantly and coherently.
- Engaging in debates.
- Participating in group discussions.
- Facing interviews.
- Writing project/research reports/technical reports.
- Making oral presentations.
- Writing formal letters.
- Transferring information from non-verbal to verbal texts and vice-versa.
- Taking part in social and professional communication.

**2. Objectives:**

This Lab focuses on using multi-media instruction for language development to meet the following targets:

- To improve the students' fluency in English, through a well-developed vocabulary and enable them to listen to English spoken at normal conversational speed by educated English speakers and respond appropriately in different socio-cultural and professional contexts.
- Further, they would be required to communicate their ideas relevantly and coherently in writing.
- To prepare all the students for their placements.

**Learning Outcomes**

- 👍 Accomplishment of sound vocabulary and its proper use contextually.
- 👍 Flair in Writing and felicity in written expression.
- 👍 Enhanced job prospects.
- 👍 Effective Speaking Abilities

**3. Syllabus:**

The following course content to conduct the activities is prescribed for the Advanced Communication Skills (ACS) Lab:

- 1. Activities on Fundamentals of Inter-personal Communication and Building Vocabulary** - Starting a conversation – responding appropriately and relevantly – using the right body language – Role Play in different situations & Discourse Skills- using

visuals - Synonyms and antonyms, word roots, one-word substitutes, prefixes and suffixes, study of word origin, business vocabulary, analogy, idioms and phrases, collocations & usage of vocabulary.

2. **Activities on Reading Comprehension** –General Vs Local comprehension, reading for facts, guessing meanings from context, scanning, skimming, inferring meaning, critical reading & effective googling.
3. **Activities on Writing Skills** – Structure and presentation of different types of writing – *letter writing/Resume writing/ e-correspondence/ Technical report writing/ Portfolio writing* – planning for writing – improving one's writing.
4. **Activities on Presentation Skills** – Oral presentations (individual and group) through JAM sessions/seminars/PPTs and written presentations through posters/projects/reports/ e-mails/assignments etc.
5. **Activities on Group Discussion and Interview Skills** – Dynamics of group discussion, intervention, summarizing, modulation of voice, body language, relevance, fluency and organization of ideas and rubrics for evaluation- Concept and process, pre-interview planning, opening strategies, answering strategies, interview through tele-conference & video-conference and Mock Interviews.

#### 4. Minimum Requirement:

The Advanced Communication Skills (ACS) Laboratory shall have the following infra-structural facilities to accommodate at least 35 students in the lab:

- Spacious room with appropriate acoustics
- Round Tables with movable chairs
- Audio-visual aids
- LCD Projector
- Public Address system
- P – IV Processor, Hard Disk – 80 GB, RAM–512 MB Minimum, Speed – 2.8 GHZ
- T. V, a digital stereo & Camcorder
- Headphones of High quality

5. **Prescribed Lab Manual:** A book titled *A Course Book of Advanced Communication Skills(ACS) Lab* published by Universities Press, Hyderabad.

#### 6. Suggested Software:

The software consisting of the prescribed topics elaborated above should be procured and used.

- **Oxford Advanced Learner's Compass**, 8<sup>th</sup> Edition
- **DELTA's key to the Next Generation TOEFL Test: Advanced Skill Practice.**
- **Lingua TOEFL CBT Insider**, by Dreamtech
- **TOEFL & GRE** (KAPLAN, AARCO & BARRONS, USA, Cracking GRE by CLIFFS)
- **The following software from 'train2success.com'**
  - **Preparing for being Interviewed**
  - **Positive Thinking**
  - **Interviewing Skills**
  - **Telephone Skills**
  - **Time Management**

#### 7. Books Recommended:

1. **Technical Communication** by Meenakshi Raman & Sangeeta Sharma, Oxford University Press 2009.
2. **English Language Communication : A Reader cum Lab Manual** Dr A Ramakrishna Rao, Dr G Natanam & Prof SA Sankaranarayanan, Anuradha Publications, Chennai 2008.
3. **Advanced Communication Skills Laboratory Manual** by Sudha Rani, D, Pearson Education 2011.
4. **Technical Communication** by Paul V. Anderson. 2007. Cengage Learning pvt. Ltd. New Delhi.
5. **Business and Professional Communication: Keys for Workplace Excellence.** Kelly M. Quintanilla & Shawn T. Wahl. Sage South Asia Edition. Sage Publications. 2011.
6. **The Basics of Communication:A Relational Perspective.** Steve Duck & David T. McMahan. Sage South Asia Edition. Sage Publications. 2012.
7. **English Vocabulary in Use** series, Cambridge University Press 2008.
8. **Management Shapers Series** by Universities Press(India)Pvt Ltd., Himayatnagar, Hyderabad 2008.
9. **Handbook for Technical Communication** by David A. McMurrey & Joanne Buckley. 2012. Cengage Learning.
10. **Communication Skills** by Leena Sen, PHI Learning Pvt Ltd., New Delhi, 2009.
11. **Handbook for Technical Writing** by David A McMurrey & Joanne Buckely CENGAGE Learning 2008.
12. **Job Hunting** by Colm Downes,Cambridge University Press 2008.
13. **Master Public Speaking** by Anne Nicholls, JAICO Publishing House, 2006.
14. **English for Technical Communication for Engineering Students,** Aysha Vishwamohan, Tata Mc Graw-Hil 2009.
15. Books on TOEFL/GRE/GMAT/CAT/IELTS by Barron's/DELTA/Cambridge University Press.
16. **International English for Call Centres** by Barry Tomalin and Suhashini Thomas, Macmillan Publishers, 2009.

#### **DISTRIBUTION AND WEIGHTAGE OF MARKS:**

##### ***Advanced Communication Skills Lab Practicals:***

1. The practical examinations for the ACS Laboratory practice shall be conducted as per the University norms prescribed for the core engineering practical sessions.
2. For the English Language lab sessions, there shall be continuous evaluation during the year for 25 sessional marks and 50 End Examination marks. Of the 25 marks, 15 marks shall be awarded for day-to-day work and 10 marks to be awarded by conducting Internal Lab Test(s). The End Examination shall be conducted by the teacher concerned, by inviting the External Examiner from outside. In case of the non-availability of the External Examiner, other teacher of the same department can act as the External Examiner.

##### **Mini Project: As a part of Internal Evaluation**

###### **1. Seminar/ Professional Presentation**

###### **2. A Report on the same has to be prepared and presented.**

- \* *Teachers may use their discretion to choose topics relevant and suitable to the needs of students.*
- \* *Not more than two students to work on each mini project.*
- \* *Students may be assessed by their performance both in oral presentation and written report.*

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****III Year B.Tech. CSE II-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>0</b>	<b>0</b>	<b>3</b>	<b>2</b>

**DATA MINING AND DATA WAREHOUSING LAB****LIST OF EXPERIMENTS:-**

Experiments using Weka &amp; Clementine Tools

1. Data Processing Techniques :
  - (i) Data cleaning
  - (ii) Data transformation - Normalization
  - (iii) Data integration
2. Partitioning - Horizontal, Vertical, Round Robin, Hash based
3. Data Warehouse schemas – star, snowflake, fact constellation
4. Data cube construction – OLAP operations
5. Data Extraction, Transformations & Loading operations
6. Implementation of Attribute oriented induction algorithm
7. Implementation of apriori algorithm
8. Implementation of FP – Growth algorithm
9. Implementation of Decision Tree Induction
10. Calculating Information gain measures
11. Classification of data using Bayesian approach
12. Classification of data using K – nearest neighbour approach
13. Implementation of K – means algorithm
14. Implementation of BIRCH algorithm
15. Implementation of PAM algorithm
16. Implementation of DBSCAN algorithm

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****III Year B.Tech. CSE II-Sem**

L	T	P	C
0	0	3	2

**WEB TECHNOLOGIES LAB**

- Develop static pages (using Only HTML) of an online Book store. The pages should resemble: [www.amazon.com](http://www.amazon.com) The website should consist the following pages.
  - Home page
  - Registration and user Login
  - User Profile Page
  - Books catalog
  - Shopping Cart
  - Payment By credit card
  - Order Conformation
- Validate the Registration, user login, user profile and payment by credit card pages using JavaScript.
- Create and save an XML document at the server, which contains 10 users information. Write a program, which takes User Id as an input and returns the user details by taking the user information from the XML document.
- Bean Assignments
  - Create a JavaBean which gives the exchange value of INR(Indian Rupees) into equivalent American/Canadian/Australian Dollar value.
  - Create a simple Bean with a label - which is the *count* of number of clicks. Than create a BeanInfo class such that only the “*count*” property is visible in the Property Window.
  - Create two Beans-a)KeyPad .b)DisplayPad .After that integrate the two Beans to make it work as a Calculator.
  - Create two Beans Traffic Light(Implemented as a Label with only three background colours-Red,Green,Yellow) and Automobile(Implemented as a TextBox which states its state/movement). The state of the Automobile should depend on the following Light Transition Table.

Light Transition	Automobile State
Red ---> Yellow	Ready
Yellow ---> Green	Move
Green --> Red	Stopped

- Install TOMCAT web server. Convert the static web pages of assignments 2 into dynamic web pages using servlets and cookies. Hint: Users information (user id, password, credit card number) would be stored in web.xml. Each user should have a separate Shopping Cart.
- Redo the previous task using JSP by converting the static web pages of assignments 2 into dynamic web pages. Create a database with user information and books information. The books catalogue should be dynamically loaded from the database. Follow the MVC architecture while doing the website.



**JNTUH COLLEGE OF ENGINEERING HYDERABAD****III Year B.Tech. CSE II-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>2</b>	<b>0</b>	<b>0</b>	<b>2</b>

**DISASTER MANAGEMENT****UNIT 1 : Understanding Disaster**

Concept of Disaster

Different approaches

Concept of Risk

Levels of Disasters

Disaster Phenomena and Events (Global, national and regional)

**Hazards and Vulnerability**

Natural and man-made hazards; response time, frequency and forewarning levels of different hazards

Characteristics and damage potential of natural hazards; hazard assessment

Dimensions of vulnerability factors; vulnerability assessment

Vulnerability and disaster risk

Vulnerabilities to flood and earthquake hazards

**UNIT 2 : Disaster Management Mechanism**

Concepts of risk management and crisis managements

Disaster Management Cycle

Response and Recovery

Development, Prevention, Mitigation and Preparedness

Planning for Relief

**UNIT 3: Capacity Building**

Capacity Building: Concept

Structural and Nonstructural Measures

Capacity Assessment; Strengthening Capacity for Reducing Risk

Counter-Disaster Resources and their utility in Disaster Management

Legislative Support at the state and national levels

**UNIT 4: Coping with Disaster**

Coping Strategies; alternative adjustment processes

Changing Concepts of disaster management

Industrial Safety Plan; Safety norms and survival kits

Mass media and disaster management

**UNIT 5: Planning for disaster management**

Strategies for disaster management planning

Steps for formulating a disaster risk reduction plan

Disaster management Act and Policy in India

Organizational structure for disaster management in India

Preparation of state and district disaster management plans

**Text Books**

1. Alexander, D. Natural Disasters, ULC press Ltd, London, 1993.

2. Carter, W.N. Disaster Management: A Disaster Management Handbook, Asian Development Bank, Bangkok, 1991.
3. Manual on Natural Disaster Management in India, NCDM, New Delhi, 2001.

**References**

1. Abarquez I. & Murshed Z. Community Based Disaster Risk Management: Field Practitioner's Handbook, ADPC, Bangkok, 2004.
2. Goudie, A. Geomorphological Techniques, Unwin Hyman, London 1990.
3. Goswami, S.C Remote Sensing Application in North East India, Purbanchal Prakesh, Guwahati, 1997.
4. Chakrabarty, U.K. Industrial Disaster Management and Emergency Response, Asian Book Pvt. Ltd., New Delhi 2007.
5. Disaster Management in India, Ministry of Home Affairs, Government of India, New Delhi, 2011.
6. National Policy on Disaster Management, NDMA, New Delhi, 2009
7. Disaster Management Act. (2005), Ministry of Home Affairs, Government of India, New Delhi, 2005.
8. District Disaster Management Plan-Model Template, NIDM, New Delhi, 2005.
9. Disaster Management, Future challenge and opportunities, Edited by Jagbir singh, I.K. International publishing home Pvt, Ltd.

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****IV Year B.Tech. CSE I-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>4</b>	<b>1</b>	<b>0</b>	<b>5</b>

**SOFTWARE REQUIREMENTS AND ESTIMATION****(PGC)****UNIT I**

Software Requirements: What and Why

Essential Software requirement, Good practices for requirements engineering, Improving requirements processes, Software requirements and risk management

Software Requirements Engineering

Requirements elicitation, requirements analysis documentation, review, elicitation techniques, analysis models, Software quality attributes, risk reduction through prototyping, setting requirements priorities, verifying requirements quality,.

**UNIT II**

Software Requirements Management

Requirements management Principles and practices, Requirements attributes, Change Management Process, Requirements Traceability Matrix, Links in requirements chain

Software Requirements Modeling

Use Case Modeling, Analysis Models, Dataflow diagram, state transition diagram.

**UNIT III**

state transition diagram, class diagrams, Object analysis, Problem Frames

Software Estimation

Components of Software Estimations, Estimation methods, Problems associated with estimation, Key project factors that influence estimation

**UNIT IV**

Size Estimation

Two views of sizing, Function Point Analysis, Mark II FPA, Full Function Points, LOC Estimation, Conversion between size measures,

Effort, Schedule and Cost Estimation

What is Productivity? Estimation Factors, Approaches to Effort and Schedule Estimation, COCOMO II, Putnam Estimation Model, Algorithmic models, Cost Estimation

**UNIT V**

Tools for Requirements Management and Estimation

Requirements Management Tools: Benefits of using a requirements management tool, commercial requirements management tool, Rational Requisite pro, Caliber – RM, implementing requirements management automation, Software Estimation Tools:

Desirable features in software estimation tools, IFPUG, USC's COCOMO II, SLIM (Software Life Cycle Management) Tools

**TEXT BOOK:**

1. Software Requirements and Estimation by Rajesh Naik and Swapna Kishore, Tata Mc Graw Hill

**REFERENCES:**

1. Software Requirements by Karl E. Weigers

2. Managing Software Requirements, Dean Leffingwell & Don Widrig, Pearson Education, 2003.
3. Mastering the requirements process, second edition, Suzanne Robertson & James Robertson, Pearson Education, 2006.
4. Estimating Software Costs, Second edition, Capers Jones, Tata McGraw-Hill, 2007.
5. Practical Software Estimation, M.A. Parthasarathy, Pearson Education, 2007.
6. Measuring the software process, William A. Florac & Anita D. Carleton, Pearson Education,

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****IV Year B.Tech. CSE I-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>4</b>	<b>1</b>	<b>0</b>	<b>5</b>

**PG Elective - I****SOFTWARE QUALITY ASSURANCE AND TESTING****UNIT I****Software Quality Assurance Framework and Standards SQA Framework:**

What is Quality? Software Quality Assurance, Components of Software Quality Assurance – Software Quality Assurance Plan: Steps to develop and implement a Software Quality Assurance Plan – Quality Standards: ISO 9000 and Companion ISO Standards, CMM, CMMI, PCMM, Malcom Balridge, 3 Sigma, 6 Sigma

**UNIT II****Software Quality Assurance Metrics and Measurement Software Quality Metrics:**

Product Quality metrics, In-Process Quality Metrics, Metrics for Software Maintenance, Examples of Metric Programs – Software Quality metrics methodology: Establish quality requirements, Identify Software quality metrics, Implement the software quality metrics, analyze software metrics results, validate the software quality metrics – Software quality indicators – Fundamentals in Measurement theory

**UNIT III****Software Testing Strategy and Environment:**

Establishing testing policy, structured approach to testing, test factors, Economics of System Development Life Cycle (SDLC) Testing Software Testing Methodology  
Defects hard to find, verification and validation, functional and structural testing, workbench concept, eight considerations in developing testing methodologies, testing tactics checklist

**UNIT IV****Software Testing Techniques**

Black-Box, Boundary value, Bottom-up, Branch coverage, Cause-Effect graphing, CRUD, Database, Exception, Gray-Box, Histograms, Inspections, JADs, Pareto Analysis, Prototyping, Random Testing, Risk-based Testing, Regression Testing, Structured Walkthroughs, Thread Testing, Performance Testing, White-Box Testing

**Software Testing Tools**

Taxonomy of Testing tools, Methodology to evaluate automated testing tools, Load Runner, Win runner and Rational Testing Tools, Silk test, Java Testing Tools, JMetra, JUNIT and Cactus.

**UNIT V****Testing Process**

Eleven Step Testing Process: Assess Project Management Development Estimate and Status, Develop Test Plan, Requirements Phase Testing, Design Phase Testing, Program Phase Testing, Execute Test and Record Results, Acceptance Test, Report test results, testing software installation, Test software changes, Evaluate Test Effectiveness.

**Testing Specialized Systems and Applications**

Testing Client/Server – Web applications, Testing off the Shelf Components, Testing Security, Testing a Data Warehouse

**Text Books:**

1. Effective Methods for Software Testing, 2nd Edition, William E. Perry , Second Edition, Wiley India, 2006.
2. Software Quality, Mordechai Ben-Menachem/Garry S. Marliss, Thomson Learning publication, 1997.

**Reference Books:**

1. Software Testing Tools, K.V.K.K. Prasad, Dream tech press, 2008.
2. Practical Software Testing, Ilene Burnstein, Springer, 2003.
3. Software Testing, Srinivasan Desikan & Gopalaswamy Ramesh, Pearson Education, 2006.
4. Software testing techniques, Scott Loveland & Geoffrey Miller, Shroff Publishers, 2005.
5. Software testing, Ron Patton, second edition, Pearson Education, 2004.
6. Software Quality, Martin Wieczorek & Dirk Meyerhoff, Springer, 2001.
7. Software Testing: A Craftsman's approach, Paul C. Jorgensen, Second edition, CRC press, 2002.

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****IV Year B.Tech. CSE I-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>4</b>	<b>1</b>	<b>0</b>	<b>5</b>

**PG Elective - I**  
**MOBILE COMPUTING**

**UNIT I****Introduction**

Mobile Communications, Mobile Computing – Paradigm, Promises/Novel Applications and Impediments and Architecture; Mobile and Handheld Devices, Limitations of Mobile and Handheld Devices.

GSM – Services, System Architecture, Radio Interfaces, Protocols, Localization, Calling, Handover, Security, New Data Services, GPRS, CSHSD, DECT.

**UNIT –II****(Wireless) Medium Access Control (MAC)**

Motivation for a specialized MAC (Hidden and exposed terminals, Near and far terminals), SDMA, FDMA, TDMA, CDMA, Wireless LAN/(IEEE 802.11)

**Mobile Network Layer**

IP and Mobile IP Network Layers, Packet Delivery and Handover Management, Location Management, Registration, Tunneling and Encapsulation, Route Optimization, DHCP.

**UNIT –III****Mobile Transport Layer**

Conventional TCP/IP Protocols, Indirect TCP, Snooping TCP, Mobile TCP, Other Transport Layer Protocols for Mobile Networks.

**Database Issues**

Database Hoarding & Caching Techniques, Client-Server Computing & Adaptation, Transactional Models, Query processing, Data Recovery Process & QoS Issues.

**UNIT IV****Data Dissemination and Synchronization**

Communications Asymmetry, Classification of Data Delivery Mechanisms, Data Dissemination, Broadcast Models, Selective Tuning and Indexing Methods, Data Synchronization – Introduction, Software, and Protocols

**UNIT V****Mobile Adhoc Networks (MANETs)**

Introduction, Applications & Challenges of a MANET, Routing, Classification of Routing Algorithms, Algorithms such as DSR, AODV, DSDV, etc. , Mobile Agents, Service Discovery.

**Protocols and Platforms for Mobile Computing**  
WAP, Bluetooth, XML, J2ME, JavaCard, PalmOS, Windows CE, SymbianOS, Linux for Mobile Devices, Android.

**Text Books:**

1. Jochen Schiller, “Mobile Communications”, Addison-Wesley, Second Edition, 2009.
2. Raj Kamal, “Mobile Computing”, Oxford University Press, 2007, ISBN: 0195686772

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****IV Year B.Tech. CSE I-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>4</b>	<b>1</b>	<b>0</b>	<b>5</b>

**PG Elective - I**  
**BUSINESS INTELLIGENCE & BIG DATA**

**UNIT-I**

**Data Science:** Introduction, working with data at scale, data scientist, the SMAQ stack for big data, scraping, cleaning & selling big data

Data Hand Tools - free data tools for journalists.

Data Issues- Introduction, anonymization, risk of de-anonymization, Big data & semantic web, meta data.

**UNIT-II**

**Applications of Data:**Product & Process – Twitter archive, data journalism & data tools, newsroom stack, bridging the data divide, data analysis path, Big data in education & academic disciplines, Discussion of Facebook

**UNIT-III**

**BI foundations:** Understanding BI, Describing BI, Defining BI cycle, Enabling BI, Bridging the Analysis Gap-Multidimensional analysis, Operation Systems, BI Systems.

**UNIT-IV**

**Defining BI Technologies:**The High-level view, Reporting & Analysis, the data warehouse and Data warehousing Framework, Identifying BI opportunities.

**UNIT-V**

**Implementing a BI solution:**implementation strategy, Fundamental decisions.

Case studies- Audi AG, The Frank Russell Company.

**Textbooks:**

1. Elizabeth Vitt, Michael Luckevich, StaciaMisner, “Business Intelligence”, Microsoft Press, 2010.
2. Big Data Now, O’Reily Radar Team.

**References:**

1. Rajiv Sabherwal, Irma Becerra- Fernandez, “Business Intelligence-Practices, Technologies and Management”, John Wiley 2011.
2. Larissa T. Moss, ShakuAtre, “Business Intelligence Roadmap”, Addison –Wesley IT Series.
3. Yuli Vasiliev, “Oracle Business Intelligence: The Condensed Guide to Analysis and Reporting”, SPD Shroff ,2012.



**JNTUH COLLEGE OF ENGINEERING HYDERABAD****IV Year B.Tech. CSE I-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>4</b>	<b>0</b>	<b>0</b>	<b>4</b>

**PG Elective - II**  
**SOFTWARE ARCHITECTURE & DESIGN PATTERNS**

**UNIT I****Envisioning Architecture**

The Architecture Business Cycle, What is Software Architecture, Architectural patterns, reference models, reference architectures, architectural structures and views.

Creating an Architecture

Quality Attributes, Achieving qualities, Architectural styles and patterns, designing the Architecture, Documenting software architectures, Reconstructing Software Architecture.

**UNIT II**

Analyzing Architectures

Architecture Evaluation, Architecture design decision making, ATAM, CBAM.

**UNIT III**

Moving from one system to many

Software Product Lines, Building systems from off the shelf components, Software architecture in future.

**UNIT IV**

Patterns

Pattern Description, Organizing catalogs, role in solving design problems ,Selection and usage.

Creational and Structural patterns

Abstract factory, builder, factory method, prototype, singleton, adapter, bridge, composite, façade, flyweight, Proxy.

**UNIT V**

Behavioral patterns

Chain of responsibility, command, Interpreter, iterator, mediator, memento, observer, state, strategy, template method, visitor.

Case Studies

A-7E – A case study in utilizing architectural structures, The World Wide Web - a case study in interoperability, Air Traffic Control – a case study in designing for high availability, Celsius Tech – a case study in product line development

**Text Books:**

1. Software Architecture in Practice, second edition, Len Bass,Paul Clements&Rick Kazman, Pearson Education,2003.
2. Design Patterns, Erich Gamma, Pearson Education, 1995.

**Reference Books:**

1. Beyond Software architecture, Luke Hohmann, Addison wesley, 2003.
2. Software architecture, David M. Dikel, David Kane and James R. Wilson, Prentice Hall PTR, 2001
3. Software Design, David Budgen, second edition, Pearson education, 2003
4. Head First Design patterns, Eric Freeman & Elisabeth Freeman, O'REILLY, 2007.
5. Design Patterns in Java, Steven John Metsker & William C. Wake, Pearson education, 2006
6. J2EE Patterns, Deepak Alur, John Crupi & Dan Malks, Pearson education, 2003.
7. Design Patterns in C#, Steven John metsker, Pearson education, 2004.

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****IV Year B.Tech. CSE I-Sem**

<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>
<b>4</b>	<b>0</b>	<b>0</b>	<b>4</b>

**PG Elective - II**  
**SOFTWARE DESIGN AND ENGINEERING**

**UNIT I**

**Introduction to Software Engineering:** The evolving role of software, Changing Nature of Software, legacy software, Software myths.

**A Generic view of process:** Software engineering- A layered technology, a process framework, The Capability Maturity Model Integration (CMMI), Process patterns, process assessment, personal and team process models.

**Process models:** The waterfall model, Incremental process models, Evolutionary process models, Specialized process models, The Unified process.

**Software Requirements:** Functional and non-functional requirements, User requirements, System requirements, Interface specification, the software requirements document.

**Requirements engineering process:** Feasibility studies, Requirements elicitation and analysis, Requirements validation, Requirements management.

**UNIT II****Software Design**

The nature of the design process, transferring design knowledge, constraints upon the design process and product, recording design decisions, designing with others, context for design, economic factors, assessing design qualities, quality attributes of the design product, assessing the design process. Representing abstract ideas, design view points, the architecture concept, design methods, design patterns, design representations, rationale for design methods.

**Design Processes and Strategies :** The role of strategy in design methods, describing the design process – The D – Matrix, design by top-down decomposition, design by composition, organizational influences upon design.

**UNIT III****Designing with objects and components**

Designing with objects : Design practices for object-oriented paradigm, Object-oriented paradigm, Object-oriented frame works, Hierarchical object oriented design process and heuristics, the fusion method, the unified process.

**Component - based design:**

The component concept, designing with components, designing components, COTS.

**User Interface design**

The Golden rules, Interface analysis and design models, user and task analysis, analysis of display content and work environment, applying interface design issues, design evaluation.

**UNIT IV****Concepts Of Software Projects**

**Project Management :** The management spectrum: people, product, process and project, W5HH principle, Critical practices

**Metrics for Process and Projects :** Process metrics, project metrics, size-oriented metrics, function-oriented metrics, Object-oriented and use-case metrics, metrics for software quality, integrating metrics with in software process.

**UNIT V****Project Scheduling and Management**

**Project Scheduling:** Basic concepts, project scheduling , defining a task set and task network, timeline charts, tracking the schedule, tracking the progress for an OO project, Earned value analysis.

**Risk Management:** Reactive Vs. Proactive risk strategies, software risks, risk identification, risk projection, risk refinement, risk mitigation and monitoring, the RMMM plan.

**Text Books:**

1. Software design, David Budgen, second edition, Pearson education, 2003
2. Software Engineering: A practitioner's Approach, Roger S Pressman, sixth edition. McGrawHill International Edition, 2005

**Reference Books:**

1. Applying domain- driven design and patterns, jimmy Nilsson, Pearson education, 2006
2. Software Engineering Foundations, Ian Sommerville, seventh edition, Pearson education, 2004.
3. Software Project Management , Bob Hughes & Mike Cotterell, Fourth edition, Tata McGraw Engineering : A Primer, Waman S Jawadekar, Tata McGraw-Hill, 2008
4. The Art of Project Management, Scott Berkun, O'Reilly, 2005.
5. Software Engineering, Project Management , Richard H. Thayer & Edward Yourdon, second edition, Wiley india, 2004.
6. Software Engineering foundations, Yingxu Wang Auerbach publications, 2008.
7. Applied Software Project Management , Andrew Stellman & Jennifer Greene, O'Reilly, 2006.

**JNTUH COLLEGE OF ENGINEERING HYDERABAD****IV Year B.Tech. CSE I-Sem****L T P C**  
**006 4****Case Tools & OOAD LAB**  
**SOFTWARE TESTING LAB****RATIONAL ROSE - CASE TOOL**  
**UML INTRODUCTION**

UML Diagrams for:

1. ATM System
2. Library Management System
3. Student Enrolment System
4. Online Reservation System
5. Stock Maintenance System
6. Payroll System

**WINRUNNER-CASE TOOL**  
**INTRODUCTION TO WINRUNNER**

1. Recording in context sensitive mode and analog mode
2. a) GUI checkpoint for single property  
b) GUI checkpoint for single object/window  
c) GUI checkpoint for multiple objects
3. a) Bitmap checkpoint for object/window  
b) Bitmap checkpoint for screen area
4. a) Database checkpoint for Default check  
b) Database checkpoint for custom check  
c) Database checkpoint for runtime record check
5. a) Data driven test for dynamic test data submission  
b) Data driven test through flat files  
c) Data driven test through front grids  
d) Data driven test through excel test
6. a) Batch testing without parameter passing  
b) Batch testing with parameter passing  
c) Data driven batch
7. Function generator
8. User defined functions
9. Executing a prepared query
10. Compiled module
11. a) Synchronization for object/window property  
b) Synchronization for object/window bitmap  
c) Synchronization point for screen area bitmap
12. a) Test case for calculator in windows application  
b) Test case for flight Reservation System
13. Test plan for library management system
14. Binary search and linear search implementation in Win runner
15. Stack & queues implementation in win runner
16. Sorting implementation in win runner

\*\*\*\*\*



## COURSE DESCRIPTOR

# Research Methodologies in Software Engineering and Computer Science

Forskningsmetodik i programvaruteknik och datavetenskap

7,5 ECTS credit points (7,5 högskolepoäng)

**Course code:** PA2404  
**Educational level:** Advanced level  
**Course level:** A1N  
**Field of education:** Technology  
**Subject group:** Computer Technology

**Subject area:** Software Engineering  
**Version:** 7  
**Applies from:** 2012-03-26  
**Approved:** 2012-02-06

### 1 Course title and credit points

The course is titled Research Methodologies in Software Engineering and Computer Science/Forskningsmetodik i programvaruteknik och datavetenskap and awards 7,5 credit points. One credit point (högskolepoäng) corresponds to one credit point in the European Credit Transfer System (ECTS).

### 2 Decision and approval

This course is established by School of Computing 2008-01-01. The course descriptor was revised by School of Computing and applies from 2012-03-26. Reg no: COM-560-0024-2012

### 3 Objectives

A key issue in the research in Software Engineering and Computer Science is developing, evaluating and comparing methods, techniques and tools and how these affect different systems or organisations. Through this course the student will achieve an understanding of research methodologies that enable such evaluations and comparisons. The student will be introduced to societal and ethical aspects of such research and will gain their first experiences in planning, conducting and reporting a research project.

### 4 Content

The course covers how to evaluate literature, how to formulate a goal for a research project, how to formulate a frame of questioning and set up matching hypotheses. The course looks at qualitative as well as quantitative research methods, such as case studies, surveys, experiments with human participants, mathematical modelling and simulation. The course also covers research ethics and how to arrange and present the research study in a research article.

### 5 Aims and learning outcomes

After completing the course, the student will:

- be able to use scientific databases and search engines to identify relevant research articles for a specific research question.
- in their own words be able to describe, relate to, and discuss the results that are presented in research articles.
- be able to formulate a concrete research problem.
- be able to explain similarities and differences between a qualitative research study and a quantitative experiment.
- be able to design a simple research study and thereby select a suitable quantitative or qualitative method.
- be able to write a simple research article according to established academic praxis.
- be able to describe how to conduct a research study from beginning to end.
- be able to describe the ethical implications of a research project.
- be able to reference other people's work according to academic standards.
- be able to discuss and relate to the concepts of plagiarism and copyright.
- be able to explain the method and practice of science and relate it to their work.
- be able to exemplify the possibilities and limits of science, its role in society, and people's responsibility for

how it is used.

- be able to determine societal aspects of software engineering and computer science.

## 6 Generic skills

During the course the following general abilities will be trained:

- Planning and time management
- Ability to generate new ideas (creativity)
- Teamwork
- Ability to work independently
- Qualitative thinking

## 7 Learning and teaching

The course is built around lectures and assignments. Students are expected to participate actively and contribute to lectures through discussions, questions and experiences. The assignments are constructed in such a manner that students are trained in designing different research studies and how to rework a research study into a research article. A diagnostic examination is used to test theoretical knowledge, comprehension, and ability to make judgments.

The teaching language is English.

## 8 Assessment and grading

### *Examination of the course*

Code	Module	Credit	Grade
0905	Assignments	6 hp	U/G
0915	Examination <sup>[1]</sup>	1.5 hp	F-A

<sup>1</sup> Determines the final grade for the course, which will only be issued when all components have been approved. The course will be graded F Fail, FX Fail, E Sufficient, D Satisfactory, C Good, B Very good or A Excellent.

## 9 Course evaluation

The course coordinator is responsible for systematically gathering feedback from the students in course evaluations and making sure that the results of these feed back into the development of the course.

## 10 Prerequisites

To gain admission into the course, the student is required to have a Bachelor's degree in computer science or software engineering.

## 11 Field of education and subject area

The course is part of the field of education Technology and is included in the subject area Software Engineering. The course can also be included in the subject area Computer Science.

## 12 Restrictions regarding degree

The course cannot form part of a degree with another course, the content of which completely or partly corresponds with the contents of this course.

## 13 Additional information

No assignments will be graded unless the student has signed the Rules of Conduct.

## 14 Course literature and other teaching material

Main literature

1. Title: Research Design - Qualitative, Quantitative and Mixed Method Approaches, 2nd ed.

Author: J.W. Creswell Publisher: Sage Publications

Published: 2002, Number of pages: 245 ISBN 0-7619-2442-6

2. Title: Projects in Computing and Information Systems

Author: Christian Dawson

Publisher: Pearson

Published: 2005, Number of pages: 256

ISBN: 0321263553

Reference literature

1. Title: Real World Research Author: C. Robson

Publisher: Blackwell Publishing, Malden MA, Published: 2002

2. Title: Experimentation in Software Engineering - An Introduction  
Authors: Wohlin, P. Runeson, M. Höst, M.C. Ohlsson, B. Regnell, A. Wesslén  
Publisher: Kluwer Academic Publishers, Dordrecht, the Netherlands  
Published: 2000

■

page 2





## COURSE DESCRIPTOR

### Software Verification and Validation

#### Verifiering och validering

7,5 ECTS credit points (7,5 högskolepoäng)

**Course code:** PA2405

**Educational level:** Advanced level

**Course level:** A1N

**Field of education:** Technology

**Subject group:** Computer Technology

**Subject area:** Software Engineering

**Version:** 6

**Applies from:** 2011-01-17

**Approved:** 2010-05-05

**Replaces course descriptor approved:** 2007-10-22

#### 1 Course title and credit points

The course is titled Software Verification and Validation/Verifiering och validering and awards 7,5 credit points. One credit point (högskolepoäng) corresponds to one credit point in the European Credit Transfer System (ECTS).

#### 2 Decision and approval

This course is established by School of Computing 2007-10-22. The course descriptor was revised by School of Computing and applies from 2011-01-17. Replaces PAD002.

Reg no COM-56-0075-2009 (Previous Reg no TEK56-405/07)

#### 3 Objectives

All software of any significant size needs to be verified and validated accordingly to assure a certain level of quality.

The objective with this course is to provide the participants with an overview of verification and validation (V&V) of software systems. In addition several techniques (as used in industry or academia) are practiced and further examined.

#### 4 Content

The course is built around three interdisciplinary thematic blocks:

- Software inspection (verification metrics, requirements, design and code inspection)
- Software testing (levels, methods and types of test, informal and formal validation, test planning, validation metrics)
- Software reliability engineering (reliability predictions, operational profiles, test efficiency, deployment, software reliability models, step-by-step process implementation)

All three blocks will cover the history (background), concepts (techniques) and applications, and applied usage of at least one technique from each theme respectively.

In addition, a framework for V&V (providing a crude systematic review of V&V) will be given and, in the end, improved on by the participants.

#### 5 Aims and learning outcomes

On completion of the course the student will be able to:

- Independently describe and, in a group discuss, techniques in the areas of software inspection, software testing and software reliability engineering.
- Individually name and explain a number of key research questions remaining to be answered in each of the three thematic blocks.
- In a group be able to apply at least one technique in each of the three thematic blocks.
- Individually apply software reliability engineering techniques.

#### 6 Generic skills

The following generic skills are trained in the course:

- Planning and time management
- Critical thinking
- Information search
- Problem solving
- Analytical ability
- Team work

## 7 Learning and teaching

The course is structured around a three-day seminar (compulsory), a number of lectures, lab assignments and a written exam. The course starts with an introductory seminar and then consists of a series of lectures where a number of themes (see Section 4) are introduced to the participants. During each lecture additional time is spent on discussions and team assignments. Assignment 1 covers the inspection process and the accompanying techniques while Assignment 2 introduces the participant to requirement-based testing.

Assignments 1 and 2 are performed in groups, while the written exam is conducted on an individual basis. The teaching language is English.

## 8 Assessment and grading

### *Examination of the course*

Code	Module	Credit	Grade
1005	Assignment 1	1.5 hp	F-A
1015	Assignment 2	1.5 hp	F-A
1025	Seminars	1 hp	F-A
1035	Written examination	3.5 hp	F-A

The course will be graded F Fail, FX Fail, E Sufficient, D Satisfactory, C Good, B Very good or A Excellent. The total grade is based on a weighted average (e.g. seminar is worth 1/7.5th of the total grading).

## 9 Course evaluation

The course coordinator is responsible for systematically gathering feedback from the students in course evaluations and making sure that the results of these feed back into the development of the course.

## 10 Prerequisites

The student must have a Bachelor of Science in Computer Science or Software Engineering

## 11 Field of education and subject area

The course is part of the field of education Technology and is included in the subject area Software Engineering.

## 12 Restrictions regarding degree

The course cannot form part of a degree with another course, the content of which completely or partly corresponds with the contents of this course.

## 13 Course literature and other teaching material

Main literature

1. Title: Software Verification and Validation for Practitioners and Managers, Second Edition Author: Steven R. Rakitin

Hardcover: 424 pages

Publisher: Artech House Publishers; 2nd edition (August 1, 2001)

ISBN: 1580532969

2. Introduction to software testing

Författare: Paul Ammann och Jeff Offutt Förlag: Cambridge University Press (2008) ISBN:

9780521880381

Reference literature

3. Title: What is Software Testing? And Why is it So Hard?

Author: James A. Whittaker

Journal: IEEE Software Vol. 17, Issue 1 (Jan. 2000), pp. 70-79.

Publisher: IEEE Computer Society Press

4. Title: A Survey of Software Inspection Technologies. In Handbook on Software Engineering and Knowledge Engineering, Vol. 2, pp. 517-555.

Author: Oliver Laitenberg

Publisher: World Scientific Publishing (2002)

5. Title: Using Inspection Data for Defect Estimation Author: Stefan Biffl  
Journal: IEEE Software Vol. 17, Issue 6 (Nov. 2000), pp. 36-43  
Publisher: IEEE Computer Society Press  
6. Title: Software Reliability Engineering  
Author: John Musa Hardcover: 632 pages  
Publisher: AuthorHouse; 2nd edition (Sept. 20, 2004) ISBN: 1418493880  
■



# COURSE DESCRIPTOR

## Software Metrics

### Mätningar av programvara

7,5 ECTS credit points (7,5 högskolepoäng)

**Course code:** PA1303

**Educational level:** Basic level

**Course level:** G2F

**Field of education:** Technology

**Subject group:** Computer Technology

**Subject area:** Software Engineering

**Version:** 6

**Applies from:** 2008-01-21

**Approved:** 2009-11-01

**Replaces course descriptor approved:** 2001-08-14

#### 1 Course title and credit points

The course is titled Software Metrics/Mätningar av programvara and awards 7,5 credit points. One credit point (högskolepoäng) corresponds to one credit point in the European Credit Transfer System (ECTS).

#### 2 Decision and approval

This course is established by School of Computing 2008-01-01. The course descriptor was revised by School of Computing and applies from 2008-01-21. Replaces PAC003.

Reg no: TEK56-112/08

#### 3 Objectives

In any other engineering discipline, the term "best engineering practice" is repeatedly applied to all aspects of the development of a system. It might be a bridge, an automobile, or a large building. Best engineering practice embodies a long tradition of experimentation, analysis, and measurement. It is the application of scientific principles to the solution of a complex development project. At the core of the best engineering practice lies the measurement. For example, measurements in radar systems enable us to detect aircraft when direct vision is obscured. Medical system measurements enable doctors to diagnose specific illnesses. In software engineering, we need to understand and control the software projects. We need to know: What does each process cost?, How productive is the staff?, How can we improve?, etc. We need to measure to answer these questions. The objectives of this course are to provide the students the fundamentals of software measurement. They will gain the knowledge of how software measures can be used for controlling, managing and predicting the software development processes. They will gain basic understanding of the software measurement process and an awareness of problems related to applying software measurement and experiences in setting up measures and models.

#### 4 Content

The course comprises the following elements: •Fundamentals of measurement: The need for measurement, scope of software metrics; Measurement theory: scales, validation, and meaningfulness; Goal-Question-Metric paradigm; Measurement data collection and analysis; A classification of software metrics

•Software measures: Internal Product Attributes; External Product Attributes; Resource Measurement; ISO 9126 software product quality characteristics

•Software measurement process: Measurement process models; ISO/IEC 15939: Software Measurement Process; CMMI's measurement requirements

#### 5 Aims and learning outcomes

On completion of the course the student will be able to

- present and discuss the fundamentals of software measurement
- professionally present, argue, discuss how software measures can be used during the software processes and be able to professionally apply them
- discuss the state-of-the-art in software measurement process models and

CMMI's measurement requirements

- professionally present, argue, discuss the issues related to applying software measurement and how to set up measures and models

## 6 Generic skills

The following generic competences are trained in the course:

- Capacity for analysis and synthesis •Capacity for applying knowledge in practice •Basic general knowledge in the field of study

- Grounding in basic knowledge of the profession in

practice

## 7 Learning and teaching

Teaching comprises lectures and exercises where the students are expected to actively participate by discussing, questioning, and contributing their own experiences. In addition, compulsory practical assignments are set, which may be solved individually or in teams within a pre-defined time limit.

The course starts with an introductory lecture and then consists of a series of lectures where a number of topics (see Section 4) are introduced. During each lecture, additional time is spent on discussions and team assignments. Each student/ team must submit written assignments in addition to the exam, practical exercise and the experiment. By written assignments, the students will review and exercise the specific topics they learn during the lectures. Two of the assignments are individual and the other two are group assignments. In the practical exercise session, the students will form groups to solve a specific measurement related issue in a software company by using the knowledge they acquired during the previous lectures. In the experiment session, the students will work on a specific experiment on software measurement. The final exam will be an individual, open book exam.

The teaching language is English.

## 8 Assessment and grading

### *Examination of the course*

Code	Module	Credit	Grade
0805	Practical assignments	2 hp	F-A
0815	Practical exercise	1 hp	F-A
0825	Practical experiment	0.5 hp	F-A
0835	Final exam	4 hp	F-A

The course will be graded F Fail, FX Fail, E Sufficient, D Satisfactory, C Good, B Very good or A Excellent. Assessment of the students' performance is carried out by written tests, and presentation of compulsory assignments. The assignments are evaluated based on the correctness and quality of the assignments. The practical exercise and the experiment are evaluated based on the presence and active participation of the students. The total grade is based on a weighted average.

## 9 Course evaluation

The course coordinator is responsible for systematically gathering feedback from the students in course evaluations and making sure that the results of these feed back into the development of the course.

## 10 Prerequisites

The student must have a Bachelor in Science in Computer Science or Software Engineering

## 11 Field of education and subject area

The course is part of the field of education Technology and is included in the subject area Software Engineering.

## 12 Restrictions regarding degree

The course cannot form part of a degree with another course, the content of which completely or partly corresponds with the contents of this course.

## 13 Course literature and other teaching material

Main literature:

Software Metrics - A Rigorous & Practical Approach, 2nd edition

Authors: N. E. Fenton, S. L. Pfleeger

Publisher: International Thomson Computer Press Published: 1996

ISBN: 1-85032-275-9 Reference literature:

1. Software Measurement and Estimation: A Practical Approach

Authors: L.M. Laird, M.C. Brennan Publisher: IEEE Computer Society Press Published: 2006

ISBN: 0471676225

2. Practical Software Measurement Authors: J. McGarry et.al

Publisher: Addison-Wesley Publishing Company Published: 2001 , ISBN: 0-201-71516-3

3. Software Metrics: Establishing a Company-Wide Program

Authors: R.B Grady, D.L Caswell Publisher: Prentice Hall PTR Published: 1998,ISBN: 0-138-1844-7

4. Metrics and Models in Software Quality Engineering, Author: Stephan H. Kan

Publisher: Addison-Wesley Publishing Company Published: 1995, ISBN: 0-201-63339-6

■



## COURSE DESCRIPTOR

### Advanced Software Project Management

Avancerad projektstyrning inom programvarutillverkning

7,5 ECTS credit points (7,5 högskolepoäng)

**Course code:** PA2414  
**Educational level:** Advanced level  
**Course level:** A1N  
**Field of education:** Technology  
**Subject group:** Computer Technology

**Subject area:** Software Engineering  
**Version:** 6  
**Applies from:** 2011-08-29  
**Approved:** 2011-06-10

#### 1 Course title and credit points

The course is titled Advanced Software Project Management/Avancerad projektstyrning inom programvarutillverkning and awards 7,5 credit points. One credit point (högskolepoäng) corresponds to one credit point in the European Credit Transfer System (ECTS).

#### 2 Decision and approval

This course is established by School of Computing 2007-08-24. The course descriptor was revised by School of Computing and applies from 2011-08-29. Reg no TEK56-284/07

#### 3 Objectives

Most software is today developed in teams. It is therefore vital that software engineers have detailed knowledge and skills to manage and work effectively in project teams.

The objective with this course is to provide the participants with a strong theoretical foundation in the field of general project management, behavioral sciences and organizational studies in connection to software project management (SPM) issues. This course assumes that the participants already have practical knowledge from project participations.

#### 4 Content

The course is built around four blocks:

- Project management
- Organizational theory
- Leadership
- Behavioral sciences

Regarding Project Management (PM), the participants are introduced to the history of PM, concepts and applications, and PM in the context of software development, i.e. SPM.

Concerning organizational theory, the participants are introduced to both macro theories (institutional, evolutionary, networks, etc.) and micro theories (cognitive, motivation, group, etc.)

In the leadership block, topics such as leadership types, classifications and terminologies are introduced and discussed.

With respect to behavioral sciences the role of the individual in a project context is examined and theories covering, in particular, decision science (psychology, management, etc.) are introduced and their role in project management is discussed.

#### 5 Aims and learning outcomes

On completion of the course the student will be able to:

- Independently describe, and in a group, discuss the area of project management (history, standards, definitions etc.) and name a number of key issues.
- Independently describe, and in a group, discuss the area of organizational theory and how it applies to SPM.
- Independently describe and, in a group, discuss the area of behavioral sciences and their connection to

SPM.

- Independently describe and, in a group, discuss the role of the leader in different types of organizations.
- Independently describe, compare and contrast different leadership types such as e.g. situational leadership.
- Independently develop an understanding of the key problems and benefits associated with managing people.
- In a group compare and contrast the different methods and techniques used to assure the quality of a software product in a SPM.

## 6 Generic skills

The following generic skills are trained in the course:

- Planning and time management
- Critical thinking
- Information search
- Problem solving
- Analytical ability
- Team work

## 7 Learning and teaching

The course is structured around a number of lectures, guest lectures and workshops. The course starts with an introductory lesson and then consists of a series of lectures where a number of topics (see Section 4) are introduced to the participants. During each lecture additional time is spent on discussions and team assignments. Each participant must, individually, submit two written assignments in addition to the written exam.

The teaching language is English.

## 8 Assessment and grading

### *Examination of the course*

Code	Module	Credit	Grade
0710	Written examination	3.5 hp	U/G/VG
0720	Assignment 1	2 hp	U/G/VG
0730	Assignment 2	2 hp	U/G/VG

The course will be graded Fail (U), Pass (G) or Pass with Distinction (VG). The total grade is based on a weighted average (e.g. the written exam is worth 3.5/7.5th of the total grading). On request grades according to ECTS will be given.

## 9 Course evaluation

The course coordinator is responsible for systematically gathering feedback from the students in course evaluations and making sure that the results of these feed back into the development of the course.

## 10 Prerequisites

Bachelor of Science with a major in Computer Science or Software Engineering.

## 11 Field of education and subject area

The course is part of the field of education Technology and is included in the subject area Software Engineering. The course can also be included in the subject area Computer Science.

## 12 Restrictions regarding degree

The course cannot form part of a degree with another course, the content of which completely or partly corresponds with the contents of this course.

## 13 Course literature and other teaching material

1. IEEE Std. 1490-2003. Adoption of PMI Standard A Guide to the Project Management Body of Knowledge (PMBOK® Guide)

Author: IEEE Publisher: IEEE Published: 2003 ISBN: 1-930699-45-X

2. Guide to the Software Engineering Body of Knowledge (SWEBOOK), 2004 version Author: Tripp, L. L. et al. Publisher: IEEE Published: 2004 ISBN: 0769523307



3. Management of Organizational Behavior: Leading  
Human Resources, Author: Blanchard, K. and Hersey, P. , Publisher: Prentice Hall, Published: 2001  
ISBN: 0130175986
4. On Becoming a Leader: The Leadership Classic  
Author: Bennis, W. G.  
Publisher: Perseus Publishing  
Published: 2003  
ISBN: 0738208175
-



## COURSE DESCRIPTOR

### Applied Software Project Management

Praktisk projektstyrning inom programvarutillverkning

7,5 ECTS credit points (7,5 högskolepoäng)

**Course code:** PA2406

**Educational level:** Advanced level

**Course level:** A1N

**Field of education:** Technology

**Subject group:** Computer Technology

**Subject area:** Software Engineering

**Version:** 7

**Applies from:** 2012-03-26

**Approved:** 2012-02-10

#### 1 Course title and credit points

The course is titled Applied Software Project Management/Praktisk projektstyrning inom programvarutillverkning and awards 7,5 credit points. One credit point (högskolepoäng) corresponds to one credit point in the European Credit Transfer System (ECTS).

#### 2 Decision and approval

This course is established by School of Computing 2007-09-27. The course descriptor was revised by School of Computing and applies from 2012-03-26. Reg no: COM-560-0023-2012

#### 3 Objectives

Most software is today developed in teams. It is therefore vital that software engineers have knowledge in this particular area.

The objective with this course is to learn how to prepare (plan), execute and finalize a project. This includes, but is not limited to, steering and administrating a project from start to finish, applying methods and techniques for making sure the project ends in a successful manner and, additionally, understanding and learning how to interpret stakeholders' interest in a typical project.

#### 4 Content

The course comprises the following:

- An Introduction to Project Management: Motivation, goals, state-of-the-art, future research directions, and related areas.
- Project Life Cycle and Organization: Planning, design, implementation in project management.
- Product Management Processes: Groups, interactions and mapping
- Management Knowledge Areas: Integration, scope, time, cost, quality, human resources, communication, risk and procurement planning.

#### 5 Aims and learning outcomes

On completion of the course the student will be:

- able to describe the area of software project management (standards, key concepts, software quality definitions etc.) and name a number of key issues related to this area.
- able to identify project management tools suggesting likely candidates for different project types.
- define risk management issues in software project management.
- in a group, able to execute and finish a project with accompanying deliverables through involvement in a team project.
- independently able to define and categorize quality issues in software project management.
- independently able to explain the difference between traditional project management and contemporary software project management.
- in a group, able to prepare a project plan for a software project that includes estimates of size and effort, a schedule, resource allocation, configuration control, change management, and project risk identification and management.

- be able to conduct post-mortem analysis with accompanying process improvement suggestions.

## 6 Generic skills

The following generic skills are trained in the course:

- Ability to work in teams
- Planning and time management
- Ability to relate theory to practice
- Critical thinking
- Information search
- Problem solving and analytical ability

## 7 Learning and teaching

The participants will work in teams executing a project.

The course starts with an introductory lesson and then consists of a series of workshops and lessons that are conducted in parallel to the participants' projects. The teams must plan, execute and finalize a project.

The project has two team assignments, which are examined under the moment "project plan and execution" and one individual assignment examined under the moment "final project report". In the two team assignments, a number of artifacts must be delivered indicating that the team is following the previously established project plan. In the individual assignment the participant must individually perform an analysis of the project with an accompanying process improvement recommendation of the project.

The team assignments must be finalized before the individual assignment can begin.

## 8 Assessment and grading

### *Examination of the course*

Code	Module	Credit	Grade
1205	Project plan and execution	4 hp	U/G/VG
1215	Final project report	3.5 hp	U/G/VG

The course will be graded Fail (U), Pass (G) or Pass with Distinction (VG). The total grade is based on a weighted average.

On request grades according to ECTS will be given.

## 9 Course evaluation

The course coordinator is responsible for systematically gathering feedback from the students in course evaluations and making sure that the results of these feed back into the development of the course.

## 10 Prerequisites

The student must have a Bachelor of Science with a major in Computer Science or Software Engineering.

## 11 Field of education and subject area

The course is part of the field of education Technology and is included in the subject area Software Engineering.

## 12 Restrictions regarding degree

The course cannot form part of a degree with another course, the content of which completely or partly corresponds with the contents of this course.

## 13 Course literature and other teaching material Main literature

IEEE Std. 1490-2003. Adoption of PMI Standard A Guide to the Project Management Body of Knowledge (PMBOK® Guide)

Author: Project Management Institute Publisher: Project Management Institute Published: 2004,

Number of pages: 388 ISBN10: 193069945X

ISBN13: 9781930699458

---

### Reference literature

A Spiral Model of Software Development and Enhancement

Author: Boehm, B

Publisher: ACM SIGSOFT Software Engineering Notes

Published: 1986

■



## COURSE DESCRIPTOR

### Master's Thesis in Software Engineering (120 credits)

#### Masterarbete i programvaruteknik

30 ECTS credit points (30 högskolepoäng)

**Course code:** PA2503  
**Educational level:** Advanced level  
**Course level:** A2E  
**Field of education:** Technology  
**Subject group:** Computer Technology

**Subject area:** Software Engineering  
**Version:** 7  
**Applies from:** 2012-09-03  
**Approved:** Not approved

#### 1 Course title and credit points

The course is titled Master's Thesis in Software Engineering (120 credits)/Masterarbete i programvaruteknik and awards 30 credit points. One credit point (högskolepoäng) corresponds to one credit point in the European Credit Transfer System (ECTS).

#### 2 Decision and approval

This course is established by School of Computing 2010-06-02. The course descriptor was revised by School of Computing and applies from 2012-09-03. Reg no:  
Replaces PA2403

#### 3 Objectives

A Master's thesis is the culmination of several years of studies within a subject area and should show that the author has matured to a level where he/she can make a scientifically valid contribution to the knowledge within that area.

The objective of this course is that the participants should further develop and show the knowledge, understanding and skills needed to work independently as professionals within Software Engineering. This includes the ability to independently identify problems/challenges/inadequacies in state-of-the-art and/or in state-of-practice. Then subsequently plan, design, conduct, collect results, and perform analysis on results, while rigorously documenting all steps in a thesis. This also includes the ability to show extensive knowledge of research methodologies, their application in the form of rigorous scientific methods towards fulfilling the aim of advancing state-of-the-art and/or state-of-practice.

The research carried out should be in line with, and through the reporting exhibit, the core characteristics of scientific research.

**Research** can be defined as the search for knowledge or any systematic investigation to establish facts. The primary purpose for applied research (as opposed to basic research) is discovering, interpreting, and the development of methods/models/processes/techniques and systems for the advancement of Software Engineering. **Scientific research** relies on the application of the scientific method, a harnessing of curiosity.

In addition to being able to perform scientific research the student should be able to judge other scientific works in the same subject area, and give constructive feedback for the betterment of the research project under scrutiny.

The written report produced in the research project (called thesis from here on in) should be a detailed account of the research conducted, reflecting all steps undertaken in the research project as well as all parts of synthesis (analysis and conclusions). The value of the scientific research conducted can never exceed the level of clarity in which the work is reported, thus the thesis itself acts as the final testament in relation to the work performed. **Development.** The ability and aptitude to develop and engineer products/processes/systems in relation to humans' abilities and demands, and societies demands for economic, social and sustainable development is also central for a civil engineering degree. There is no specific demand for including an explicit development effort, however, this can be seen as a contribution. It is however important to realize that the development effort in itself cannot replace the demand on research,

using a scientific approach. That is, a development effort can be a part of thesis work, but not replace it. For example, developing a simulator to test an algorithm is fine, but the actual research is only enhanced by the development of the simulator. Another example is developing a quality standard (see e.g. ISO). The development of the quality standard can be a contribution, but does not replace the research that led to the development and contents of the standard.

#### **4 Content**

The course consists of two parts: a master thesis project and an opposition to a master thesis. Students work in pairs (only special circumstances allow individual work) on a project. Common to all projects are that they include four major phases: (1) Developing a thesis proposal, (2) Planning the project, (3) Conducting the project and documenting it in a written Master's thesis, and (4) Reporting on and presenting the thesis in a defense.

Throughout these phases an academic supervisor helps and guides the students. It should be observed that the timely execution, adherence to deadlines, and fulfillment of the quality requirements of the thesis project is the student's responsibility. The opposition consists of a thorough study of a thesis, critically analyzing it, writing an opponent report and acting as opponent when it is defended.

#### **5 Aims and learning outcomes**

On completion of this course the student should be able to:

- Plan, design, execute, and report a research project in a timely and independent manner.
- Show the ability to use one or combine several areas within Software Engineering and apply them to investigate and/or solve a problem in a rigorous and scientific manner following a defined research methodology. This includes, but is not limited to, the following sub steps:
  - a. identify and formulate a non-trivial problem/challenge and through this problem state research questions and/or hypothesis that is possible to answer/test,
  - b. describe in detail related work and the planned research's contribution in the context of other peer-reviewed research,
  - c. design and execute a study/investigation that is aimed at answering/testing the research question(s)/hypothesis,
  - d. weigh and compare different research methods to motivate the ones chosen for the research project at hand,
  - e. collect, analyze, and weigh results with the purpose of achieving a rigorous analysis thus obtaining synthesis, and ultimately provide an insightful and non-trivial answer to the research questions,
  - f. weigh the value of conclusions and to what extent they can be generalized
- Report the work described above (2) in a thesis utilizing appropriate scientific language: a. written completely in English (consistently using either US or UK),
  - b. correct use of scientific terms.
- Evaluate and review other scientific work in Software Engineering to the level of being able to: a. judge the relevance of the research questions posed,
  - b. judge the appropriateness of the research methodology applied, and rigor of a research design,
  - c. judge the execution, collection of results, and analysis,
  - d. judge the validity of conclusions.

#### **6 Generic skills**

The following generic skills are trained in the course:

- Capacity for applying knowledge in practice,
- Critical and self-critical abilities,
- Capacity for generating new ideas (creativity),
- Problem solving,
- Decision-making,
- Teamwork,
- Ability to work autonomously,

#### **7 Learning and teaching**

The course starts with introductory lectures. The remainder of the course consists of the planning, design, execution, and presentation of a Master's thesis project, which is independently managed by the students themselves but supervised by an academic supervisor (a member of the faculty with a Licentiate degree or higher in software engineering or computer science). Thesis projects are conducted in groups of two students or, in exceptional cases, by one student, what is appropriate is mandated by the examiner of the course. Before starting the thesis project, the group must refine the thesis topic into a thesis proposal, which will be reviewed quantitatively and qualitatively by the academic supervisor and faculty reviewer. The presentation of the thesis can be carried out no sooner than 15 weeks after the supervisor has submitted the approved version of the

proposal along with his/her proposal review as well as the review of the faculty reviewer to the examiner. During the course the group should attend at least five presentations of theses, excluding their own. The group should also act as opponents on one thesis and write a detailed opponent report.

In addition to the academic supervisor, a group may also have an additional advisor in industry. The academic supervisor helps and guides the group during the thesis project but it is up to the student to request guidance. The group is expected to inform the academic supervisor of the progress at least every three (3) weeks during the project.

In order to be allowed to present the thesis, the supervisor must have submitted his/her approval of the thesis (indicating that the supervisor is confident that the thesis will at least be graded Pass) to the examiner no less than two (2) weeks before the date of the presentation. The final, revised thesis should be sent to the examiner at the latest two weeks after the presentation. The examiner grades the thesis on the basis of his/her own review, taking into consideration the review of the final thesis conducted by the academic supervisor and the faculty reviewer.

In keeping with the study administrative rules of BTH:

- A student has three scheduled presentation opportunities, or at least 12 calendar months, (at least 2 are offered per year) to present a finished thesis from the time the thesis proposal is approved, failing to do so will give an F on the course.
- A student that has failed the course, for whatever reason, can re-register and redo the thesis project once. That is, a student has two separate chances to complete the thesis course and may change topic and supervisor.
- Deviations from these rules, given special circumstances, is only permitted at the discretion of the examiner.

The teaching language is English.

## 8 Assessment and grading

### *Examination of the course*

Code	Module	Credit	Grade
1010	Thesis proposal	2 hp	U/G
1020	Presentation/Defense	1 hp	U/G
1030	Thesis opposition	1 hp	U/G
1040	Thesis [1]	26 hp	F-A

<sup>1</sup> Determines the final grade for the course, which will only be issued when all components have been approved. The course will be graded F Fail, FX Fail, E Sufficient, D Satisfactory, C Good, B Very good or A Excellent. The course will be graded according to the ECTS scale A to F, when the supervisor has approved the thesis proposal, the course examiner has graded and approved the Thesis, Opponent report/Opposition and the Defense. The course supervisor and faculty reviewer recommend a grade to the examiner, however the final grade decision is up to the examiner alone.

## 9 Course evaluation

The course coordinator is responsible for systematically gathering feedback from the students in course evaluations and making sure that the results of these feed back into the development of the course.

## 10 Prerequisites

The student must have successfully completed the course PA2404 Research Methodology 7.5 ECTS (or equivalent) as well as 90 ECTS credit points in the area of Software Engineering

## 11 Field of education and subject area

The course is part of the field of education Technology and is included in the subject area Software Engineering.

## 12 Restrictions regarding degree

The course cannot form part of a degree with another course, the content of which completely or partly corresponds with the contents of this course.

## 13 Course literature and other teaching material Main literature

1. How to Write a Thesis, 2nd Edition, Author: Rowena Murray, Publisher: McGraw Hill, 2006, Number of pages: 320 ISBN13: 9780335219681, ISBN10: 0335219683

---

### Reference literature

1. Real World Research Author: Colin Robson Publisher: Blackwell Publ

Published: 2002, Number of pages: 528 ISBN10: 0631213058  
ISBN13: 9780631213055  
2. Experimentation in Software Engineering: An Introduction  
Author: Claes Wohlin, Per Runeson Publisher: Springer  
Published: 1999, Number of pages: 228 ISBN10: 0792386825  
ISBN13: 9780792386827  
■



# COURSE DESCRIPTOR

## Server Architectures

### Serverarkitekturer

7,5 ECTS credit points (7,5 högskolepoäng)

**Course code:** PA2514

**Educational level:** Advanced level

**Course level:** A1N

**Field of education:** Technology

**Subject group:** Computer Technology

**Subject area:** Computer Science, Software Engineering

**Version:** 1

**Applies from:** 2013-09-02

**Approved:** Not approved

#### 1 Course title and credit points

The course is titled Server Architectures/Serverarkitekturer and awards 7,5 credit points. One credit point (högskolepoäng) corresponds to one credit point in the European Credit Transfer System (ECTS).

#### 2 Decision and approval

This course is established by School of Computing . The course descriptor is approved by School of Computing and applies from 2013-09-02.

#### 3 Objectives

When engineering a larger single or distributed computer system, a number of, possibly conflicting, requirements needs to be addressed.

In the design of large computer systems, a number of hardware & software components is typically selected and integrated so that an acceptable balance between conflicting requirements can be met. This demands deep technical knowledge of the components themselves and the aimed non-functional properties.

A typical use case for both reliable and scalable infrastructures is the realization of a server system. Modern server systems are no longer implemented by one single machine, but utilize cluster computing techniques. Cluster systems realize a comparatively inexpensive and scalable computing platform, fulfilling both availability and high performance demands for distributed applications. The major topic of the course are cluster systems, including the trade-offs and conflicts between design time quality attributes (e.g. maintainability, reusability) on the one hand and run time quality attributes (e.g. performance, reliability) on the other hand.

#### 4 Content

The course includes discussion of cluster hardware and software, typical configurations, programming models, quality attributes, their conflicts and their realization.

#### 5 Aims and learning outcomes

On completion of the course the individual student will be able to:

- Architect a larger computer system while balancing conflicting requirements
- Identify the performance and scalability limitations of different cluster architectures
- Select an appropriate system architecture based on given requirements

#### 6 Generic skills

The following generic competences are trained in the course:

- Capacity for analysis and synthesis
- Capacity for applying knowledge in practice
- Information management skills (ability to retrieve and analyze information from different sources)
- Critical and self-critical abilities
- Problem solving
- Concern for quality



## 7 Learning and teaching

The course comprises a number of lectures, covering the theoretical aspects of server architectures and clusters with regard to performance and availability. The content of the lectures is subject of an oral exam. The course also contains one mandatory laboratory exercise, where conflicting application requirements needs to be addressed. As third major building block, students need to write an extensive report on a topic relevant to the course.

Language of instruction will be English. The teaching language is English.

## 8 Assessment and grading

### *Examination of the course*

Code	Module	Credit	Grade
	Oral examination <sup>[1]</sup>	1.5 hp	A-F
	Laboration	1.5 hp	G-U
	Report	4.5 hp	G-U

<sup>1</sup> Determines the final grade for the course, which will only be issued when all components have been approved. The course will be graded A Excellent, B Very good, C Good, D Satisfactory, E Sufficient, FX Insufficient, supplementation required, F Fail.

## 9 Course evaluation

The course coordinator is responsible for systematically gathering feedback from the students in course evaluations and making sure that the results of these feed back into the development of the course.

## 10 Prerequisites

The prerequisites for the course is that the students should have passed the following courses: Programming, 7.5 ECTS, Datastructures and Algorithms, 7.5 ECTS, Operating Systems, 7.5 ECTS and Computer Networks, 7.5 ECTS

## 11 Field of education and subject area

The course is part of the field of education Technology and is included in the subject area Computer Science and the subject area Software Engineering.

## 12 Restrictions regarding degree

The course cannot form part of a degree with another course, the content of which completely or partly corresponds with the contents of this course.

## 13 Additional information

The programming language for the laboratory exercise is C/C++ or Java, students are expected to have learnt these programming languages earlier, or being able to learn them on their own during the course. Replaces PA2416 and PA2508.

## 14 Course literature and other teaching material

Main literature

1. Server Architectures: Multiprocessors, Clusters, Parallel Systems, Web Servers, and Storage Solutions

Author: Rene J. Chevanche Publisher: Digital Press Published: 2005

ISBN: 1-55558-333-4

2. High Availability and Disaster Recovery: Concepts, Design, Implementation Author: Klaus Schmidt

Publisher: Springer Published: 2006 ISBN: 3-540-24460-3 Reference literature

1. High Performance Cluster Computing: Architectures and Systems, Vol. 1 Author: Rajkumar Buyya

Publisher: Prentice Hall PTR Published: 1999

ISBN: 0130137847

2. Blueprints for High Availability Author: Hal Stern, Evan Marcus Publisher: J Wiley & Sons  
Published: 2003

Edition: 2

ISBN: 0471430269

3. C Programming Language

Author: Brian W. Kernighan, Dennis Ritchie

Publisher: Prentice Hall PTR

Published: 1988

Edition: 2

ISBN: 0131103628

4. Mpi: The Complete Reference (Vol. 1) - 2nd Edition: Volume 1 - The Mpi Core

Author: Marc Snir et al. Publisher: MIT Press Published: 1998 Edition: 2

ISBN: 0262692155

5. Mpi: The Complete Reference (Vol. 2): Volume 2 - The Mpi-2 Extensions

Author: William Gropp et al. Publisher: MIT Press Published: 1998

ISBN: 0262571234

■



## COURSE DESCRIPTOR

---

### Global Software Engineering

#### Global programvaruteknik

7,5 ECTS credit points (7,5 högskolepoäng)

---

**Course code:** PA2518

**Educational level:** Advanced level

**Course level:** A1N

**Field of education:** Technology

**Subject group:** Computer Technology

**Subject area:** Software Engineering

**Version:** 1

**Applies from:** 2013-09-02

**Approved:** 2013-06-19

#### 1 Course title and credit points

The course is titled Global Software Engineering/Global programvaruteknik and awards 7,5 credit points. One credit point (högskolepoäng) corresponds to one credit point in the European Credit Transfer System (ECTS).

#### 2 Decision and approval

This course is established by School of Computing 2013-06-19. The course descriptor is approved by School of Computing and applies from 2013-09-02. Reg.no: BTH 4.1.1-0461-2013

#### 3 Objectives

The objective of this course is therefore to provide knowledge, competence and practical experience regarding communication, cooperation and coordination of software engineering activities given the global setting. The student will learn how to communicate on a global network, in a global team, and interpret and sensitively exploit diversity in their professional (and personal) life.

#### 4 Content

The course investigates and discusses two main GSE scenarios: transfers and distribution of work. The lectures cover such topics as offshore benefits, drivers and locations of offshore development, GSE project management and coordination, teamwork, communication and tools in distributed projects, and the importance of human factors, trust and cultural diversity. The challenges of global software engineering are illustrated through multiple case studies.

#### 5 Aims and learning outcomes

##### *Knowledge and understanding:*

After the course the student will:

- Be able to account for the complexity of global software engineering.
- Describe different scenarios that are possible

through the GSEs.

- Be able to account for the special circumstances at work in international teams.

##### *Skills and Abilities:*

After the course the student will:

- Be able to plan a global software projects.

##### *Values and attitudes:*

After the course the student will:

- Be able to select and evaluate methods and tools for communication, collaboration and coordination in global projects.
- Be able to evaluate the applicability of methods and approaches in software engineering for GSE scenarios.

#### 6 Generic skills

The following generic skills are trained in the course:

- problem-solving

- decision-making
- teamwork
- computer-mediated communication
- understanding of cultural diversity

## 7 Learning and teaching

The course is structured around a number of lectures and workshops. The course starts with an introductory lesson and then consists of a series of lectures where a number of topics (see Section 4) are introduced to the participants. There will be one project assignment given during the course that aims to analyze a given case study. In this assignments the students are expected to demonstrate abilities to plan a highly distributed software development project.

The course will be concluded with a written exam. The teaching language is English.

## 8 Assessment and grading

### *Examination of the course*

Code	Module	Credit	Grade
1310	Assignment	3 hp	A-F
1320	Written exam	4.5 hp	A-F

The course will be graded A Excellent, B Very good, C Good, D Satisfactory, E Sufficient, FX Insufficient, supplementation required, F Fail. The total grade is based on a weighted average. Rounding occurs upwards.

## 9 Course evaluation

The course coordinator is responsible for systematically gathering feedback from the students in course evaluations and making sure that the results of these feed back into the development of the course.

## 10 Prerequisites

Bachelor of Science with a major in Computer Science or Software Engineering

## 11 Field of education and subject area

The course is part of the field of education Technology and is included in the subject area Software Engineering.

## 12 Restrictions regarding degree

The course cannot form part of a degree with another course, the content of which completely or partly corresponds with the contents of this course.

## 13 Additional information

Replaces PA2411.

## 14 Course literature and other teaching material

Title: Global Software Teams: Collaborating Across Borders and Time Zones

Author: Erran Carmel Publisher: Prentice Hall PTR

Published: 1999, Number of pages: 292 ISBN10: 013924218X

ISBN13: 9780139242182

■



## COURSE DESCRIPTOR

---

### Software Architectures and Quality

#### Programvaruarkitektur och kvalitet

7,5 ECTS credit points (7,5 högskolepoäng)

---

**Course code:** PA1410

**Educational level:** Basic level

**Course level:** G1F

**Field of education:** Technology

**Subject group:** Computer Technology

**Subject area:** Software Engineering

**Version:** 1

**Applies from:** 2013-09-02

**Approved:** 2013-06-25

#### 1 Course title and credit points

The course is titled Software Architectures and Quality/Programvaruarkitektur och kvalitet and awards 7,5 credit points. One credit point (högskolepoäng) corresponds to one credit point in the European Credit Transfer System (ECTS).

#### 2 Decision and approval

This course is established by School of Computing 2013-06-25. The course descriptor is approved by School of Computing and applies from 2013-09-02. Reg.no: BTH 4.1.1-0290-2013

#### 3 Objectives

Software Architectures is an important technical concepts in modern large-scale software development that serves multiple purposes; planning development resources based on the architecture, analyzing the problem domain from a number of perspectives with the help of architecture, and one abstracts large amounts of information to obtain a useful overview by architecture. In addition, the architecture, and how one chooses to construct the architecture, a key component in planning and achieving a certain level of quality in a system, thereby determining how successfully the system will be.

In this course students are expected to acquire detailed knowledge of software architectures and software quality and, in particular, how the latter influenced by the former.

Furthermore, the student acquire an understanding of how to konstruerar a software architecture based on modern methods and ideas such as design patterns, object-oriented frameworks, and component-based software technology that takes into account the intended product, the surrounding technology and the development organization in a way that creates long-lasting and durable system with a planned and predictable level of quality.

#### 4 Content

The course comprises the following elements:

- Quality aspects in software and software architecture.
- Architectural styles, languages and patterns.
- Methods for architectural design and evaluation.
- Component-based software engineering.

#### 5 Aims and learning outcomes

##### *Knowledge and Understanding*

After the course the student will:

- Be able to name and describe a number of concepts and key components in the area (standards, architectural styles, languages and patterns; methods for architecture design and evaluation, and component-based software engineering).
- Be able to describe different aspects of quality in software and software architectures and how these are realized in quantifiable goals.

### ***Skills and Abilities***

After the course the student will:

- Carefully and with an attention to detail to create and document a software architecture consisting of multiple views and handles several different types of affairs.
- Independent, both on a theoretical level and in practice, be able to choose from a number of architectural styles, languages and designs depending on the requirements of the image and differentiate between them.

### ***Judgement and approach***

After the course the student will:

- Be able to discuss how the design of an architecture influenced by the surrounding technology and the developing organization for durability and planned and predictable level of quality.

## **6 Generic skills**

The following generic skills are trained in the course:

- Planning and time management
- Critical thinking
- Teamwork
- Problem solving
- Analytical ability

## **7 Learning and teaching**

The course is organized around a number of lectures where the students are expected to actively participate by discussing, questioning, and contributing their own experiences. The lectures are given early in the course, in order to provide a solid foundation for the subsequent assignments. The assignments are constructed to help the students reflect upon previous experiences, literature and research articles, and relate these with each other. This gives a mixture of state-of-the-art as described in research articles, and state-of-practice as evidenced by personal experiences, the course literature, and industry studies. The students are also expected to reflect upon what can be done better, i.e. process improvement.

The teaching language is English.

## **8 Assessment and grading**

### ***Examination of the course***

Code	Module	Credit	Grade
1310	Assignment 1	1.5 hp	A-F
1320	Assignment 2	1.5 hp	A-F
1330	Assignment 3	1.5 hp	A-F
1340	Assignment 4	1.5 hp	A-F
1350	Assignment 5	1.5 hp	A-F

The course will be graded A Excellent, B Very good, C Good, D Satisfactory, E Sufficient, FX Insufficient, supplementation required, F Fail. The examination consists of a number of tasks, including at least one individual, where the unweighted and rounded average gives the course final grade.

## **9 Course evaluation**

The course coordinator is responsible for systematically gathering feedback from the students in course evaluations and making sure that the results of these feed back into the development of the course.

## **10 Prerequisites**

The student must have successfully completed a total of 30 ECTS of the following courses: Object-oriented Systems Development, 7.5 ECTS, Datastructures and Algorithms, 7.5 ECTS, Database Systems, 7.5 ECTS, Computer Networks, 7.5 ECTS, Real-Time Systems, 7.5 ECTS, Operating Systems and Distributed Systems, 7.5 ECTS

## **11 Field of education and subject area**

The course is part of the field of education Technology and is included in the subject area Software Engineering.

## **12 Restrictions regarding degree**

The course cannot form part of a degree with another course, the content of which completely or partly corresponds with the contents of this course.

## **13 Additional information**

Replaces PA1308 and PA1309.

## **14 Course literature and other teaching material**

Applied Software Architecture Authors: Hofmeister, Nord

Publisher: Addison-Wesley Professional Published: 2000, Number of pages: 400 ISBN10: 0201325713  
ISBN13: 9780201325713 Reference literature

1. Software Architecture in Practice, Second Edition Authors: Bass, L., Clements, P., and Kazman, R. Publisher:  
Addison-Wesley, Published: 2003, Number of pages: 560 ISBN10: 0321154959, ISBN13: 9780321154958

2. Zen and the art of Motorcycle Maintenance Authors: R.M. Pirsig

Publisher: William Morrow Published: 1974, Number of Pages: 412 ISBN-10: 0688002307  
ISBN-13: 978-0688002305

■



## COURSE DESCRIPTOR

### Software Quality Management

#### Kvalitetsstyrning

7,5 ECTS credit points (7,5 högskolepoäng)

**Course code:** PA2408

**Educational level:** Advanced level

**Course level:** A1N

**Field of education:** Technology

**Subject group:** Computer Technology

**Subject area:** Software Engineering

**Version:** 7

**Applies from:** 2007-07-01

**Approved:** 2009-11-01

**Replaces course descriptor approved:** 2001-08-14

#### 1 Course title and credit points

The course is titled Software Quality Management/Kvalitetsstyrning and awards 7,5 credit points. One credit point (högskolepoäng) corresponds to one credit point in the European Credit Transfer System (ECTS).

#### 2 Decision and approval

Decision regarding the course and approval of this course descriptor have been made by 2007-01-30. This revision applies from 2007-07-01.

Replaces PAD006. Reg no: TEK56-296/07

#### 3 Objectives

Quality is an elusive concept that affect and is affected by the whole process of product development. In software development, which is a complex undertaking, quality thinking is particularly important since software quality is affected by every process, support function and sub-product within the development effort. The objective with this course is that the participants should get a deeper understanding of what quality is and what the challenges are to achieving high quality. The participants in the course are expected to get a general understanding of the concept of quality, and are expected to develop an understanding of the challenges faced especially with software quality. The participants should get an understanding of software quality management and its role in relation to other parts of software development, as well as of the activities, techniques, and models that are central for software quality assurance. The overall objective for the course is to educate participants to an awareness of the current state-of-the-art to be able to improve the state-of-the-practice in the software industry.

#### 4 Content

The following general themes are covered:

- Philosophy of Quality – what is software quality and what benefits and shortcomings exist?
- Quality Issues – what is quality and how does quality management differ in different industries, application domains, market situations, etc.?
- Software Process Improvement – what is a software process and how is it possible to work with software process improvement?
- Quality Management Systems and Models – what different quality management systems exist, what are their purposes, what are their similarities and weaknesses, and what are their pros and cons?
- Defect Prevention – what is defect prevention, what are the pros and cons, and how can it be utilized in different phases of software development?
- Quality Attributes – which quality attributes should be cared for and which perspectives (e.g. developers, managers) care for different attributes?
- Software Processes – what software processes exist and how do they relate to different quality activities, techniques, models, etc.?
- Software Engineering Culture and Ethics



## 5 Aims and learning outcomes

On completion of the course the student will:

- understand the concept and the application of quality, as well as be able to position quality within software development
- have basic understanding about quality and be able to discuss software quality and how it affects software development for different scenarios
- have a basic awareness of state-of-the-art within software quality management and how it relates to state-of-the practice
- have deep knowledge about methods, models, techniques, etc. central for software quality management, position them in relation to each other as well as argue about their strengths, weaknesses, and application areas
- have deep understanding about different

perspectives of quality and how to make trade-offs among (often conflicting) quality issues

- be able to relate own experience to theories put forward in literature in a mature way
- be able to professionally present, argue, discuss issues related to software quality, and defend personal viewpoints

## 6 Generic skills

The following generic skills are trained in the course:

Ability to work independently and in teams, planning and time management, ability to relate theory and practice, ability to understand different cultures and work in an international context, critical thinking, information search, problem solving, responsibility.

## 7 Learning and teaching

The main focus of the course is that the participants themselves shall seek for information. A few lectures are given and the aim with these is to discuss previous and upcoming assignments, and introduce upcoming topics of the course. The lectures are to a large extent possible to be adjusted by the participants since the lectures are based on interesting discussions from previous parts of the course, as well as explicit wishes from the participants.

Besides the lectures, the course is based on the participants own knowledge from experience and reading literature. This is used as input for the different assignments and meetings throughout the course. The assignments and meetings are performed through reports, discussion seminars, argumentation seminars, meetings, lectures, presentations, etc. This way of performing the teaching requires very active participants where they should relate the topic in focus to his/her knowledge, experience, and viewpoints, and present his/her view. At the same time, the teacher contributes by being a moderator and giving feedback by relating to examples from industry and research in the area.

The idea behind the course is to start wide and discuss overall quality concepts to get a common vocabulary and understanding, and then focus more and more on different concepts and activities within quality management throughout the course.

The language of instruction is English.

The teaching language is partly, or fully, English

## 8 Assessment and grading

### *Examination of the course*

Code	Module	Credit	Grade
0710	Assignment 1	1.5 hp	U/G/VG
0720	Assignment 2	1.5 hp	U/G/VG
0730	Assignment 3	1.5 hp	U/G/VG
0740	Assignment 4	1.5 hp	U/G/VG
0750	Assignment 5	1.5 hp	U/G/VG

The course will be graded Fail (U), Pass (G) or Pass with Distinction (VG).The examination in the course is continuous.

The participants are assessed according to their performance and activity on seminars, lectures, etc. and their performance on assignments that are handed in.

On request grades according to ECTS will be given.

## 9 Course evaluation

The course coordinator is responsible for systematically gathering feedback from the students in course evaluations and making sure that the results of these feed back into the development of the course.

**10 Prerequisites**

Prerequisite courses for this course are: Passed courses: PA1304-Large Team Software Engineering Project.

**11 Field of education and subject area**

The course is part of the field of education Technology and is included in the subject area Software Engineering.

**12 Restrictions regarding degree**

The course cannot form part of a degree with another course, the content of which completely or partly corresponds with the contents of this course.

**13 Course literature and other teaching material**

Main literature

1. Quality from Customer Needs to Customer Satisfaction

Authors: Bergman, B. & Klefsjö, B. Publisher: Studentlitteratur Published: 2003, Number of pages: 606 ISBN 91-44-04166-7

2. Accelerating Process Improvement Using Agile Techniques

Author: Jacobs, Deb

Publisher: Auerbach Publications Published: 2005, Number of pages: 390 ISBN: 0-8493-3796-8

3. Additional material published at the course webpage and from the library (including research databases, etc.)

Reference literature

Software Process Improvement – Practical guidelines for Business success

Author: Zahran, Sami Publisher: Addison-Wesley

Published: 1997, Number of pages: 447 ISBN 0-201-17782-X

■



**Blekinge Institute of Technology**  
School of Computing

## COURSE DESCRIPTOR

# Large Scale Requirements Engineering

## Storskalig kravhantering

7,5 ECTS credit points (7,5 högskolepoäng)

**Course code:** PA2417

**Educational level:** Advanced level

**Course level:** A1F

**Field of education:** Technology

**Subject group:** Computer Technology

**Subject area:** The course is not part of a main field of study at BTH.

**Version:** 5

**Applies from:** 2008-09-01

**Approved:** 2009-11-01

### 1 Course title and credit points

The course is titled Large Scale Requirements Engineering/Storskalig kravhantering and awards 7,5 credit points. One credit point (högskolepoäng) corresponds to one credit point in the European Credit Transfer System (ECTS).

### 2 Decision and approval

This course is established by School of Computing 2008-09-01. The course descriptor was revised by School of Computing and applies from 2008-09-01. Regno: TEK56-254/08

### 3 Objectives

One obstacle in software development is making sure that you are developing the right system, i.e. Requirements Engineering. In this course the students is expected to gain an understanding of how to collect relevant requirements from relevant sources before software development commences. In this course we focus on the problems that arise when requirements engineering is conducted in a fast-paced, cost-sensitive reality. In the course we discuss problems related to large scale requirements engineering and market driven requirements engineering. Issues such as continuous requirements engineering, requirements management, and requirements engineering process improvement, and technical product management are discussed and related to current industry practice.

### 4 Content

This course covers the problems that arise when requirements engineering is conducted in a fast-paced, cost-sensitive industry reality. The following topics are included in the course:

- Challenges in Large Scale Requirements Engineering
- Processes for Large Scale Requirements Engineering
- Market-Driven Requirements Engineering
- Technical Product Management
- Managing Large amounts of Requirements
- Release Planning for multiple releases
- Quality Assurance of Requirements

### 5 Aims and learning outcomes

On completion of the course the student will:

- Thoroughly understand the challenges involved in market-driven requirements engineering and large scale requirements engineering
- Be able to select and motivate appropriate techniques for market-driven requirements engineering
- In a controlled way be able to apply appropriate techniques in a large scale market-driven requirements engineering context
- Be able to quickly and accurately manage large amounts of requirements and a large influx of requirements.
- Be able to plan relevant release packages for several releases of a software product.

- Be able to create a sufficient requirements engineering process for managing requirements in a large organisation from requirements' entry into the organisation to requirements' deployment in a product.

## 6 Generic skills

The following generic skills are trained in the course:

- Information Retrieval: Finding, reading, and assessing research articles
- Teamwork
- Personal planning

## 7 Learning and teaching

The course is organized around a number of lectures where the students are expected to actively participate by discussing, questioning, and contributing their own experiences. The lectures are given early in the course, in order to provide a solid foundation for the subsequent assignments. The assignments are constructed to help the students reflect upon previous experiences, literature and research articles, and relate these with each other.

This gives a mixture of state-of-the-art as described in research articles, and state-of-practice as evidenced by personal experiences, the course literature, and industry studies. The students are also expected to reflect upon what can be done better when processing requirements, i.e. process improvement.

The teaching language is English.

## 8 Assessment and grading

### *Examination of the course*

Code	Module	Credit	Grade
0810	MDRE Process	2 hp	F-A
0820	Release planning	1 hp	F-A
0830	Reflecting report	2 hp	F-A
0840	Written examination	2.5 hp	F-A

The course will be graded F Fail, FX Fail, E Sufficient, D Satisfactory, C Good, B Very good or A Excellent. The examination consists of several assignments and reports which are weighed together for a course total.

## 9 Course evaluation

The course coordinator is responsible for systematically gathering feedback from the students in course evaluations and making sure that the results of these feed back into the development of the course.

## 10 Prerequisites

The student must have successfully completed the course Practical Requirements Engineering or Requirements Engineering, 7.5 ECTS

## 11 Field of education and subject area

The course is part of the field of education Technology and is not part of a main field of study at BTH.

## 12 Restrictions regarding degree

The course cannot form part of a degree with another course, the content of which completely or partly corresponds with the contents of this course.

## 13 Course literature and other teaching material

Main literature

1. Just Enough Requirements Management - Where Software Development Meets Marketing

Author: A. Davis Publisher: Dorset House

Published: 2005, Number of pages: 240 ISBN: 0932633641

Reference literature

1. Software Requirements Author: K.E. Wiegers Publisher: Microsoft Press

Published: 1999, Number of pages: 350 ISBN: 0735606315

2. Mastering the Requirements Process

Authors: S. Robertson and J. Robertson Publisher: Addison-Wesley Published: 1999, Number of pages: 400 ISBN: 0201360462

3. Requirements Engineering - Processes and Techniques

Authors: G. Kotonya and I. Sommerville. Publisher: John Wiley & Sons Ltd Published: 1998, Number of pages: 400 ISBN: 0471982088

4. Software Requirements - Styles and Techniques Author: S. Lauesen

Publisher: Addison-Wesley Published: 2002, Number of pages: 600 ISBN: 0201745704

■



## COURSE DESCRIPTOR

---

### Software Security

### Programvarusäkerhet

7,5 ECTS credit points (7,5 högskolepoäng)

---

**Course code:** DV2546

**Educational level:** Advanced level

**Course level:** A1N

**Field of education:** Technology

**Subject group:** Computer Technology

**Subject area:** Computer Science

**Version:** 1

**Applies from:** 2013-09-02

**Approved:** 2013-06-17

#### 1 Course title and credit points

The course is titled Software Security/Programvarusäkerhet and awards 7,5 credit points. One credit point (högskolepoäng) corresponds to one credit point in the European Credit Transfer System (ECTS).

#### 2 Decision and approval

This course is established by School of Computing 2013-06-17. The course descriptor is approved by School of Computing and applies from 2013-09-02. Reg.no: BTH 4.1.1-0442-2013

#### 3 Objectives

The main objective of this course is to teach students to understand and how to address various software security problems in a secure and controlled environment. During this course the students will gain knowledge (both theoretical and practical) in various kinds of software security problems, and techniques that could be used to protect the software from security threats. The students will also learn to understand the "modus operandi" of adversaries; which could be used for increasing software dependability.

#### 4 Content

The course comprises the following:

- Software security background: historical overview, why software needs to be protected, traditional techniques used.
- Detailed analysis of different groups of software vulnerabilities, their characteristics, how adversaries can exploit them, and how to protect against them.
- Specific problems relating to software security within a Web context in terms of threats and countermeasures.
- Source code analysis, different methods used, and introduction to existing tools.
- Software security research: motivation, goals, state-of-the-art, and related areas.

#### 5 Aims and learning outcomes

##### *Knowledge and understanding*

On completion of the course the student will:

- be able to individually reason about software security problems and protection techniques on both an abstract and a more technically advanced level.
- be able to individually explain how software exploitation techniques, used by adversaries, function and how to protect against them.

##### *Skills and abilities*

On completion of the course the student will:

- be able to individually review executing software systems and its source code in search for security flaws.
- be able to individually correctly address identified common security flaws relating to software in both web applications and client/server systems.

- use the repositories of vulnerabilities to investigate and keep updated about current threats.

## 6 Generic skills

The following generic competences are trained in the course:

- Problem solving
- Planning and time management
- Information search

## 7 Learning and teaching

The course consists of:

- Lectures where the students are introduced to theories within a software security context
- Seminars where the students in groups implement the theories, resulting in a more profound understanding of core concepts
- Assignments with tasks about source code analysis, binary file analysis, web security and client-server security problems.

The teaching language is English.

## 8 Assessment and grading

### *Examination of the course*

Code	Module	Credit	Grade
1310	Web-/Client server	1.5 hp	G-U
1320	Source code analysis	3 hp	A-F
1330	Binary file analysis	1.5 hp	A-F
1340	Identification and management of software vulnerabilities	1.5 hp	A-F

The course will be graded A Excellent, B Very good, C Good, D Satisfactory, E Sufficient, FX Insufficient, supplementation required, F Fail.

## 9 Course evaluation

The course coordinator is responsible for systematically gathering feedback from the students in course evaluations and making sure that the results of these feed back into the development of the course.

## 10 Prerequisites

## 11 Field of education and subject area

The course is part of the field of education Technology and is included in the subject area Computer Science.

## 12 Restrictions regarding degree

The course cannot form part of a degree with another course, the content of which completely or partly corresponds with the contents of this course.

## 13 Additional information

Replaces DV2409 and DV2513.

## 14 Course literature and other teaching material

1. Gray Hat Hacking, Third Edition Reviews Author: Harris, S., Harper, A., Eagle, C., & Ness, J. . Publisher: McGraw-Hill  
Published: 2011
2. The Web Application Hacker's handbook Author: Dafydd Stuttard, Marcus Pinto Publisher: John Wiley & Sons  
Published: 2011

■



# COURSE DESCRIPTOR

## Performance Optimization

### Prestandaoptimering

7,5 ECTS credit points (7,5 högskolepoäng)

**Course code:** DV1307

**Educational level:** Basic level

**Course level:** G1F

**Field of education:** Technology

**Subject group:** Computer Technology  
06-15

**Subject area:** Computer Science

**Version:** 9

**Applies from:** 2007-03-06

**Approved:** 2009-11-01

**Replaces course descriptor approved:** 2006-

#### 1 Course title and credit points

The course is titled Performance Optimization/Prestandaoptimering and awards 7,5 credit points. One credit point (högskolepoäng) corresponds to one credit point in the European Credit Transfer System (ECTS).

#### 2 Decision and approval

Decision regarding the course and approval of this course descriptor have been made by 2007-03-06. This revision applies from 2007-03-06.

#### 3 Objectives

Performance is a key aspect in any software, and particularly important in a great computer game. In order to develop good programs with high performance, it is essential to students develop a good understanding of different methods and techniques to analyse and optimize the performance for a computer program.

#### 4 Content

The course includes the following elements:

- overview of which factors that effect performance in a computer system
- the hardware capabilities and limitations
- methods for performance measurement and instrumentation
- methods to analyse performance in a computer system
- high-level and low-level optimizations
- compiler's effect on performance
- assembler programming
- introduction to multiprocessors and parallel programming.
- independently be able to measure and analyse the performance of a program
- independently identify performance problems and implement appropriate optimizations
- in detail explain and independently put into practice different techniques to improve performance of program
- overall be able to explain structure and function of multiprocessors, basic ability to independently develop parallel programs.

#### 5 Aims and learning outcomes

On completion of course the student will:

- overall be able to describe how a modern computer system is constructed and how it affects performance

Code	Module	Credit	Grade
0710	Written examination	3 hp	U/G/VG

0720 Laboration	1.5 hp	U/G
0730 Project, assembler	1.5 hp	U/G
0740 Project, parallel programming	1.5 hp	U/G

-----  
The course will be graded Fail (U), Pass (G) or Pass with Distinction (VG).  
On request grades according to ECTS will be given.

## 6 Generic skills

The following general skills are trained in the course:

- Ability for analysis and synthesis
- Ability to apply knowledge in practice
- General knowledge in main field of study
- Ability to work independently and in group
- Ability to plan and carry out work within given time frames.

## 7 Learning and teaching

The theoretical basics in the course is presented in lectures and / or exercises The student is also expected to independently gain theoretical knowledge through independent study of relevant literature.

Theoretical knowledge is then applied practical both in teacher-led mandatory laboratory exercises and partly in project assignments carried out individually or in groups within a given time frame.

The teaching takes place in Swedish.

The teaching language is Swedish.

## 8 Assessment and grading

*Examination of the course*

## 9 Course evaluation

The course coordinator is responsible for systematically gathering feedback from the students in course evaluations and making sure that the results of these feed back into the development of the course.

## 10 Prerequisites

Admission to the course requires completed course in Programming, 15 ECTS credits and Datastructures and Algorithms, 7,5 ECTS credits.

## 11 Field of education and subject area

The course is part of the field of education Technology and is included in the subject area Computer Science. The course can also be included in the subject area Software Engineering.

## 12 Restrictions regarding degree

The course cannot form part of a degree with another course, the content of which completely or partly corresponds with the contents of this course.

## 13 Course literature and other teaching material

Main literature

Title: The Software Optimization Cookbook High Performance Recipes, 2nd edition

Author: Kevin Smith, Richard Gerber, X. Tian, Aart Bik

Publisher: Intel Press Published: 2005 ISBN: 0976483211

Material from the school. Reference literature

Title: Performance Solutions: A Practical Guide to Creating Responsive, Scalable Software

Author: C.U. Smith & L.G. Williams Publisher: Addison-Wesley Published: 2001

ISBN: 0201722291

Title: Sun Performance and Tuning: Java and the Internet, 2nd Edition

Author: A. Cockcroft Publisher: Prentice-Hall Published: 1998

ISBN: 0130952494

Title: High Performance Computing Author: K. Dowd

Publisher: O'Reilly & Associates Published: 1993

ISBN: 1-56592-032-5

Title: System Performance Tuning, 2nd Edition Author: Gian-Paolo D. Musumeci & Mike Loukides

Publisher: O'Reilly Media Inc.

Published: 2002

ISBN: 059600284X



Title: Computer Architecture – A Quantitative Approach, 4th edition  
Author: J.L. Hennessy & D.A. Patterson  
Publisher: Morgan Kaufmann Publishers Published: 2006  
ISBN: 978-0-12-370490-0  
Title: The Practical Performance Analyst Author: N.J. Gunther  
Publisher: Authors Choice Press Published: 2000  
ISBN: 059512674X





## COURSE DESCRIPTOR

### Programming in UNIX Environment

#### Programmering i UNIX-miljö

7,5 ECTS credit points (7,5 högskolepoäng)

**Course code:** DV1305

**Educational level:** Basic level

**Course level:** G2F

**Field of education:** Technology

**Subject group:** Computer Technology

**Subject area:** Computer Science

**Version:** 7

**Applies from:** 2007-08-24

**Approved:** 2009-11-01

**Replaces course descriptor approved:** 2003-03-17

#### 1 Course title and credit points

The course is titled Programming in UNIX Environment/Programmering i UNIX-miljö and awards 7,5 credit points. One credit point (högskolepoäng) corresponds to one credit point in the European Credit Transfer System (ECTS).

#### 2 Decision and approval

This course is established by School of Computing 2007-08-24. The course descriptor was revised by School of Computing and applies from 2007-08-24. Replaces DVC011.

Reg no: TEK56-268/07

#### 3 Objectives

Almost no programmer interacts with the bare hardware of the computer any longer (this has in fact been true for the last thirty years or so). Instead the programmer (and even the user) interacts via some form of higher level programming interface that together with resource allocation and other bookkeeping constitutes an operating system. Other courses teach the basics of the how and why of operating systems and in doing so they lay the groundwork for this course, that teaches the details of how to write software that interacts with the computer via the UNIX operating system (and by extension one of the UNIX based/similar operating systems, such as Linux or MacOS). The reason that UNIX is preferred is that it has a long and rich history, many commercially important systems are being built on it but it is also equally at home in single person; free or open source endeavours, the source code to the operating system itself is available for study, the programming interface is reasonable in size and well documented, it is equally at home on the largest computers as well as small embedded systems and it is popular with a large following. As such, the objective of the course is for the students to gain an enhanced understanding of UNIX-based systems and an enhanced understanding of programming close to the bare operating systems. This includes becoming comfortable with programming the operating system at the most abstract level, closest to the user level and further down the layers of abstraction to the lowest levels of system calls. This lays the groundwork for further study of other topics (e.g. computer security) that requires a concrete as opposed to abstract understanding of the underlying technology. It goes without saying that the above cannot come about without the student acquiring a healthy dose of programming experience through laboratory work.

#### 4 Content

Specifically, the following different levels of abstraction in UNIX programming are treated in more detail:

- Commands and script languages – Script languages are a way of automating what is normally thought of as user interaction, i.e. interactively giving commands to the computer to, for example, download files, sort the contents and print them out. As such script languages (exemplified with the Bourne shell) are very powerful and as a result less generally useful (i.e. it might be difficult to obtain exactly the result you want).
- Program development in C – The mainstay of UNIX application programming where the programmer uses what is commonly thought of as a programming language and by making system calls and library calls to the operating system implements network services and other applications (e.g. a web server) that explores the full gamut of power and

pitfalls inherent in advanced programming.

- Program development in assembly language – Sometimes higher level languages such as C aren't expressive enough, or carries unwanted baggage (i.e. provides features that are unneeded and drags on performance) and it becomes necessary to talk to the computer in its native tongue. In fairness

though, the real benefit from learning to program close to the machine comes from gaining a deeper understanding of what really goes on under the hood.

## 5 Aims and learning outcomes

All laboratory work in the course is performed in groups of two students. On completion of the course the student will:

- individually be able to effectively use the UNIX command line interface
- individually be able to write programs using a substantial subset of the complete UNIX programming interface
- individually be able to decide which level of programming abstraction is best suited to solve a particular task
- individually be able to understand the major parts of a UNIX system and how they fit together to run a program and accomplish a task
- individually be able to develop programs to run in a UNIX environment in C and assembly language.

## 6 Generic skills

The following generic skills are trained in the course:

- Capacity for applying knowledge in practice
- Grounding in basic knowledge of the profession in practice
- Programming in the imperative paradigm
- Problem solving
- Teamwork
- Concern for quality

## 7 Learning and teaching

The course is campus-based. The education is divided into lectures and lab work. The course includes a series of lectures and lab work that together serve to give the student a firm theoretical understanding and practical experience of the topic. The gained knowledge is put to the test, and further deepened, by the submission of lab assignments in which topic-related problems should be solved by implementation on the system in question. The lab assignments must be done in groups of two students. Each lab assignment is presented in writing and presented and examined orally by the individual members of each group. This course uses a learning platform on which course material and information is published. Student discussions, submission of assignments and feedback is carried out via this learning platform. English is the language used for all lectures, assignments, and examinations.

The teaching language is English.

## 8 Assessment and grading

### *Examination of the course*

Code	Module	Credit	Grade
0710	Shell script programming	2.5 hp	U/G/VG
0720	UNIX Programming in C	2.5 hp	U/G/VG
0730	Assembly Programming	2.5 hp	U/G/VG

The course will be graded Fail (U), Pass (G) or Pass with Distinction (VG). All laboratory assignments will be reported in writing and will also be presented orally. In order to achieve the highest grade the individual student must have demonstrated considerable grasp of, and familiarity with all the topics touched on in the lab work during the oral examination of said without undue prodding by the examiner.

On request grades according to ECTS will be given.

## 9 Course evaluation

The course coordinator is responsible for systematically gathering feedback from the students in course evaluations and making sure that the results of these feed back into the development of the course.

## 10 Prerequisites

Prerequisite courses for this course are: Passed courses: DV1204-Operating Systems and DV1124-Data Structures and Algorithms and DVC016-Local Area Networks.

## 11 Field of education and subject area

The course is part of the field of education Technology and is included in the subject area Computer Science.

**12 Restrictions regarding degree**

The course cannot form part of a degree with another course, the content of which completely or partly corresponds with the contents of this course.

**13 Additional information**

The main programming languages are C and Bourne Shell. Students are expected to have learnt these programming languages earlier or be able to learn them on their own during the course.

**14 Course literature and other teaching material**

Advanced Programming in the UNIX Environment, 2nd ed.

Authors: Rago, Stephen & Stevens, Richard Publisher: Addison-Wesley Professional, Published: 2005, Number of pages: 927

ISBN: 0201433079

■



## COURSE DESCRIPTOR

### Advanced Topic in Computing

#### Fördjupningskurs i Datavetenskap och kommunikation

7,5 ECTS credit points (7,5 högskolepoäng)

**Course code:** DV2545

**Educational level:** Advanced level

**Course level:** A1F

**Field of education:** Technology

**Subject group:** Computer Technology

**Subject area:** Computer Science, Software Engineering

**Version:** 1

**Applies from:** 2013-09-02

**Approved:** 2013-05-28

---

#### 1 Course title and credit points

The course is titled Advanced Topic in Computing/Fördjupningskurs i Datavetenskap och kommunikation and awards 7,5 credit points. One credit point (högskolepoäng) corresponds to one credit point in the European Credit Transfer System (ECTS).

#### 2 Decision and approval

This course is established by School of Computing 2013-05-28. The course descriptor is approved by School of Computing and applies from 2013-09-02. Reg.no: BTH 4.1.1-0441-2013

Code	Module	Credit	Grade
1310	Seminars	3 hp	G-U
1320	Report	2 hp	G-U
1330	Examination[1]	2.5 hp	A-F

#### 3 Objectives

The purpose of the course is to give the students possibilities to deepen their knowledge and understanding within a specific area within computer science or software engineering. The course constitutes a direct preparation for a future master thesis.

#### 4 Content

#### 5 Aims and learning outcomes

On completion of course the student will:

- have acquired advanced knowledge of a specific area within computer science or software engineering
- have orientation of current research within chosen area
- demonstrate an ability to articulate bases for a more extensive research work and to discuss and motivate choices
- ability to write a scientific text

#### 6 Generic skills

The following general skills are trained in the course:

- ability to analyse
- ability to work independently and in group
- general knowledge in main subject area of studies
- presentation and argumentation

## **7 Learning and teaching**

The teaching language is English.

## **8 Assessment and grading**

### *Examination of the course*

<sup>1</sup> Determines the final grade for the course, which will only be issued when all components have been approved.

The course will be graded A Excellent, B Very good, C Good, D Satisfactory, E Sufficient, FX Insufficient, supplementation required, F Fail.

## **9 Course evaluation**

The course coordinator is responsible for systematically gathering feedback from the students in course evaluations and making sure that the results of these feed back into the development of the course.

## **10 Prerequisites**

The student must have successfully completed 90 ECTS in Software Engineering or Computer Science and 30 ECTS on advanced level in these areas, or possess equivalent knowledge.

## **11 Field of education and subject area**

The course is part of the field of education Technology and is included in the subject area Computer Science and the subject area Software Engineering.

## **12 Restrictions regarding degree**

The course cannot form part of a degree with another course, the content of which completely or partly corresponds with the contents of this course.

## **13 Additional information**

Replaces DV2508.

## **14 Course literature and other teaching material Main literature**

Literature is to a large extent scientific papers within the selected area. The papers are independently selected by the students in relation to the different projects during the course.

### **Reference literature**

Projects in Computing and Information Systems. A Student's Guide

Author: Christian Dawson Publisher: Addison Wesley; 2 edition Published: 2009

Number of pages: 304

ISBN10: 0273721313; ISBN13: 978-0273721314

■



# COURSE DESCRIPTOR

## Product-line Architecture

### Produktlinjearkitektur

7,5 ECTS credit points (7,5 högskolepoäng)

**Course code:** PA2410

**Educational level:** Advanced level

**Course level:** A1N

**Field of education:** Technology

**Subject group:** Computer Technology

**Subject area:** Software Engineering

**Version:** 6

**Applies from:** 2010-11-01

**Approved:** 2010-10-06

**Replaces course descriptor approved:**

#### 1 Course title and credit points

The course is titled Product-line Architecture/Produktlinjearkitektur and awards 7,5 credit points. One credit point (högskolepoäng) corresponds to one credit point in the European Credit Transfer System (ECTS).

#### 2 Decision and approval

This course is established by School of Computing 2007-11-20. The course descriptor was revised by School of Computing and applies from 2010-11-01. Reg no: COM-56-0158-2010

#### 3 Objectives

The structuring of software products into product-lines allows sharing of development effort within the product-family and as such counters the impact of ever growing system complexity. This makes it possible to sustain the rate of product innovation, while keeping guaranteed levels of overall system performance and quality.

The fundamental concept of a software product line is a domain specific product architecture based upon a layered set of platforms. This is supported by a software engineering process focused on pervasive reuse. The issues in software product line are not just technical but include process, organizational and business aspects as well. In this course the students should attain an in depth insight in the area of software product lines. Domain-Specific Modeling (DSM) is the latest approach to software development, promising to greatly increase in productivity, quality and ease of software creation as well as use of expertise to share the knowledge within the development team. Domain specific modeling is about creating a domain specific environment (language and framework) for the use by other developers in the domain.

Those two approaches try address similar problems and are complementary.

#### 4 Content

The course comprises the following elements:

##### **Software Product Line Engineering**

- introduction to the basic ideas behind and means to express fundamental concepts related to software product lines and domain specific modeling,
- scoping and domain analysis,
- design of the software product-line architecture,
- development of software product-line components,
- instantiation of family members, i.e. products, and
- evolution of software product-line assets, \*\*Domain Specific Software Development
- introduction to the basic ideas behind and means to express fundamental concepts related to domain specific modeling,
- scoping and domain analysis,
- development of the language and the framework,
- instantiation of family members, i.e. products, and
- evolution of software product-line assets,



## 5 Aims and learning outcomes

On the completion of the course the participants

- will have the knowledge on concepts and ideas behind the software product lines,
- will be able to model a specific domain related to a desired product line and consider development of specific domain oriented language,
- will be able independently specify, design, evaluate, implement and use small size software product line,
- evaluate the specific developed solutions.

## 6 Generic skills

The following generic competences will be trained:

- Decision making
- Ability to work autonomously
- Ability to work in a scientific way

## 7 Learning and teaching

The studies comprise lectures, individual reading,

and seminars.

In addition, compulsory assignments will be set, which may be solved individually and/or in teams within a pre-defined time limit.

A compulsory moment is to give an organized and professional written and oral presentation of the results at a significant level of detail.

The teaching language is English.

## 8 Assessment and grading

### *Examination of the course*

Code	Module	Credit	Grade
0710	Group assignments	4.5 hp	F-A
0720	Individual assignment[1]	3 hp	F-A

<sup>1</sup> Determines the final grade for the course, which will only be issued when all components have been approved.

The course will be graded F Fail, FX Fail, E Sufficient, D Satisfactory, C Good, B Very good or A Excellent.

## 9 Course evaluation

The course coordinator is responsible for systematically gathering feedback from the students in course evaluations and making sure that the results of these feed back into the development of the course.

## 10 Prerequisites

Prerequisite courses for this course are: Passed courses: PA1301-Software Architectures and Quality.

## 11 Field of education and subject area

The course is part of the field of education Technology and is included in the subject area Software Engineering.

## 12 Restrictions regarding degree

The course cannot form part of a degree with another course, the content of which completely or partly corresponds with the contents of this course.

## 13 Course literature and other teaching material Main Litteratur

Pohl K., et al., Software Product Line Engineering, Springer, 2005, ISBN 3-540-243720

Kelly S., Tolvalen J-P., Domain Specific Modeling, Willey-Blackwell, 2008, ISBN-978-0470036662

---

### Referens litterature

Van der Linden F., Software Product Lines in Action, Springer, 2007

Clements P., Northrop L.M., Software Product Lines: Practices and Patterns, 563 s, Addison-Wesley, 2002, ISBN 0-201-70332-7

Gommaa H., Designing Software Product Lines wit UML, Addison Wesley, 2004

■