ACADEMIC REGULATIONS
COURSE STRUCTURE AND
DETAILED SYLLABUS

COMPUTER SCIENCE & ENGINEERING

For

MCA (Master of Computer Applications)
(Three Year Full Time Programme)

JNTUH COLLEGE OF ENGINEERING HYDERABAD
(Autonomous)
Kukatpally, Hyderabad – 500 085, Telangana, India.

2015
Jawaharlal Nehru Technological University Hyderabad
College of Engineering Hyderabad
(Autonomous)
Kukatpally, Hyderabad – 500 085

Academic Regulations 2015
for CBCS Based M.C.A. (Regular/Full Time) Programmes
(Effective for the students admitted into I year from the Academic Year 2015-16 and onwards)

1.0 Post-Graduate Degree Programmes in Engineering & Technology (PGP in E & T):
JNTUH offers 3 Year (6 Semesters) full-time MCA. Degree Programmes, under Choice Based Credit System (CBCS) at its Constituent Autonomous College - JNTUH College of Engineering Hyderabad with effect from the Academic Year 2015-16 onwards.

2.0 Eligibility for Admission:
Admission to the above programme shall be made subject to the eligibility and qualifications prescribed by the College/University from time to time. Admission shall be made on the basis of the merit rank obtained by the qualifying candidate at ICET for MBA/MCA conducted by the TSCHE subject to reservations as prescribed by the Government from time to time.

3.0 M.C.A. Programme Structure:
3.1 The M.C.A. Programme in JNTUH-CEH is of Semester Pattern, with 6 Semesters constituting 3 Academic Years, each Academic Year having TWO Semesters (First/Odd and Second/Even Semesters). Each Semester shall be of 22 Weeks duration (inclusive of Examinations), with a minimum of 90 Instructional Days per Semester.

3.2 UGC/ AICTE specified Definitions/ Descriptions are adopted appropriately for various terms and abbreviations used in these MCA - Academic Regulations.

3.2.1 Semester Scheme:
Each Semester having - ‘Continuous Internal Evaluation (CIE)’ and ‘Semester End Examination (SEE)’. Choice Based Credit System (CBCS) and Credit Based Semester System (CBSS) as denoted are taken as ‘references’ for the present set of Regulations. The terms ‘SUBJECT’ or ‘COURSE’ imply the same meaning here, and refer to ‘Theory Subject’, or ‘Lab Course’, or ‘Design/ Drawing Subject’, or ‘Seminar’, or ‘Comprehensive Viva’, or ‘Project’, as the case may be.
3.2.2 **Credit Courses:**
All Subjects (or Courses) are to be registered by a student in a Semester to earn Credits. Credits shall be assigned to each Subject/Course in a L:T:P:C (Lecture Periods: Tutorial Periods: Practicals Periods: Credits) Structure, based on the following general pattern …
- One hour/Week/Semester for Theory/Lecture (L) Courses; and,
- Two hours/Week/Semester for Laboratory/Practical (P) Courses or Tutorials (T).

Other student activities like Study Tour, Guest Lecture, Conference/Workshop Participations, Technical Paper Presentations etc., and identified Mandatory Courses if any, will not carry Credits.

3.2.3 **Subject/Course Classification:**
All Subjects/Courses offered for the MCA are broadly classified as: (a) Core Courses (CoC), and (b) Elective Courses (EtC).
- Core Courses (CoC) and Elective Courses (EtC) are categorized as PS (Professional Subjects), which are further subdivided as – (i) PC (Professional/Departmental Core) Subjects, (ii) PE (Professional/Departmental Electives), (iii) Seminar, (iv), and (v) Project Work (PW).

3.2.4 **Course Nomenclature:**
The Curriculum Nomenclature or Course-Structure Grouping for the M.C.A. Degree Programmes is as listed below …

<table>
<thead>
<tr>
<th>S. No.</th>
<th>Broad Course Classification</th>
<th>Course Group/Category</th>
<th>Courses Description</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>1)</td>
<td>Foundation Courses (FnC)</td>
<td>BS – Basic Sciences</td>
<td>Includes - Mathematics, Physics and Chemistry Subjects</td>
<td>04</td>
</tr>
<tr>
<td>2)</td>
<td>HS – Humanities and Social Sciences</td>
<td>Includes subjects related to Humanities, Social Sciences and Management</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>3)</td>
<td>Core Courses (CoC)</td>
<td>PC-Professional Core</td>
<td>Includes core subjects related to the Parent Discipline/Department/Branch of Engg.</td>
<td>108</td>
</tr>
<tr>
<td>4)</td>
<td>Elective Courses (EtC)</td>
<td>PE-Professional Electives</td>
<td>Includes Elective subjects related to the Parent Discipline/Department/Branch of Engg.</td>
<td>12</td>
</tr>
<tr>
<td>5)</td>
<td>OE - Open Elective</td>
<td>OE- Open Elective</td>
<td>Elective subjects which include inter-disciplinary subjects or subjects in an area outside the Parent Discipline/Department/Branch of Engg.</td>
<td>04</td>
</tr>
<tr>
<td>6)</td>
<td>Core Courses</td>
<td>Project Work M.C.A. Project</td>
<td>22</td>
<td></td>
</tr>
<tr>
<td>7)</td>
<td>Seminar</td>
<td>Seminar/Colloquium based on core contents related to Parent Discipline/Department/Branch of Engg.</td>
<td>2</td>
<td></td>
</tr>
</tbody>
</table>

**Total Credits for PGP** 162
4.0 Course Work:

4.1 A Student, after securing admission, shall pursue and complete the M.C.A. in a minimum period of 3 Academic Years (6 Semesters), and within a maximum period of 6 Academic Years (starting from the Date of Commencement of I Year).

4.2 Each student shall Register for and Secure the specified number of Credits required for the completion and Award of the M.C.A.

4.3 I Year is structured to provide typically 28 Credits (28 C) in each of the I and II Semesters, and II Year comprises of 28 Credits (28 C), in each of the I and II Semesters, III Year comprises of 28 credits (28 C) in I-Sem and 22 credits in II Semester, totaling to 162 Credits (162 C) for the entire M.C.A. Programme.

5.0 Course Registration:

5.1 A ‘Faculty Advisor’ shall be assigned to M.C.A., who will advise the Students about the M.C.A. Programme, its Course Structure and Curriculum, Choice/Option for Subjects/Courses, based on his competence, progress, pre-requisites and interest.

5.2 A Student may be permitted to Register for Subjects/Courses of ‘his CHOICE’ with a typical total of 28 Credits per Semester in the first five semesters (Minimum being 24 C and Maximum being 32 C, permitted deviation being ±15%), 22 credits (inclusive of Project) in VI Semester in III Year (minimum being 20 C and maximum 26 C), based on his interest, competence, progress, and ‘PRE-REQUISITES’ as indicated for various Subjects/Courses, in the Department Course Structure (for the relevant Specialization) and Syllabus contents for various Subjects/Courses.

5.3 Choice for ‘additional Subjects/Courses’ in any Semester (above the typical 28/22 Credit norm, and within the Maximum Permissible Limit of 32/26 Credits, as applicable) must be clearly indicated in the Registration, which needs the specific approval and signature of the Faculty Advisor/Counselor on hard-copy.

5.4 Dropping of Subjects/Courses in any of the first five semesters of the program may be permitted, ONLY AFTER obtaining prior approval and signature from the Faculty Advisor (subject to retaining a minimum of 24 Credits), ‘within 15 Days of Time’ from the beginning of the current Semester.

6.0 Attendance Requirements:

6.1 A Student shall be eligible to appear for the End Semester Examination (SEE) of any Subject, if he acquires a minimum of 75% of attendance in that Subject for that Semester.

6.2 A Student’s Seminar Report and Seminar Presentation shall be eligible for evaluation, only if he ensures a minimum of 75% of his attendance in Seminar Presentation Classes during that Semester.

6.3 Condoning of shortage of attendance up to 10% (65% and above, and below 75%) in each Subject or Seminar of a Semester may be granted by the College
Academic Council on genuine and valid grounds, based on the Student’s representation with supporting evidence.

6.4 A stipulated fee per Subject/Seminar shall be payable towards condoning of shortage of attendance.

6.5 Shortage of Attendance below 65% in any Subject/Seminar shall in NO case be condoned.

6.6 A Student, whose shortage of attendance is not condoned in any Subject(s) or Seminar in any Semester, is considered as ‘Detained in that Subject(s)/Seminar’, and is not eligible to take End Examination(s) of such Subject(s) (and in case of Seminars, his Seminar Report or Presentation are not eligible for evaluation) in that Semester; and he has to seek Re-registration for those Subject(s)/Seminar in subsequent Semesters, and attend the same as and when offered.

7.0 **Academic Requirements:**
The following Academic Requirements have to be satisfied, in addition to the Attendance Requirements mentioned in Item No. 6.

7.1 A Student shall be deemed to have satisfied the Academic Requirements and earned the Credits allotted to each Subject/Course, if he secures not less than 40% Marks (28 out of 70 Marks) in the End Semester Examination, and a minimum of 50% of Marks in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together; in terms of Letter Grades, this implies securing B Grade or above in that Subject.

7.2 A Student shall be deemed to have satisfied the Academic Requirements and earned the Credits allotted to - Seminar, if he secures not less than 50% of the total Marks to be awarded for each. The Student would be treated as failed, if he - (i) does not present the Seminar as required, or (ii) secures less than 50% of Marks (<50 Marks) in Seminar.

7.3 A Student shall - register for all Subjects covering 162 Credits as specified and listed in the Course Structure for the chosen PGP Specialization, put up all the Attendance and Academic requirements for securing 162 Credits obtaining a minimum of B Grade or above in each Subject, and ‘earn all 162 Credits securing SGPA ≥ 5.0 (in each Semester) and final CGPA (ie., CGPA at the end of PGP) ≥ 5.0, to successfully complete the PGP.

7.4 Students who fail to earn 162 Credits as per the specified Course Structure, and as indicated above, within 6 Academic Years from the Date of Commencement of their I Year, shall forfeit their seats in M.C.A. Programme and their admissions shall stand cancelled.

7.5 When a Student is detained due to shortage of attendance in any Subject(s)/Seminar in any Semester, no Grade Allotment will be done for such Subject(s)/Seminar, and SGPA/CGPA calculations of that Semester will not include the performance evaluations of such Subject(s)/Seminar in which he got detained. However, he becomes eligible for re-registration of such Subject(s)/Seminar (in which he got detained) in the subsequent Semester(s), as and when next offered, with the Academic Regulations of the Batch into
which he gets readmitted, by paying the stipulated fees per Subject. In all these re-registration cases, the Student shall have to secure a fresh set of Internal Marks (CIE) and End Semester Examination Marks (SEE) for performance evaluation in such Subject(s), and subsequent SGPA/CGPA calculations.

7.6 A Student eligible to appear in the End Semester Examination in any Subject, but absent at it or failed (failing to secure B Grade or above), may reappear for that Subject at the supplementary examination (SEE) as and when conducted. In such cases, his Internal Marks (CIE) assessed earlier for that Subject/ Course will be carried over, and added to the marks to be obtained in the supplementary examination (SEE), for evaluating his performance in that Subject.

8.0 Evaluation - Distribution and Weightage of Marks:

8.1 The performance of a Student in each Semester shall be evaluated Subject-wise (irrespective of Credits assigned) with a maximum of 100 Marks for Theory or Practicals or Seminar etc; however, the M.C.A. Project Work (Major Project) will be evaluated for 200 Marks.

8.2 a) For Theory Subjects, there shall be a Continuous Internal Evaluation (CIE) during the Semester for 30 Internal Marks, and 70 Marks are assigned for Lab./Practicals End Semester Examination (SEE). CIE Marks shall comprise of - Mid-Term Examination Marks (for 25 Marks), and Assignment Marks (for 5 Marks).

b) During the Semester, there shall be 2 Mid-Term examinations. Each Mid-Term examination shall be for 25 Marks (with 120 minutes duration). The better performance out of these two Mid-Term Examinations shall be considered for the award of 25 Marks.

8.3 For Practical Subjects, there shall be a Continuous Internal Evaluation (CIE) during the Semester for 30 Internal Marks, and 70 Marks are assigned for Lab./Practicals End Semester Examination (SEE). Out of the 30 Marks for Internals, day-to-day work assessment in the laboratory shall be evaluated for 20 Marks; and the performance in an internal Lab./Practical Test shall be evaluated for 10 marks. The SEE for Lab./Practicals shall be conducted at the end of the Semester by the concerned Lab. Teacher and another faculty member of the same Department as assigned by the Head of the Department.

8.4 There shall be a Seminar Presentation in I Year I Semester or II Semester. For the Seminar, the Student shall collect the information on a specialized topic, prepare a Technical Report and submit to the Department at the time of Seminar Presentation. The Seminar Presentation (along with the Technical Report) shall be evaluated by Two Faculty Members assigned by Head of the Department, for 100 Marks. There are no internal marks for the seminar.

8.5 a) Every MCA Student shall be required to execute his M.C.A Project, under the guidance of the Supervisor assigned to him by the Head of Department. The MCA Project shall start immediately after the completion of
the II Year II Semester, and shall continue through III Year I and II Semesters. The Student shall carry out the literature survey, select an appropriate topic and submit a Project Proposal within 6 weeks (immediately after his I Year II Semester End Examinations), for approval by the Project Review Committee (PRC). The PRC shall be constituted by the Head of Department, and shall consist of the Head of Department, Project Supervisor, and a Senior Faculty Member of the Department. The Student shall present his Project Work Proposal to the PRC (PRC-I Presentation), on whose approval he can ‘REGISTER for the PG Project’. Every Student must compulsorily register for his M.C.A. Project Work, within the 6 weeks of time-frame as specified above. After Registration, the Student shall carry out his work, and continually submit ‘a fortnightly progress report’ to his Supervisor throughout the Project period. The PRC will monitor the progress of the Project Work and review, through PRC-II and PRC-III Presentations – one at the end of the II Year I Semester, and one before the submission of M.Tech. Project Work Report/ Dissertation.

b) After PRC-III presentation, the PRC shall evaluate the entire performance of the Student and declare the Project Report as ‘Satisfactory’ or ‘Unsatisfactory’. Every Project Work Report/ Dissertation (that has been declared ‘satisfactory’) shall undergo ‘Plagiarism Check’ as per the University/ College norms to ensure content plagiarism below a specified level of 30%, and to become acceptable for submission. In case of unacceptable plagiarism levels, the student shall resubmit the Project Work Report, after carrying out the necessary modifications/ additions to his Project Work/ Report as per his Supervisor’s advice, within the specified time, as suggested by the PRC.

c) If any Student could not be present for PRC-II at the scheduled time (after approval and registration of his Project Work at PRC-I), his submission and presentation at the PRC-III time (or at any other PRC specified dates) may be treated as PRC-II performance evaluation, and delayed PRC-III dates for him may be considered as per PRC recommendations. Any Student is allowed to submit his M.C.A. Project Dissertation ‘only after completion of 40 weeks from the date of approval/registration’ of his Project, and after obtaining all approvals from the PRC.

d) A total of 200 Marks are allotted for the M.C.A Project Work, ( out of which 100 Marks are allotted for internal evaluation and 100 Marks for external evaluation). For internal Evaluation of 100 marks, Project Supervisor shall evaluate for 60 marks based on the continuous Internal Evaluation(CIE) of the student’s performance and combined PRC-I, II & III performance evaluation will be for 40 marks (to be awarded by PRC, as SEE).

8.6  a) The Student shall be allowed to submit his Project Dissertation, only on the successful completion of all the prescribed PG Subjects (Theory and Labs.), Seminar, (securing B Grade or above), and after obtaining all approvals from PRC. In such cases, the M.C.A. Dissertations will be sent to an External Examiner nominated by the Principal of the College, on whose ‘approval’, the Student can appear for the M.C.A. Project Viva-voce Examination, which shall be conducted by a Board, consisting of the PG Project Supervisor, Head of the Department, and the External Examiner who adjudicated the M.C.A. Project Work and Dissertation. The Board shall
jointly declare the Project Work Performance as ‘satisfactory’, or ‘unsatisfactory’; and in successful cases, the External Examiner shall evaluate the Student’s Project Work presentation and performance for 100 Marks (SEE).

b) If the adjudication report of the External Examiner is ‘not favourable’, then the Student shall revise and resubmit his Dissertation after one Semester, or as per the time specified by the External Examiner and/or the PRC. If the resubmitted report is again evaluated by the External Examiner as ‘not favourable’, then that Dissertation will be summarily rejected. Subsequent actions for such Dissertations may be considered, only on the specific recommendations of the External Examiner and/or PRC.

c) In cases, where the Board declared the Project Work Performance as ‘unsatisfactory’, the Student is deemed to have failed in the Project Viva-voce Examination, and he has to reappear for the Viva-voce Examination as per the Board recommendations. If he fails in the second Viva-voce Examination also, he will not be considered eligible for the Award of the Degree, unless he is asked to revise and resubmit his Project Work by the Board within a specified time period (within 6 years from the date of commencement of his I Year I Semester).

9.0 Re-Admission / Re-Registration:

9.1 Re-Admission for Discontinued Students:
Students, who have discontinued the M.C.A. Degree Programme due to any reasons whatsoever, may be considered for ‘Readmission’ into the same Degree Programme (with same specialization) with the Academic Regulations of the Batch into which he gets readmitted, with prior permission from the concerned authorities, subject to Item 4.1.

9.2 Re-Registration for Detained Students:
When any Student is detained in a Subject(s)/ Seminar due to shortage of attendance in any Semester, he may be permitted to re-register for the same Subject in the ‘same category’ (Core or Elective Group) or equivalent Subject if the same Subject is not available, as suggested by the Board of Studies of that Department, as when offered in the subsequent Semester(s), with the Academic Regulations of the Batch into which he seeks re-registration, with prior permission from the concerned authorities, subject to Item 4.1.

10.0 Grading Procedure:

10.1 Marks will be awarded to indicate the performance of each student in each Theory Subject, or Lab/Practicals, or Seminar, or Project, etc., based on the % marks obtained in CIE + SEE (Continuous Internal Evaluation + Semester End Examination, both taken together) as specified in Item 6 above, and a corresponding Letter Grade shall be given.
10.2 As a measure of the student’s performance, a 10-point Absolute Grading System using the following Letter Grades (UGC Guidelines) and corresponding percentage of marks shall be followed:

<table>
<thead>
<tr>
<th>% of Marks Secured (Class Intervals)</th>
<th>Letter Grade (UGC Guidelines)</th>
<th>Grade Points</th>
</tr>
</thead>
<tbody>
<tr>
<td>80% and above (≥ 80%, ≤ 100%)</td>
<td>O (Outstanding)</td>
<td>10</td>
</tr>
<tr>
<td>Below 80% but not less than 70%</td>
<td>A+ (Excellent)</td>
<td>9</td>
</tr>
<tr>
<td>(≥ 70%, &lt; 80%)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Below 70% but not less than 60%</td>
<td>A (Very Good)</td>
<td>8</td>
</tr>
<tr>
<td>(≥ 60%, &lt; 70%)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Below 60% but not less than 55%</td>
<td>B+ (Good)</td>
<td>7</td>
</tr>
<tr>
<td>(≥ 55%, &lt; 60%)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Below 55% but not less than 50%</td>
<td>B (above Average)</td>
<td>6</td>
</tr>
<tr>
<td>(≥ 50%, &lt; 55%)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Below 50% (≤ 50%)</td>
<td>F (FAIL)</td>
<td>0</td>
</tr>
<tr>
<td>Absent</td>
<td>Ab</td>
<td>0</td>
</tr>
</tbody>
</table>

10.3 A student obtaining F Grade in any Subject shall be considered ‘failed’ and is be required to reappear as ‘Supplementary Candidate’ in the Semester End Examination (SEE), as and when offered. In such cases, his Internal Marks (CIE Marks) in those Subjects will remain the same as those he obtained earlier.

10.4 A Letter Grade does not imply any specific % of Marks.

10.5 A student earns Grade Point (GP) in each Subject/ Course, on the basis of the Letter Grade obtained by him in that Subject/ Course (excluding Mandatory non-credit Courses). Then the corresponding ‘Credit Points’ (CP) are computed by multiplying the Grade Point with Credits for that particular Subject/ Course.

Credit Points (CP) = Grade Point (GP) x Credits …. For a Course

10.6 The Semester Grade Point Average (SGPA) is calculated by dividing the Sum of Credit Points (ΣCP) secured from ALL Subjects/ Courses registered in a Semester, by the Total Number of Credits registered during that Semester. SGPA is rounded off to TWO Decimal Places. SGPA is thus computed as

\[
SGPA = \frac{\sum_{i=1}^{N} C_i G_i}{\sum_{i=1}^{N} C_i} \quad \text{For each Semester},
\]

where ‘i’ is the Subject indicator index (takes into account all Subjects in a Semester), ‘N’ is the no. of Subjects ‘REGISTERED’ for the Semester (as specifically required and listed under the Course Structure of the parent Department), \(C_i\) is the no. of Credits allotted to the \(i^{th}\) Subject, and \(G_i\) represents the Grade Points (GP) corresponding to the Letter Grade awarded for that \(i^{th}\) Subject.
10.7 The Cumulative Grade Point Average (CGPA) is a measure of the overall cumulative performance of a student over all Semesters considered for registration. The CGPA is the ratio of the Total Credit Points secured by a student in ALL registered Courses in ALL Semesters, and the Total Number of Credits registered in ALL the Semesters. CGPA is rounded off to TWO Decimal Places. CGPA is thus computed from the I Year second Semester onwards, at the end of each Semester, as per the formula

\[ \text{CGPA} = \frac{\sum_{j=1}^{M} C_{j} \cdot G_{j}}{\sum_{j=1}^{M} C_{j}} \text{ ... for all } S \text{ Semesters registered} \]

(i.e., up to and inclusive of S Semesters, \( S \geq 1 \)).

where ‘M’ is the TOTAL no. of Subjects (as specifically required and listed under the Course Structure of the parent Department) the Student has ‘REGISTERED’ from the 1st Semester onwards up to and inclusive of the Semester S (obviously M > N), ‘j’ is the Subject indicator index (takes into account all Subjects from 1 to S Semesters), \( C_{j} \) is the no. of Credits allotted to the \( j^{th} \) Subject, and \( G_{j} \) represents the Grade Points (GP) corresponding to the Letter Grade awarded for that \( j^{th} \) Subject. After registration and completion of I Year I Semester however, the SGPA of that Semester itself may be taken as the CGPA, as there are no cumulative effects.

10.8 For Merit Ranking or Comparison Purposes or any other listing, ONLY the ‘ROUNDED OFF’ values of the CGPAs will be used.

10.9 For Calculations listed in Item 10.5 – 10.8, performance in failed Subjects/ Courses (securing F Grade) will also be taken into account, and the Credits of such Subjects/ Courses will also be included in the multiplications and summations. However, Mandatory Courses will not be taken into consideration.

10.10 A student shall be declared successful or ‘passed’ in a Semester, only when he gets a SGPA \( \geq 5.00 \) (at the end of that particular Semester); and a student shall be declared successful or ‘passed’ in the entire M.C.A, only when gets a CGPA \( \geq 5.00 \); subject to the condition that he secures a GP \( \geq 6 \) (B Grade or above) in every registered Subject/ Course in each Semester (during the entire PGP), for the Award of the Degree, as required.

10.11 After the completion of each Semester, a Grade Card or Grade Sheet (or Transcript) shall be issued to all the Registered Students of that Semester, indicating the Letter Grades and Credits earned. It will show the details of the Courses Registered (Course Code, Title, No. of Credits, Grade Earned etc.), Credits earned, SGPA, and CGPA.

10.12 **Passing Standards** :

10.12.1 A Student shall be declared successful or ‘passed’ in a Semester, only when he gets a SGPA \( \geq 5.00 \) (at the end of that particular Semester); and a Student shall be declared successful or ‘passed’ in the entire PGP, only when gets a CGPA \( \geq 5.00 \); subject to the condition that he secures a GP \( \geq 6 \) (B Grade or above) in every registered Subject/ Course in each Semester (during the entire PGP), for the Award of the Degree, as required.

10.12.2 After the completion of each Semester, a Grade Card or Grade Sheet (or Transcript) shall be issued to all the Registered Students of that Semester, indicating the Letter Grades and Credits earned. It will show
the details of the Courses Registered (Course Code, Title, No. of Credits, Grade Earned), Credits earned, SGPA, and CGPA etc.

11.0 Declaration of Results:

11.1 Computation of SGPA and CGPA are done using the procedure listed in 10.5 – 10.8.

11.2 For Final % of Marks equivalent to the computed CGPA, the following formula may be used..

\[
\% \text{ of Marks} = (\text{CGPA} - 0.5) \times 10
\]

12.0 Award of Degree and Class:

12.1 A Student who registers for all the specified Subjects/ Courses as listed in the Course Structure, satisfies all the Course Requirements, and passes the examinations prescribed in the entire PG Programme (PGP), and secures the required number of 90 Credits (with GP \( \geq 6.0 \)), shall be declared to have ‘QUALIFIED’ for the award of the M.Tech. Degree in the chosen Branch of Engineering and Technology with specialization as he admitted.

12.2 Award of Class

After a student has satisfied the requirements prescribed for the completion of the program and is eligible for the award of M. Tech. Degree, he shall be placed in one of the following four classes based on the % CGPA:

<table>
<thead>
<tr>
<th>Class Awarded</th>
<th>CGPA</th>
</tr>
</thead>
<tbody>
<tr>
<td>First Class with Distinction</td>
<td>( \geq 7.75 )</td>
</tr>
<tr>
<td>First Class</td>
<td>( 6.75 \leq \text{CGPA} &lt; 7.75 )</td>
</tr>
<tr>
<td>Second Class</td>
<td>( 6.0 \leq \text{CGPA} &lt; 6.75 )</td>
</tr>
</tbody>
</table>

12.3 A student with final CGPA (at the end of the PGP) < 6.00 will not be eligible for the Award of Degree.

13.0 Withholding of Results:

13.1 If a Student has not paid fees to University/ College at any stage, or has pending dues against his name due to any reason whatsoever, or if any case of indiscipline is pending against him, the result of the Student 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.

14.0 Transitory Regulations:

14.1 A Student - who has discontinued for any reason, or who has been detained for want of attendance as specified, or who has failed after having undergone M.C.A, may be considered eligible for readmission to the M.C.A with same set of Subjects/ Courses (or equivalent Subjects/ Courses as the case may be), and same Professional Electives (or from same set/category of Electives or equivalents as suggested), as and when they are offered (within the time-
frame of 6 years from the Date of Commencement of his I Year I Semester).

15.0 Student Transfers:

15.1 There shall be no Branch/ Specialization transfers after the completion of Admission Process.

15.2 There shall be no transfer among the Constituent Colleges and Units of Jawaharlal Nehru Technological University Hyderabad.

16.0 Scope:

i) Where the words “he”, “him”, “his”, occur in the write-up of regulations, they include “she”, “her”, “hers”.

ii) Where the words “Subject” or “Subjects”, occur in these regulations, they also imply “Course” or “Courses”.

iii) The Academic Regulations should be read as a whole, for the purpose of any interpretation.

iv) In case of any doubt or ambiguity in the interpretation of the above regulations, the decision of the Vice-Chancellor/ Principal is final.

v) The College may change or amend the Academic Regulations, and/or Course Structure, and/or Syllabi at any time, and the changes or amendments made shall be applicable to all students with effect from the dates as notified by the University/ College.

17. MALPRACTICES RULES:

<table>
<thead>
<tr>
<th>Nature of Malpractices</th>
<th>Punishment</th>
</tr>
</thead>
<tbody>
<tr>
<td>If the candidate:</td>
<td></td>
</tr>
<tr>
<td>1 (a) Possesses or keeps accessible in examination hall, any paper, notebook, 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)</td>
<td>Expulsion from the examination hall and cancellation of the performance in that subject only.</td>
</tr>
<tr>
<td>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.</td>
<td>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.</td>
</tr>
<tr>
<td>2 Has copied in the examination hall</td>
<td>Expulsion from the examination hall and</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>1</td>
<td>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.</td>
</tr>
<tr>
<td>2</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Impersonates any other candidate in connection with the examination.</td>
</tr>
<tr>
<td>4</td>
<td>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.</td>
</tr>
<tr>
<td>5</td>
<td>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.</td>
</tr>
<tr>
<td>6</td>
<td>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</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>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.</td>
<td>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.</td>
</tr>
<tr>
<td>7</td>
<td>Leaves the exam hall taking away answer script or intentionally tears of the script or any part thereof inside or outside the examination hall.</td>
</tr>
<tr>
<td>8</td>
<td>Possess any lethal weapon or firearm in the examination hall.</td>
</tr>
<tr>
<td>9</td>
<td>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.</td>
</tr>
</tbody>
</table>
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.

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.

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.

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.

18. GENERAL:

- **Credit**: A unit by which the course work is measured. It determines the number of hours of instructions required per week. One credit is equivalent to one hour of teaching (lecture or tutorial) or two hours of practical work/field work per week.

- **Credit Point**: It is the product of grade point and number of credits for a course.

- 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’, ‘Practical Subject’ or ‘Lab.’ and ‘Seminar’.

- In case of any ambiguity or doubt in the interpretations of the above regulations, the decision of the Vice-Chancellor will be final.

*****
# Master of Computer Applications (MCA) w.e.f 2015-16

## MCA 1st Year (I-Semester)

<table>
<thead>
<tr>
<th>Code</th>
<th>Subject</th>
<th>L</th>
<th>T</th>
<th>P</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>MC 1.1</td>
<td>Mathematical Foundations of Computer Science</td>
<td>4</td>
<td>-</td>
<td>-</td>
<td>4</td>
</tr>
<tr>
<td>MC 1.2</td>
<td>Computer Organization</td>
<td>4</td>
<td>-</td>
<td>-</td>
<td>4</td>
</tr>
<tr>
<td>MC 1.3</td>
<td>Computer Programming &amp; Data Structures</td>
<td>4</td>
<td>-</td>
<td>-</td>
<td>4</td>
</tr>
<tr>
<td>MC 1.4</td>
<td>Probability and Statistics</td>
<td>4</td>
<td>-</td>
<td>-</td>
<td>4</td>
</tr>
<tr>
<td>MC 1.5</td>
<td>Accountancy and Financial Management</td>
<td>4</td>
<td>-</td>
<td>-</td>
<td>4</td>
</tr>
<tr>
<td>MC 1.6</td>
<td>Professional Communication Skills</td>
<td>4</td>
<td>-</td>
<td>-</td>
<td>4</td>
</tr>
<tr>
<td>MC 1.7</td>
<td>Professional Communication Skills Lab</td>
<td>-</td>
<td>-</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>MC 1.8</td>
<td>Computer Programming &amp; Data Structures Lab</td>
<td>-</td>
<td>-</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>28</td>
</tr>
</tbody>
</table>

## MCA 1st Year (II-Semester)

<table>
<thead>
<tr>
<th>Code</th>
<th>Subject</th>
<th>L</th>
<th>T</th>
<th>P</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>MC 2.1</td>
<td>Object Oriented Programming through Java</td>
<td>4</td>
<td>-</td>
<td>-</td>
<td>4</td>
</tr>
<tr>
<td>MC 2.2</td>
<td>Advanced Data Structures &amp; Algorithms</td>
<td>4</td>
<td>-</td>
<td>-</td>
<td>4</td>
</tr>
<tr>
<td>MC 2.3</td>
<td>Computer Networks</td>
<td>4</td>
<td>-</td>
<td>-</td>
<td>4</td>
</tr>
<tr>
<td>MC 2.4</td>
<td>Operating Systems</td>
<td>4</td>
<td>-</td>
<td>-</td>
<td>4</td>
</tr>
<tr>
<td>MC 2.5</td>
<td>Database Management Systems</td>
<td>4</td>
<td>-</td>
<td>-</td>
<td>4</td>
</tr>
<tr>
<td>MC 2.6</td>
<td>Software Engineering</td>
<td>4</td>
<td>-</td>
<td>-</td>
<td>4</td>
</tr>
<tr>
<td>MC 2.7</td>
<td>Advanced Data Structures &amp; Algorithms Lab</td>
<td>-</td>
<td>-</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>MC 2.8</td>
<td>Database Management Systems Lab</td>
<td>-</td>
<td>-</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>28</td>
</tr>
</tbody>
</table>

## MCA 2nd Year (III-Semester)

<table>
<thead>
<tr>
<th>Code</th>
<th>Subject</th>
<th>L</th>
<th>T</th>
<th>P</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>MC 3.1</td>
<td>Network Administration</td>
<td>4</td>
<td>-</td>
<td>-</td>
<td>4</td>
</tr>
<tr>
<td>MC 3.2</td>
<td>Management Information Systems</td>
<td>4</td>
<td>-</td>
<td>-</td>
<td>4</td>
</tr>
<tr>
<td>MC 3.3</td>
<td>Computer Graphics</td>
<td>4</td>
<td>-</td>
<td>-</td>
<td>4</td>
</tr>
<tr>
<td>MC 3.4</td>
<td>Object Oriented Analysis and Design (using UML)</td>
<td>4</td>
<td>-</td>
<td>-</td>
<td>4</td>
</tr>
<tr>
<td>MC 3.5</td>
<td>Operations Research</td>
<td>4</td>
<td>-</td>
<td>-</td>
<td>4</td>
</tr>
<tr>
<td>MC 3.6</td>
<td>Software Testing Methodologies</td>
<td>4</td>
<td>-</td>
<td>-</td>
<td>4</td>
</tr>
<tr>
<td>MC 3.7</td>
<td>Network Administration Lab</td>
<td>-</td>
<td>-</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>MC 3.8</td>
<td>CASE Tools Lab</td>
<td>-</td>
<td>-</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>28</td>
</tr>
</tbody>
</table>
## MCA 2nd Year (IV-Semester)

<table>
<thead>
<tr>
<th>Code</th>
<th>Subject</th>
<th>L</th>
<th>T</th>
<th>P</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>MC 4.1</td>
<td>Java and Web Technologies</td>
<td>4</td>
<td></td>
<td></td>
<td>4</td>
</tr>
<tr>
<td>MC 4.2</td>
<td>Data Warehousing and Data Mining</td>
<td>4</td>
<td></td>
<td></td>
<td>4</td>
</tr>
<tr>
<td>MC 4.3</td>
<td>Android Application Development</td>
<td>4</td>
<td></td>
<td></td>
<td>4</td>
</tr>
<tr>
<td>MC 4.4</td>
<td>Scripting Languages</td>
<td>4</td>
<td></td>
<td></td>
<td>4</td>
</tr>
<tr>
<td>MC 4.5</td>
<td>Dept. Elective-I</td>
<td>4</td>
<td></td>
<td></td>
<td>4</td>
</tr>
<tr>
<td>MC 4.6</td>
<td>Open Elective</td>
<td>4</td>
<td></td>
<td></td>
<td>4</td>
</tr>
<tr>
<td>MC 4.7</td>
<td>Web Programming Lab</td>
<td></td>
<td>4</td>
<td></td>
<td>2</td>
</tr>
<tr>
<td>MC 4.8</td>
<td>Android Programming Lab</td>
<td></td>
<td>4</td>
<td></td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>28</td>
</tr>
</tbody>
</table>

**MC 4.5 Dept. Elective-I**
- MC 4.5.1 Soft Computing
- MC 4.5.2 Software Project Management
- MC 4.5.3 Digital Forensics
- MC 4.5.4 Information Security & Audit Control

**MC 4.6 Open Elective**
- MC 4.6.1 Professional Ethics
- MC 4.6.2 Intellectual Property Rights & Cyber Laws
- MC 4.6.3 Ethical Hacking
- MC 4.6.4 Engineering Management

## MCA 3rd Year (V-Semester)

<table>
<thead>
<tr>
<th>Code</th>
<th>Subject</th>
<th>L</th>
<th>T</th>
<th>P</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>MC 5.1</td>
<td>Mobile Commerce</td>
<td>4</td>
<td></td>
<td></td>
<td>4</td>
</tr>
<tr>
<td>MC 5.2</td>
<td>Information Security</td>
<td>4</td>
<td></td>
<td></td>
<td>4</td>
</tr>
<tr>
<td>MC 5.3</td>
<td>Big Data Analytics</td>
<td>4</td>
<td></td>
<td></td>
<td>4</td>
</tr>
<tr>
<td>MC 5.4</td>
<td>Cloud Computing</td>
<td>4</td>
<td></td>
<td></td>
<td>4</td>
</tr>
<tr>
<td>MC 5.5</td>
<td>Dept. Elective-II</td>
<td>4</td>
<td></td>
<td></td>
<td>4</td>
</tr>
<tr>
<td>MC 5.6</td>
<td>Dept. Elective-III</td>
<td>4</td>
<td></td>
<td></td>
<td>4</td>
</tr>
<tr>
<td>MC 5.7</td>
<td>Information Security Lab</td>
<td>-</td>
<td>4</td>
<td></td>
<td>2</td>
</tr>
<tr>
<td>MC 5.8</td>
<td>Project Stage I</td>
<td></td>
<td>4</td>
<td></td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>28</td>
</tr>
</tbody>
</table>

**MC 5.5 Dept. Elective-II**
- MC 5.5.1 Distributed Systems
- MC 5.5.2 Machine Learning
- MC 5.5.3 Business Intelligence
- MC 5.5.4 Real Time Operating Systems
**MC 5.6 Dept. Elective-III**
- MC 5.6.1 Distributed Databases
- MC 5.6.2 Web Services & Service Oriented Architecture
- MC 5.6.3 Simulation & Modeling
- MC 5.6.4. Information Retrieval Systems

**MCA 3rd Year (VI-Semester)**

<table>
<thead>
<tr>
<th>Code</th>
<th>Subject</th>
<th>L</th>
<th>T</th>
<th>P</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>MC 6.1</td>
<td>Technical Seminar (External Evaluation)</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>2</td>
</tr>
<tr>
<td>MC 6.2</td>
<td>Project Stage II</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>20</td>
</tr>
</tbody>
</table>

Total Credits: 22
MATHEMATICAL FOUNDATIONS OF COMPUTER SCIENCE

Prerequisites
- An understanding of Math in general is sufficient.

Objectives
- Introduces the elementary discrete mathematics for computer science and engineering.
- Topics include formal logic notation, methods of proof, induction, sets, relations, algebraic structures, elementary graph theory, permutations and combinations, counting principles; recurrence relations and generating functions.

Outcomes
- Ability to understand and construct precise mathematical proofs
- Ability to use logic and set theory to formulate precise statements
- Ability to analyze and solve counting problems on finite and discrete structures
- Ability to describe and manipulate sequences
- Ability to apply graph theory in solving computing problems

UNIT I

UNIT II

UNIT III
Algebraic structures: Algebraic systems Examples and general properties, Groups, Semi groups, Subgroups and monoids, groups sub groups’ homomorphism, Isomorphism, Rings, Integral domains and Fields, Ring Homomorphism’s and Polynomial Rings.

UNIT IV
Recurrence Relations: Concepts of Generating Functions, Recurrence Relations, Solving Recurrence Relations by Substitution and Generating Functions, the Method of Characteristics Roots, Solutions of Inhomogeneous Recurrence Relations.
UNIT V
Graphs: Graph Terminology and Special Types of Graphs, Representing Graphs and Graph Isomorphism, Connectivity, Euler and Hamilton Paths, Shortest-Path Problems, Planar Graphs, Graph Coloring


TEXT BOOKS:

REFERENCE BOOKS:
Prerequisites
No prerequisites

Objectives
1. The purpose of the course is to introduce principles of computer organization, design of digital logic circuits and the basic architectural concepts.
2. It begins with digital components, digital logic circuit design, basic organization, and introduces simple register transfer language to specify various computer operations.
3. Topics include computer arithmetic, instruction set design, micro programmed control unit, pipelining, memory organization and I/O systems, and multiprocessors

Outcomes
1. Understand the basics of instructions sets and their impact on processor design.
2. Demonstrate an understanding of the design of the functional units of a digital computer system.
3. Design a pipeline for consistent execution of instructions with minimum hazards.
4. Recognize and manipulate representations of numbers stored in digital computers

UNIT-I
Digital Components: Integrated Circuits, Decoders, Multiplexers, Registers, Shift Registers, Binary Counters, Memory Unit
Data Representation: Data Types, Complements, Fixed-Point Representation, Floating-Point Representation, Other Binary Codes, Error Detection Codes

UNIT-II
Register Transfer Language and Micro operations: Register Transfer Language, Register Transfer, Bus and Memory Transfers, Arithmetic Micro operations, Logic Micro operations, Shift Micro operations, Arithmetic Logic Shift Unit
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, Design of Basic Computer, Design of Accumulator Logic

UNIT III
Micro Programmed Control: Control Memory, Address sequencing, Micro Program Example, Design of Control Unit.
Central Processing Unit: Introduction, General Register Organization, STACK organization, Instruction Formats, Addressing modes, Data Transfer and Manipulation, Program Control, Reduced Instruction Set Computer.

UNIT IV
Pipelining: Parallel Processing, Pipelining, Arithmetic Pipeline, Instruction Pipeline, RISC Pipeline

UNIT V
Input-Output Organization: Peripheral Devices, Input-Output Interface, Modes of Transfer, Priority Interrupt, Direct memory Access, Input –Output Processor (IOP)
Memory Organization: Memory Hierarchy, Main Memory, Auxiliary memory, Associate Memory, Cache Memory, Virtual Memory, Memory Management Hardware.

TEXT BOOKS:

REFERENCE BOOKS:
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
JNTUH COLLEGE OF ENGINEERING HYDERABAD

I Year MCA II Semester

COMPUTER PROGRAMMING & DATA STRUCTURES

Prerequisites

- No prerequisites
- Requires analytical skills and logical reasoning.

Objectives

- This course starts from the basics of computers and program development.
- It covers various concepts of C programming language
- It introduces searching and sorting algorithms
- It provides an understanding of data structures such as stacks and queues.

Outcomes

- Ability to develop C programs for computing and real life applications using basic elements like control statements, arrays, functions, pointers and strings; and data structures like stacks, queues and linked lists.
- Ability to implement searching and sorting algorithms

UNIT - I


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

UNIT – II

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.

Designing Structured Programs- Functions, basics, user defined functions, inter function communication, Scope, Storage classes-auto, register, static, extern, scope rules, type qualifiers, recursion- recursive functions, Preprocessor commands, example C programs

UNIT – III

Arrays and Strings – Concepts, using arrays in C, inter function communication, array applications, two – dimensional arrays, multidimensional arrays, C program examples. Concepts, C Strings, String Input / Output functions, arrays of strings, string manipulation functions, string / data conversion, C program examples.

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.

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 and Searching selection sort, bubble sort, insertion sort, 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:
3. The C Programming Language, B.W. Kernighan and Dennis M.Ritchie, PHI/Pearson Education

REFERENCES:
3. C Programming & Data Structures, P. Dey, M Ghosh R Thereja, Oxford University Press
I Year MCA                          I Semester

PROBABILITY AND STATISTICS

Prerequisites

To learn concepts of Probability, Estimation and Testing.

Objectives

To understand Probability theory
To get the skill to apply the probability theory and distributions to solve the real time problems.
To set the skills of application of various types of statistical testing techniques

UNIT I
Probability: Sample space and events – Probability – The axioms of probability - Some elementary theorems - Conditional probability – Baye’s theorem.
Distribution - Binomial, poisson and normal distribution – related properties.

UNIT II
Sampling distribution: Populations and samples - Sampling distributions of mean (known and unknown) proportions, sums and differences.

UNIT III
Estimation: Point estimation – interval estimation - Bayesian estimation.
Test of Hypothesis – Means and proportions – Hypothesis concerning one and two means – Type I and Type II errors. One tail, two-tail tests.

UNIT IV
Tests of significance – Student’s t-test, F-test, \( \chi^2 \) test. Estimation of proportions.

UNIT V

TEXT BOOKS:

REFERENCE BOOKS:
I Year MCA                          I Semester

ACCOUNTANCY AND FINANCIAL MANAGEMENT

Prerequisites

- No prerequisites

Objectives

- To learn Financial Management and Accounting
- To learn different types of costing

Outcomes

- Able to prepare balance sheets of budget.
- Get the skill to manage finances of a firm/company

UNIT I

Introduction to Accounting: Principles, concepts, conventions, double entry system of accounting, introduction of basic books of accounts ledgers.

Preparation of trial balance: Final accounts - company final accounts.

UNIT II

Financial Management - meaning and scope, role, objectives of time value of money - over vitalization - under capitalization - profit maximization - wealth maximization - EPS maximization.

Ratio Analysis - advantages - limitations - Fund flow analysis - meaning, importance, preparation and interpretation of Funds flow and cash flow statements.

UNIT III

Costing: nature and importance and basic principles. Absorption costing vs. marginal costing - Financial accounting vs. cost accounting vs. management accounting.

Marginal costing and Break-even Analysis: nature, scope and importance - practical applications of marginal costing, limitations and importance of cost - volume, profit analysis.

UNIT IV

Standard costing and budgeting: nature, scope and computation and analysis - materials variance, labor variance and sales variance - budgeting - cash budget, sales budget - flexible Budgets, master budgets.

UNIT V

Introduction to computerized accounting system: coding logic and codes, master files, transaction files, introduction documents used for data collection, processing of different files and Outputs obtained.

TEXT BOOKS:

3. Financial Management, S.N.Maheshwari, Sultan Chand Company
I Year MCA                         I Semester

PROFESSIONAL COMMUNICATION SKILLS

Prerequisite: ---- NIL ----

Objectives:

- To learn the four language skills - Listening, Speaking, Reading and Writing; critical thinking skills to students.
- To enable students comprehend the concept of communication.
- To help students cultivate the habit of Reading and develop their critical reading skills.

Outcomes:

- Ability to convert the conceptual understanding of communication into every day practice.
- Ability to communicate their ideas relevantly and coherently in professional writing.

UNIT I
INTRODUCTION

UNIT II
READING & STUDY SKILLS
Reading Comprehension – Reading Strategies - Skimming and Scanning- Intensive and Extensive Reading– Unknown Passage for Comprehension - Critical Reading of Short Stories – Study Skills – Note Making – Summarizing – Articles and Prepositions – Synonyms and Antonyms

UNIT III
WRITING SKILLS

UNIT IV
PROFESSIONAL WRITING

UNIT V
REPORT WRITING
REFERENCE BOOKS:
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
JNTUH COLLEGE OF ENGINEERING HYDERABAD

I Year MCA                          I Semester

PROFESSIONAL COMMUNICATION SKILLS (PCS) LAB

Prerequisite: ----NIL---

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

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.
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.

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, 8th 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
- Raman, M & Sharma, S. 2011. Technical Communication, OUP
SUGGESTED READING:
4. Sasi Kumar, V & Dhamija, P.V. How to Prepare for Group Discussion and Interviews. Tata McGraw Hill
10. 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 30 sessional marks and 70 semester-end Examination marks. Of the 30 marks, 20 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.

* * * * *
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
JNTUH COLLEGE OF ENGINEERING HYDERABAD

I Year MCA                          I Semester

COMPUTER PROGRAMMING & DATA STRUCTURES LAB

Prerequisites

- No prerequisites
- Requires analytical skills and logical reasoning.

Objectives

- Covers various concepts of C programming language
- Introduces searching and sorting algorithms
- Provides an understanding of data structures such as stacks and queues.

Outcomes

- Develop C programs for computing and real life applications using basic elements like control statements, arrays, functions, pointers and strings, and data structures like stacks, queues and linked lists.
- Implement searching and sorting algorithms

Week 1:
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.

Week 2:
5. Write a C program to find the factorial of a given integer.
6. Write a C program to find the GCD (greatest common divisor) of two given integers.
7. Write a C program to solve Towers of Hanoi problem.
8. 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)

Week 3:
9. Write a C program to find both the largest and smallest number in a list of integers.
10. Write a C program that uses functions to perform the following:
   i) Addition of Two Matrices
   ii) Multiplication of Two Matrices

Week 4:
11. 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.
12. Write a C program to determine if the given string is a palindrome or not
13. 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.
14. Write a C program to count the lines, words and characters in a given text.

Week 5:
15. Write a C program to generate Pascal’s triangle.
16. Write a C program to construct a pyramid of numbers.
17. Write a C program to read in two numbers, x and n, and then compute the sum of this geometric progression:
   \[ 1 + x + x^2 + x^3 + \ldots \ldots + x^n \]
   For example: if n is 3 and x is 5, then the program computes 1+5+25+125.
   Print x, n, the sum.
   Perform error checking. For example, the formula does not make sense for negative exponents – if n is less than 0. Have your program print an error message if n<0, then go back and read in the next pair of numbers of without computing the sum. Are any values of x also illegal? If so, test for them too.

Week 6:
18. 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.
19. Write a C program to convert a Roman numeral to its decimal equivalent.

Week 7:
20. 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.)

Week 8:
21. 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.)
22. 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)

Week 9:
23. Write a C program that uses functions to perform the following operations on singly linked list:
    i) Creation
    ii) Insertion
    iii) Deletion
    iv) Traversal

Week 10:
24. Write C programs that implement stack (its operations) using
    i) Arrays
    ii) Pointers
25. Write C programs that implement Queue (its operations) using
    i) Arrays
    ii) Pointers
Week 11:
26. 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

Week 12:
27. 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

TEXT BOOKS:
3. The C Programming Language, B.W. Kernighan and Dennis M. Ritchie, PHI/Pearson Education

REFERENCES:
3. C Programming & Data Structures, P. Dey, M. Ghosh R Thereja, Oxford University Press
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
JNTUH COLLEGE OF ENGINEERING HYDERABAD

I Year MCA                          II Semester

OBJECT ORIENTED PROGRAMMING THROUGH JAVA

Prerequisites

- A course on “Computer Programming & Data Structures”

Objectives

- Introduces object oriented programming concepts using the Java language.
- Introduces the principles of inheritance and polymorphism; and demonstrates how they relate to the design of abstract classes
- Introduces the implementation of packages and interfaces
- Introduces exception handling, event handling and multithreading
- Introduces the design of Graphical User Interface using applets and swings

Outcomes

- Develop applications for a range of problems using object-oriented programming techniques
- Design simple Graphical User Interface applications

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.


TEXT BOOKS:
1. Java the complete reference, 7th edition, Herbert Schildt, TMH.
2. Understanding OOP with Java, updated edition, T. Budd, Pearson Education.

REFERENCES:
2. Introduction to Java programming, Y. Daniel Liang, Pearson Education.
3. An introduction to Java programming and object oriented application development, R.A. Johnson- Thomson.
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING  
JNTUH COLLEGE OF ENGINEERING HYDERABAD

I Year MCA                         II Semester

ADVANCED DATA STRUCTURES & ALGORITHMS

Prerequisites
- A course on “Computer Programming & Data Structures”

Objectives
- Introduces the notations for analysis of the performance of algorithms
- Introduces a variety of data structures such as hash tables, disjoint sets and Priority Queue
- Describes major algorithmic techniques (divide-and-conquer, backtracking, dynamic programming and greedy) and mention problems for which each technique is appropriate;
- Introduces sorting, searching and pattern matching algorithms

Outcomes
- Ability to analyze the performance of algorithms
- Ability to select the data structures that efficiently model the information in a problem
- Ability to choose appropriate data structures and algorithm design methods for a specified application
- Ability to understand how the choice of data structures and the algorithm design methods impact the performance of programs
- Design programs using a variety of data structures, including hash tables, disjoint sets, trees and graphs
- Implement and know the application of algorithms for sorting and pattern matching

Unit-I
**Introduction:** Algorithm, Psuedo code for expressing algorithms, Performance Analysis- Space complexity, Time complexity, Asymptotic Notations, Substitution method, Master’s theorem

**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

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

**Priority Queues** – Definition, Realizing a Priority Queue using Heaps, operations of priority queue

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

**Backtracking:** General method, applications-n-queens problem, sum of subsets problem, graph coloring.
Unit – IV
Review of basic data structures: The list, Stack, Queue
linear list representation, skip list representation, operations - insertion, deletion and searching.
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.
Searching: Linear Search and Binary Search
Sortings: Bubble sort, Selection sort, Insertion sort, Radix sort, Heap sort

Unit – V
Trees: Definition, Types of trees, Binary Trees, Binary Tree Traversal Methods, Binary search tree operations, operations of AVL tree, B-Tree, Red-Black tree, Splay tree
Graphs: Definition, Representation of graphs, Graphs Traversal Methods.
String Matching algorithms: Brute Force algorithm, Boyer Moore algorithm, Knuth- Morris-Pratt algorithm

TEXT BOOKS:
2. Fundamentals of Data Structures, Ellis Horowitz, Satraj Sahni and Rajasekharam, Universities Press.

REFERENCES:
2. Data structures using C and C++, Langsam, Augenstein and Tanenbaum, PHI.
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
JNTUH COLLEGE OF ENGINEERING HYDERABAD

I Year MCA                          II Semester

COMPUTER NETWORKS

Prerequisites

⇒ A course on “Computer Programming and Data Structures”

Objectives

⇒ To introduce equip the students with a general overview of the concepts and fundamentals of computer networks.
⇒ Familiarize the students with the standard models for the layered approach to communication between machines in a network and the protocols of the various layers.

Outcomes

⇒ Gain the knowledge of the functions of each layer in the OSI and TCP/IP reference model.
⇒ Obtain the skills of subnetting and routing mechanisms.
⇒ Understanding the essential protocols of computer networks, and how they can be applied in network design and implementation.

UNIT-I
Network hardware, Network software, OSI, TCP/IP Reference models, Example Networks: ARPANET, Internet.

UNIT-II
Data link layer: Design issues, framing, Error detection and correction.
Elementary data link protocols: simplex protocol, A simplex stop and wait protocol for an error-free channel, A simplex stop and wait protocol for noisy channel.
Sliding Window protocols: A one-bit sliding window protocol, A protocol using Go-Back-N, A protocol using Selective Repeat, Example data link protocols.
Medium Access sub layer: The channel allocation problem, Multiple access protocols: ALOHA, Carrier sense multiple access protocols, collision free protocols. Wireless LANs, Data link layer switching.

UNIT-III

UNIT –IV
Transport Layer: Transport Services, Elements of Transport protocols, Connection management, TCP and UDP protocols.

UNIT –V
Application Layer –Domain name system, SNMP, Electronic Mail; the World WEB, HTTP, Streaming audio and video.
TEXT BOOK:

REFERENCE BOOKS:
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
JNTUH COLLEGE OF ENGINEERING HYDERABAD

I Year MCA                         II Semester

OPERATING SYSTEMS

Prerequisites

- A course on “Computer Programming and Data Structures”
- A course on “Computer Organization and Architecture”

Objectives

- Provide an introduction to operating system concepts (i.e., processes, threads, scheduling, synchronization, deadlocks, memory management, file and I/O subsystems and protection)
- Introduce the issues to be considered in the design and development of operating system
- Introduce basic Unix commands, system call interface for process management, interprocess communication and I/O in Unix

Outcomes

- Will be able to control access to a computer and the files that may be shared
- Demonstrate the knowledge of the components of computer and their respective roles in computing.
- Ability to recognize and resolve user problems with standard operating environments.
- Gain practical knowledge of how programming languages, operating systems, and architectures interact and how to use each effectively.

UNIT I:
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.

UNIT II:
System call interface for process management -fork, exit, wait, waitpid, exec
Deadlocks - System Model, Deadlocks Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, and Recovery from Deadlock.

UNIT – III:
**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**

**UNIT V:**

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

**TEXT BOOKS:**

**REFERENCE BOOKS:**
2. Operating System A Design Approach-Crowley,TMH.
4. Unix programming environment, Kernighan and Pike, PHI / Pearson Education
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
JNTUH COLLEGE OF ENGINEERING HYDERABAD

I Year MCA                        II Semester

DATABASE MANAGEMENT SYSTEMS

Prerequisites
- A course on “Advanced Data Structures”

Objectives
- To understand the basic concepts and the applications of database systems.
- To master the basics of SQL and construct queries using SQL.
- Topics include datamodels, database design, relational model, relational algebra, transaction control, concurrency control, storage structures and access techniques.

Outcomes
- Gain knowledge of fundamentals of DBMS, database design and normal forms
- Gain knowledge of fundamentals of relational algebra and relational model
- Master the basics of SQL for retrieval and management of data.
- Understand the basics of transaction processing and concurrency control.
- Familiarity with database storage structures and access techniques

UNIT I:
Database System Applications: database 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 application programs, database users and administrator, transaction management, database 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, aggregation 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, problems related to decomposition, reasoning about functional dependencies, FIRST, SECOND, THIRD normal forms, BCNF, lossless join decomposition, dependency preserving decomposition, schema refinement in database design, multi valued dependencies, FOURTH normal form, FIFTH normal form.
UNIT IV:
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:

Text Books:

References:
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.
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING  
JNTUH COLLEGE OF ENGINEERING HYDERABAD

I Year MCA                        II Semester

SOFTWARE ENGINEERING

Prerequisites
- A course on “Computer Programming and Data Structures”

Objectives
- To provide an understanding of the working knowledge of the techniques for estimation, design, testing and quality management of large software development projects.

Outcomes
- Ability to translate end-user requirements into system and software requirements, using e.g. UML, and structure the requirements in a Software Requirements Document (SRD).
- Identify and apply appropriate software architectures and patterns to carry out high level design of a system and be able to critically compare alternative choices.
- Will have experience and/or awareness of testing problems and will be able to develop a simple testing report

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:**
3. The unified modeling language user guide Grady Booch, James Rambaugh, Ivar Jacobson, Pearson Education.

**References:**
I Year MCA                                          II Semester

ADVANCED DATA STRUCTURES & ALGORITHMS LAB

Prerequisites
- A course on “Computer Programming & Data Structures” and “Advanced Data Base Engineering”.

Objectives
- To get practical exposure on Advanced Data Structures like AVL Trees, Red-Black trees etc.,
- Implementation of data structures such as trees and graphs,
- Programming of sorting and pattern matching algorithms.

Outcomes
- Ability to select the data structures that efficiently model the information in a problem.
- Ability to assess efficiency trade-offs among different data structure implementations or combinations.
- Implement and know the application of algorithms for sorting and pattern matching.
- Design programs using tree structures, including Optimal binary search tree, All pairs shortest path problem, AVL tree, Red-Black trees.

1. Write a program to implement the following sorting techniques
   a. Merge sort
   b. Quick sort
   c. Bubble sort
   d. Selection sort
   e. Insertion sort
   f. Radix sort
   g. Heap sort
2. Write a program to implement the single source shortest path problem
3. Write a program to implement the Optimal binary search tree
4. Write a program to implement the All pairs shortest path problem
5. Write a program to implement the n-queens problem
6. Write a program to implement the following searching techniques
   a. Linear Search
   b. Binary Search
7. Write a program to implement the operations of Binary search tree
8. Write a program to implement the tree traversal methods
9. Write a program to implement the graph traversal methods
10. Write a program to implement the operations of AVL tree
11. Write a program to implement the operations of Red-Black tree
12. Write a program to implement the following Pattern matching algorithms
    a. Brute Force algorithm
    b. Boyer Moore algorithm
    c. Knuth- Morris-Pratt algorithm
DATABASE MANAGEMENT SYSTEMS LAB

Pre-requisites

Objectives
1. Introduce ER data model, database design and normalization
2. Learn SQL basics for data definition and data manipulation

Outcomes
1. Design database schema for a given application and apply normalization
2. Acquire skills in using SQL commands for retrieval and management of data.
3. Develop solutions for database applications using procedures, cursors and triggers

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)
9) Procedures
10) Usage of Cursors

Text Books:

References:
2. Fundamentals of Database Systems, Elmasri Navr ate 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.
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING  
JNTUH COLLEGE OF ENGINEERING HYDERABAD  

II Year MCA                       III Semester  

NETWORK ADMINISTRATION  

Prerequisites:  
- Computer Networks  

Objectives:  
- To train the students in network implementation, administration and support  

Outcomes:  
- Students achieve requisite skills in system and network maintenance and management  

UNIT 1:  
OS Overview:  

UNIT 2:  
UNIX Basics:  
Introduction to UNIX files system, vi editor, file handling utilities, security by file permissions, process utilities, disk utilities, network commands, text processing utilities and backup utilities. Working with shell, shell script examples.  

UNIT 3:  
Network services and protocols - I  

UNIT 4:  
Network services and protocols -II  
Understanding of HTTP, FTP, NFS, Telnet, Proxy server, SAMBA server. Installation in Linux environment  

UNIT 5:  
Network services and protocols -III  
Email service. Email server configuration in linux. Proxy web services (IIS), Windows Deployment Service (WDS), Print services installation in windows environment.  

Text Books:  
2. Unix the ultimate guide, sumithabha Das, TMH  
4. Red Hat Enterprise Linux 6 Administration, Sander van Vugt, John wiley &sons.  

References:  
3. Linux system Administration, Tom Adelstein & Bill Lubanovic, Oreilly.
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
JNTUH COLLEGE OF ENGINEERING HYDERABAD

II Year MCA                        III Semester

MANAGEMENT INFORMATION SYSTEM

Prerequisites

---NIL---

Objectives

To introduce Information Systems Models
Topics include Types of Information Systems, ERP Modules etc.,

Outcomes

To understand different Types of Information Systems
To gain good knowledge of ERP Modules and ERP Implementation and Maintenance.

UNIT I

UNIT II

UNIT III

UNIT IV

UNIT V
TEXT BOOKS:

REFERENCE BOOKS:
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
JNTUH COLLEGE OF ENGINEERING HYDERABAD

II Year MCA                        III Semester

COMPUTER GRAPHICS

Prerequisites
- Familiarity with the theory and use of coordinate geometry and linear algebra, and matrix multiplications.
- A course on “Computer Programming and Data Structures"

Objectives
- The aim of this course is to provide an introduction of fundamental concepts and theory of computer graphics.
- Topics covered include graphics systems and input devices; geometric representations and 2D/3D transformations; viewing and projections; illumination and color models; animation; rendering and implementation; visible surface detection;

Outcomes
- Be able to understand graphics input and output devices.
- Be able to design basic graphics application programs, including animation
- Be able to design applications that display 2-D and 3-D graphic images to given specifications

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 (Bresenham’s and DDA Algorithm), 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 viewport coordinate transformation, viewing functions, Cohen-Sutherland and Cyrus-beck line clipping algorithms, Sutherland–Hodgeman polygon clipping algorithm, Polygon Filling

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.

UNIT-IV:
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-V:

**Computer animation**: Design of animation sequence, general computer animation functions, raster animation, computer animation languages, key frame systems, motion specifications

**Visible surface detection methods**: Classification, back-face detection, depth-buffer, scan-line, depth sorting, BSP-tree methods, area sub-division and octree methods

**Text Books**:

**References**:
4. Computer Graphics, Steven Harrington, TMH
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING  
JNTUH COLLEGE OF ENGINEERING HYDERABAD  

II Year MCA                        III Semester  

OBJECT ORIENTED ANALYSIS AND DESIGN (USING UML)  

Prerequisites  
- Software Engineering  

Objectives  
- To train students on object modeling  
- To apply unified process phases  
- To apply unified modeling language for software design of any applications  
- To study case studies for OOAD  

Outcomes  
- Will be able to use UML notations  
- Ability to apply unified process in software development  
- Ability to perform analysis and design using object modeling  

UNIT I  
Introduction to UML: Importance of modeling, principles of modeling, object oriented modeling, conceptual model of the UML, Architecture, Software Development Life Cycle.  

UNIT II  
Basic Structural Modeling: Classes, Relationships, common Mechanisms, and diagrams.  
Advanced Structural Modeling: Advanced classes, advanced relationships, Interfaces, Types and Roles, Packages.  
Class & Object Diagrams: Terms, concepts, modeling techniques for Class & Object Diagrams.  

UNIT III  
Basic Behavioral Modeling-II: Use cases, Use case Diagrams, Activity Diagrams.  

UNIT IV  
Advanced Behavioral Modeling: Events and signals, state machines, processes and Threads, time and space, state chart diagrams.  
Architectural Modeling: Component, Deployment, Component diagrams and Deployment diagrams.  

UNIT V  
Patterns and Frameworks, Artifact Diagrams. Case Study: The Unified Library application  

TEXT BOOKS:  
2. Hans-Erik Eriksson, Magnus Penker, Brian Lyons, David Fado: UML 2 Toolkit, WILEY-Dreamtech India Pvt. Ltd.
REFERENCE BOOKS:
4. Mark Priestley: Practical Object-Oriented Design with UML, TMH.
7. UML and C++, R.C.Lee, and W.M.Tepfenhart, PHI.
10. Learning UML 2.0, Russ Miles and Kim Hamilton, O’Reilly, SPD.
OPERATIONS RESEARCH

Prerequisites
A course on “Mathematics”

Objectives
To introduce Linear Programming, dynamic programming Operation Research Technique.
To understand Theory of games.

Outcomes
Gain the knowledge of Operation Research Techniques
Get the skill to apply Operation Research Techniques to address the real time problems.

UNIT I
ALLOCATION: Introduction, Linear programming Formulation, Graphical solution, Simplex method, artificial variable technique, Duality principle.

UNIT II
EEQUE CEING: Introduction, optimal solution for processing each of n-jobs through three machines, travelling salesman problem i.e., shortest acyclic route models.
REPLACEMENT: Introduction, replacement of items that deteriorate when money value is not counted and counted, replacement items that fail completely i.e., group replacements.

UNIT III
WAITING LINES: Introduction, single channel, poisson arrivals, exponential service times, unrestricted queue, with infinite population and finite population models, single channel, poisson arrivals, exponential service times with infinite population and restricted queue, multi channel, poisson arrivals, exponential service times with infinite population and unrestricted queue.

UNIT IV
INVENTORY: Introduction, single item deterministic models, production is instantaneous or at a constant rate, shortages are allowed or not allowed and withdrawals from stock is continuous, purchase inventory model with one price break, shortages are not allowed, Instantaneous production demand, production or purchase cost is relevant, stochastic models, demand may be discrete or variable or instantaneous production, instantaneous demand and no setup cost.
UNIT V
THEORY OF GAMES: Introduction, Minimax (maximum) criterion and optimal strategy, solution of games with saddle points, rectangular games without saddle points.
DYNAMIC PROGRAMMING: Introduction, Billman’s Principal of optimality, solution of problems with finite number of stages.

TEXT BOOKS:
1. S.D.SHARMA : Operations Research
   kedar Nath Ramnath,1972
Prerequisites
1. A course on “Software Engineering”

Objectives
1. To provide knowledge of the concepts in software testing such as testing process, criteria, strategies, and methodologies.
2. To develop skills in software test automation and management using latest tools.

Outcomes
1. Design and develop the best test strategies in accordance to the development model.

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 data flow testing, strategies in data flow 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:

References:
1. The craft of software testing - Brian Marick, Pearson Education.
2. Software Testing Techniques – SPD(Oreille)
II Year MCA                        III Semester

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
JNTUH COLLEGE OF ENGINEERING HYDERABAD

NETWORK ADMINISTRATION LAB

Prerequisites:
- Computer Networks

Objective:
- Practical exposure to students in network installation, administration and maintenance

Outcomes:
- Student understands Linux commands for configuring system
- Student acquires requisite skills in network installation, administration maintenance

1. Working with shell commands related to file permissions, processes, networking and backup.
2. Write a shell script to generate a multiplication table
3. Write a shell script that copies multiple files to a directory
4. Write a shell script which counts the number of lines and words present in a given file
5. Write a shell script that displays the list of files in the given directory
6. Write a shell script that adds, subtracts, multiplies and divides the given two numbers. There are two division options: one returns the quotient and the other returns reminder. The script requires 3 arguments: The operation to be used and two integer numbers. The options are add (-a), subtract (-s), multiply (-m) and reminder (-r).
7. Write a shell script to reverse the rows and columns of a matrix

System and network experiments:
1. Demonstration of server construction
2. Create the "LVM" with the name "development" by using 21PE's from the volume group "engineering". Consider the PE size as "8MB". Mount it on /mnt/secret with filesystem ext3.
3. Create a group named "sysadmin"
   A user curly and larry should belongs to "sysadmin" group as a secondary group. A user node should not have access to interactive shell and he should not be a member of "sysadmin" group. passwd for all user created should be "jenny".
4. Create the Directory "/home/manager" with the following characteristics. Group ownership of "/home/manager" should go to "manager" group. The directory should be have full permission for all members off "manager" group but not to any other users accept "root". Files created under "/home/manager" should get the same group ownership is set to the "manager" group
5. The user sarah must configure a cron job that runs daily at 14:23. and executes "/bin/echo "hyer" and deny the user max for creating cronjob
6. Resize the lvm "/dev/vgsrv/home" so that after reboot size should be in between 90MB to 120MB

7. Configure your system as "NTP" client for "instructor.example.com".

8. Note the following. instructor.example.com(192.168.0.254) "Nfs exports" /home/guests to your system where "x" is your station ip. Ldapuser's home directory is instructor. example.com:/home/guests/ldapuserX. Ldapuser's home directory should be automounted locally beneath at /home/guests/ldapuserX. While login with any of the ldapuser then only home directory should accesible from your system that ldapuserX.

9: Copy the file /etc/fstab to /var/tmp and configure the "ACL" as mention following. The file /var/tmp/fstab is owned by the "root". The file /var/tmp/fstab belongs to the group "root" The file /var/tmp/fstab should not be executable by other's. The user "sarah" should able to read and write to the file. The user "natasha" can neither read nor write to the file. other users (future and current) shuold be able to read /var/tmp/fstab.

10: Create the user "dax" with uid 4223

11: Extend the SWAP space with "250" MB dont remove the existing swap.

12: Configure FTP access from your system. Clients within the remote.test should not have anomyous FTP access to your system.

13: locate the files of owner "dax" and copy to the directory /root/found directory

14: List all lines which have string "enter" from "/usr/share/dict/words" file and copy the lines in /root/word. found.
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
JNTUH COLLEGE OF ENGINEERING HYDERABAD

II Year MCA                        III Semester

CASE TOOLS LAB

Prerequisites
A basic knowledge of programming.

Objectives
1. To provide knowledge of Software Testing Methods.
2. To develop skills in software test automation and management using latest tools.

Outcomes
1. Design and develop the best test strategies in accordance to the development model.

USING WINRUNNER-CASE TOOL
1. Recording in context sensitive mode and analog mode
2. GUI checkpoint for single property
3. GUI checkpoint for single object/window
4. GUI checkpoint for multiple objects
5. a) Bitmap checkpoint for object/window
   b) Bitmap checkpoint for screen area
6. Database checkpoint for Default check
7. Database checkpoint for custom check
8. Database checkpoint for runtime record check
9. 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
10. a) Batch testing without parameter passing
     b) Batch testing with parameter passing
     c) Data driven batch
11. Silent mode test execution without any interruption
12. Test case for calculator in windows application
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING  
JNTUH COLLEGE OF ENGINEERING HYDERABAD

II Year MCA                        IV Semester

JAVA AND WEB TECHNOLOGIES

Prerequisites
1. A course on “Computer Programming and Data Structures”
2. A course on “Objected-Oriented Programming Through Java”

Objectives
1. To learn the basic web concepts and Internet protocols
2. To introduce XML and processing of XML data
3. To introduce client side scripting with Javascript and DHTML
4. To introduce server side programming with Java servlets and JSP

Outcomes
1. Ability to create dynamic and interactive web sites
2. Gain knowledge of client side scripting using java sript and DHTML.
3. Demonstrate understanding of what is XML and how to parse and use XML data
4. Able to do server side programming with Java Servelets and JSP

UNIT I
HTML Common tags - List, Tables, images, forms, Frames; Cascading Style sheets;
Introduction to Java Scripts, Objects in Java Script, Dynamic HTML with Java Script
XML: Document type definition, XML Schemas, Document Object model, Presenting XML,
Using XML Processors: DOM and SAX.

UNIT II
Review of Applets, Class, Event Handling, AWT Programming
Introduction to Swing:

UNIT III
Java Beans Introduction to Java Beans, Advantages of Java Beans, BDK Introspection,
Using Bound properties, Bean Info Interface, Constrained properties Persistence,

UNIT IV
Introduction to JSP The Problem with Servelet. The Anatomy of a JSP Page, JSP Processing.
JSP Application Design with MVC Setting Up and JSP Environment: Installing the Java Software Development Kit, Tomcat Server & Testing Tomcat
JSP Application Development: Generating Dynamic Content, Using Scripting Elements
Implicit JSP Objects, Conditional Processing – Displaying Values Using an Expression to Set an Attribute, Declaring Variables and Methods Error Handling and Debugging Sharing Data Between JSP pages, Requests, and Users Passing Control and Date between Pages – Sharing Session and Application Data – Memory Usage Considerations.
UNIT V
Database Access Database Programming using JDBC Studying Javax.sql.* package
Accessing a Database from a JSP Page Application – Specific Database Actions Deploying
JAVA Beans in a JSP Page. Introduction to struts framework.

TEXT BOOKS:
1. Internet and World Wide Web – How to program by Dietel and Nieto Pearson Education
   Asia. (Chapters: 3, 4, 8, 9, 10, 11, 12 – 18)
2. The complete Reference Java 2 Third Edition by Patrick Naughton and Herbert Schildt.
   (Chapters: 19, 20, 21, 22, 25, 27)
3. Java Server Pages by Hans Bergstan. (Chapters: 1 – 9)

REFERENCE BOOKS:
1. Internet and World Wide Web – How to program by Dietel and Nieto PHI/Pearson
   Education Asia.
3. Murach’s beginning JAVA JDK 5, Murach, and SPD
5. Web Applications Technologies Concepts-Knuckles,John Wiley
6. Programming world wide web-Sebesta,Pearson
7. Building Web Applications-NIIT, PHI
8. Web Warrior Guide to Web Programming-Bai/Ekedaw-Thomas
9. Beginning Web Programming-Jon Duckett WROX.
DATA WAREHOUSING AND DATAMINING

Prerequisites
1. A course on “Database Management Systems”
2. Knowledge of probability and statistics

Objectives
This course presents the techniques for preprocessing data before mining, and describes the concepts related to data warehousing, On-Line Analytical Processing (OLAP), and data generalization.
It also presents methods for mining frequent patterns, associations, and correlations.
It then describes methods for data classification and prediction, and data-clustering approaches.

Outcomes
1. Ability to understand the types of the data to be mined and present a general classification of tasks and primitives to integrate a data mining system.
2. Apply preprocessing methods for any given raw data.
3. Devise efficient and cost effective methods for designing data warehouses.
4. Extract interesting patterns from large amounts of data.
5. Discover the role played by data mining in various fields.
6. Choose and employ suitable data mining algorithms to build analytical applications.
7. Evaluate the accuracy of supervised and unsupervised models and algorithms.

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

Text Books:

References:
1. Building the Data Warehouse- W. H. Inmon, Wiley Dreamtech India Pvt. Ltd..
2. Data Mining Introductory and Advanced topics –MARGARET H DUNHAM, PEA.
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
JNTUH COLLEGE OF ENGINEERING HYDERABAD

II Year MCA                        IV Semester

ANDROID APPLICATION DEVELOPMENT

Prerequisites:

- A Course on JAVA
- A Course on DBMS

Objectives:

- To demonstrate their understanding of the fundamentals of Android operating systems
- To improve their skills of using Android software development tools
- To demonstrate their ability to develop software with reasonable complexity on mobile platform
- To demonstrate their ability to deploy software to mobile devices
- To demonstrate their ability to debug programs running on mobile devices

Outcomes:

- Student understands the working of Android OS Practically.
- Student will be able to develop Android user interfaces
- Student will be able to develop, deploy and maintain the Android Applications.

Unit I:
Introduction to Android Operating System: Android OS design and Features – Android development framework, SDK features, Installing and running applications on Eclipse platform, Creating AVDs, Types of Android applications, Best practices in Android programming, Android tools
Android application components – Android Manifest file, Externalizing resources like values, themes, layouts, Menus etc, Resources for different devices and languages, Runtime Configuration Changes
Android Application Lifecycle – Activities, Activity lifecycle, activity states, monitoring state changes

Unit II:
Android User Interface: Measurements – Device and pixel density independent measuring units Layouts – Linear, Relative, Grid and Table Layouts
User Interface (UI) Components – Editable and non editable TextViews, Buttons, Radio and Toggle Buttons, Checkboxes, Spinners, Dialog and pickers
Event Handling – Handling clicks or changes of various UI components
Fragments – Creating fragments, Lifecycle of fragments, Fragment states, Adding fragments to Activity, adding, removing and replacing fragments with fragment transactions, interfacing between fragments and Activities, Multi-screen Activities

Unit III
Intents and Broadcasts: Intent – Using intents to launch Activities, Explicitly starting new Activity, Implicit Intents, Passing data to Intents, Getting results from Activities, Native Actions, using Intent to dial a number or to send SMS
Broadcast Receivers – Using Intent filters to service implicit Intents, Resolving Intent filters, finding and using Intents received within an Activity
Notifications – Creating and Displaying notifications, Displaying Toasts

Unit IV
Persistent Storage:Files – Using application specific folders and files, creating files, reading data from files, listing contents of a directory
Shared Preferences – Creating shared preferences, saving and retrieving data using Shared Preference
Database – Introduction to SQLite database, creating and opening a database, creating tables, inserting retrieving and deleting data, Registering Content Providers, Using content Providers (insert, delete, retrieve and update)

Unit V
Advanced Topics: Alarms – Creating and using alarms
Using Internet Resources – Connecting to internet resource, using download manager
Location Based Services – Finding Current Location and showing location on the Map, updating location

Text Books:
1. Professional Android 4 Application Development, Reto Meier, Wiley India, (Wrox), 2012

References:
1. Beginning Android 4 Application Development, Wei-Meng Lee, Wiley India (Wrox), 2013
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING  
JNTUH COLLEGE OF ENGINEERING HYDERABAD  

II Year MCA                        IV Semester  

SCRIPTING LANGUAGES  

Prerequisites  
- A course on “Computer Programming and Data Structures”  
- A course on “Object Oriented Programming Concepts”  

Objectives  
- This course provides an introduction to the script programming paradigm  
- introduces scripting languages such as Perl, PHP and Python.  
- Learning TCL  

Outcomes  
- Comprehend the differences between typical scripting languages and typical system and application programming languages.  
- Gain knowledge of the strengths and weakness of Perl, PHP TCL and Python; and select an appropriate language for solving a given problem.  
- Acquire programming skills in scripting language  
- Able to design web pages using Advanced PHP feature.  

UNIT – I Introduction to PERL and Scripting  
Scripts and Programs, Origin of Scripting , Scripting Today, Characteristics of Scripting Languages, Uses for Scripting Languages, Web Scripting, and the universe of Scripting Languages. PERL- Names and Values, Variables, Scalar Expressions, Control Structures, arrays, list, hashes, strings, pattern and regular expressions, subroutines.  

UNIT – II Advanced perl  
Finer points of looping, pack and unpack, filesystem, eval, datastructures, packages, modules, objects, interfacing to the operating system, Creating Internet ware applications, Dirty Hands Internet Programming, security Issues.  

UNIT – III PHP Basics  
PHP Basics- Features, Embedding PHP Code in your Web pages, Outputting the data to the browser, Datatypes, Variables, Constants, expressions, string interpolation, control structures . Function, Creating a Function, Function Libraries, Arrays, strings and Regular Expressions. PHP and Web Forms, Files, PHP Authentication and Methodologies - Hard Coded, File Based, Database Based, IP Based, Login Administration, Uploading Files with PHP, Sending Email using PHP, PHP Encryption Functions, the Mcrypt package, Building Web sites for the World.  

UNIT - IV TCL  
TCL Structure, syntax, Variables and Data in TCL, Control Flow, Data Structures, input/output, procedures , strings , patterns, files, Advance TCL- eval, source, exec and uplevel commands, Name spaces, trapping errors, event driven programs, making applications internet aware, Nuts and Bolts Internet Programming, Security Issues, C Interface.  
Tk
Tk-Visual Tool Kits, Fundamental Concepts of Tk, Tk by example, Events and Binding, Perl-Tk.

UNIT – V Python
Introduction to Python language, python-syntax, statements, functions, Built-in-functions and Methods, Modules in python, Exception Handling.

TEXT BOOKS:
1. The World of Scripting Languages, David Barron, Wiley Publications.

REFERENCE BOOKS:
1. Open Source Web Development with LAMP using Linux Apache, MySQL, Perl and PHP, J. Lee and B.Ware (Addison Wesley) Pearson Education.
2. Programming Python, M. Lutz, SPD.
4. PHP 5.1, I. Bayross and S. Shah, The X Team, SPD.
5. Core Python Programming, Chun, Pearson Education.
7. Perl by Example, E. Quigley, Pearson Education.
8. Programming Perl, Larry Wall, T. Christiansen and J. Orwant, O’Reilly, SPD.
9. Tcl and the Tk Toolkit, Ousterhout, Pearson Education.
10. PHP and MySQL by Example, E. Quigley, Prentice Hall (Pearson).
12. PHP Programming solutions, V. Vaswani, TMH.
Prerequisites
- A course on “Computer Programming and Data Structures”
- A course on “Object Oriented Programming Concepts”

Objectives
- This course provides an introduction to the script programming paradigm
- Introduces scripting languages such as Perl, PHP and Python.
- Learning TCL

Outcomes
- Comprehend the differences between typical scripting languages and application programming languages. Acquire programming skills using scripting languages.
- Gain knowledge of the strengths and weakness of Perl, PHP, TCL and Python; and select an appropriate language for solving a given problem
- Ability to design web pages using advanced features of PHP.

UNIT-I
AI Problems and Search: AI problems, Techniques, Problem Spaces and Search, Heuristic Search Techniques- Generate and Test, Hill Climbing, Best First Search Problem reduction, Constraint Satisfaction and Means End Analysis. Approaches to Knowledge Representation- Using Predicate Logic and Rules.

UNIT-II
Supervised Learning Networks-perceptron, Back propagation algorithm-Classification Problem-Speech Processing Case study.
Unsupervised Learning Network- Introduction, Fixed Weight Competitive Nets, Maxnet, Hamming Network, Kohonen Self-Organizing Feature Maps, Learning Vector Quantization,

UNIT-III
Introduction to Classical Sets (crisp Sets) and Fuzzy Sets- operations and Fuzzy sets. Classical Relations -and Fuzzy Relations- Cardinality, Operations, Properties and composition. Tolerance and equivalence relations.
Membership functions- Features, Fuzzification, membership value assignments, Defuzzification.

UNIT-IV
Fuzzy Arithmetic and Fuzzy Measures, Fuzzy Rule Base and Approximate Reasoning Fuzzy Decision making

UNIT-V
Text Books:

Reference Books:
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
JNTUH COLLEGE OF ENGINEERING HYDERABAD

II Year MCA                        IV Semester
Dept. Elective – I
SOFTWARE PROJECT MANAGEMENT

Prerequisites
1. A course on “Software Engineering”

Objectives
To develop skills in software project management
The topics include—software economics; software development life cycle; artifacts of the process; workflows; checkpoints; project organization and responsibilities; project control and process instrumentation;

Outcomes
Gain knowledge of software economics, phases in the life cycle of software development, project organization, project control and process instrumentation.
Analyze the major and minor milestones, artifacts and metrics from management and technical perspective.
Design and develop software product using conventional and modern principles of software project management

UNIT I
Conventional Software Management: The waterfall model, conventional software Management performance.

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.

UNIT IV
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:**

**REFERENCE BOOKS:**
2. Software Project Management, Joel Henry, Pearson Education.
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
JNTUH COLLEGE OF ENGINEERING HYDERABAD

II Year MCA                        IV Semester
Dept. Elective – I
DIGITAL FORENSICS

Prerequisites: --Nil—

Objectives:
- To gain understanding of Network intrusion and Cybercrime
- To understand forensic analysis of large amount of data & social networks.

Outcomes:
- Ability to use the current computer forensics tools
- Obtain skills for operating system forensics
- Ability to perform forensic investigation on data & image files.

Unit I:

Unit II:

Unit-III:
Investigating Data & Image Files-Steganography, Data Acquisition & Duplication, Forensics Investigation using EnCase, Recovering Deleted Files & Deleted Partitions, Image File Forensics.

Unit-IV:

Unit-V:

Text Books:
1. Computer Forensics (5 volume Set) mapping to CHFI (Certified Hacking Forensics Investigator), by EC-Council (international Council of Electronic Commerce Consultants):
   a) Computer forensics: investigation procedure and response, Vol. 1 of 5,
   b) Computer forensics: investigating hard disk, file and operating systems, Vol. 2 of 5,
   c) Computer forensics: investigating data and image files, Vol. 3 of 5,
   d) Computer forensics: investigating networks intrusions and cybercrime, Vol. 4 of 5,
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
JNTUH COLLEGE OF ENGINEERING HYDERABAD

II Year MCA                        IV Semester
Dept. Elective – I
INFORMATION SECURITY & AUDIT CONTROL

Prerequisites
---NIL---

Objectives
- Introduces the basic concepts of Information System Auditing.
- To understand The Management Control Frameworks and The Application Control Framework.

Outcomes
- Understand function of Information Systems Audit and Management
- To acquire skills on Evidence Collection & Evaluation

Unit-I
Overview of Information System Auditing-Conducting an Information Systems Audit, Overview & steps in an Audit.

Unit-II

Unit -III
The Application Control Framework-I : Boundary Controls, Input Controls, Processing Controls, Database Controls, output Controls.

Unit -IV

Unit-V

Text Books:

References:
- Davis: IT Auditing, TMH, 2007
- Alter: Information Systems, Pearson, 20
Prerequisites:--- Nil ----

Objectives

- To introduce the basic concepts of universal human values
- To familiarize the students with desirable business and professional ethics, rights and responsibilities
- To prepare students against possible gaps and unethical practices in contemporary times
- To sensitise the students so that they can protect themselves and the organization from the possible professional crime malpractices

Outcomes

- The students learn about diverse ethical issues rooted in society, trade, business, and environment on local as well as a global platform.
- The students appreciate their role as a responsible citizen, professional, and as managers, advisors, experts and consultants.
- The students will reflect and learn major values and ethics from their observations of a spiritual discourse and a visit to a business organization as a practical part of this course.


Unit IV Professional Rights: professional rights and employee rights communicating risk and public policy – Whistle blowing - Professionals /engineers as managers, advisors, experts, witnesses and consultants – moral leadership- Regulatory compliances, Monitoring and control- 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.
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
JNTUH COLLEGE OF ENGINEERING HYDERABAD

II Year MCA                        IV Semester

Open Elective
INTELLECTUAL PROPERTY RIGHTS & CYBER LAWS

Prerequisites

Objectives
- Introduces the basic concepts of intellectual properties Rights and Cyber Laws..
- To understand Trademark Registration Process
- Explains Copyright Ownership and Law of Patents

Outcomes
- Students are enabled with techniques of protecting IPRS
- Students are enabled and appreciate consequences acts which relate IT act 2000

UNIT I

UNIT II

UNIT III
Inter Partes Proceedings, Infringement, Dilution, New developments in Trademark Law
Inter parts proceedings – Infringement of Trademarks – Dilution of Trademarks- Related Trademarks claims. Protecting a Domain Name - Other Cyberspace Trademark issues.

UNIT IV
Copyright Ownership, Transfers, Duration, Registration and Searching Copyright Ownership issues- joint works – ownership in derivative works – works made for hire- transfers of copyright- termination of Transfers of copyright- duration of copyright. Copyright registration. Application-Deposit Materials- Application Process and Registration of Copyright-Searching Copyright Office Records- Obtaining Copyright Office Records and Deposit Materials- Copyright Notice.
Copyright Infringement, New Developments in Copyright Law, semiconductor chip protection act: Elements of Infringement – Contributory Infringement and Vicarious

UNIT V


TEXT BOOKS:

REFERENCE BOOKS:
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING  
JNTUH COLLEGE OF ENGINEERING HYDERABAD

II Year MCA                        IV Semester

Open Elective
ETHICAL HACKING

Prerequisites
- A course on “Computer Networks”
- A course on “Network Security and Cryptography”

Objectives
- To introduce the methodologies and framework of ethical hacking for enhancing the security.

Outcomes
- Gain the knowledge of the use and availability of tools to support an ethical hack
- Gain the knowledge of interpreting the results of a controlled attack
- Comprehend the dangers associated with penetration testing

UNIT I
Introduction: Hacking Impacts, The Hacker
Framework: Planning the test, Sound Operations, Reconnaissance, Enumeration, Vulnerability Analysis, Exploitation, Final Analysis, Deliverable, Integration

UNIT II
The Business Perspective: Business Objectives, Security Policy, Previous Test Results, Business Challenges

UNIT III
Preparing for a Hack: Technical Preparation, Managing the Engagement
Reconnaissance: Social Engineering, Physical Security, Internet Reconnaissance

UNIT IV
Enumeration: Enumeration Techniques, Soft Objective, Looking Around or Attack, Elements of Enumeration, Preparing for the Next Phase
Exploitation: Intuitive Testing, Evasion, Threads and Groups, Operating Systems, Password Crackers, RootKits, applications, Wardialing, Network, Services and Areas of Concern

UNIT V
Deliverable: The Deliverable, The Document, Overall Structure, Aligning Findings, Presentation
Integration: Integrating the Results, Integration Summary, Mitigation, Defense Planning, Incident Management, Security Policy, Conclusion
TEXT BOOK

REFERENCE BOOKS
   1. EC-Council, “Ethical Hacking and Countermeasures Attack Phases”, Cengage Learning
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING  
JNTUH COLLEGE OF ENGINEERING HYDERABAD 

II Year MCA                  IV Semester 
Open Elective  
ENGINEERING MANAGEMENT

Prerequisite: 
⇒ A course on “Accountancy and Financial Management”  
⇒ A course on “Management Information Systems”

objectives: 
⇒ To sensitize and orient the future engineers about the challenges in managing engineering enterprises  
⇒ To learn how to provide value through innovations, leadership in technology projects, and the application of emerging technologies through web-based tools.

outcomes: 
⇒ The student learns how to manage operations in engineering organizations.  
⇒ The students will be enabled to become better engineering leaders who can manage personnel, projects, products, and services.  
⇒ This course will enable the students to do their professional engineering practice through the nuances of engineering management.

Unit-I 

Unit-II 
Value Engineering: Cost Reduction Vs Cost control- Cost Management – Value engineering methodologies

Unit-III 

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

Unit-V 
Advanced Manufacturing Technologies and systems -Web-Based Enablers For Engineering And Management – Managerial Issues relating to CAD/CAM//CAE- High performance Computing.

Text Book: 

References: 
2. A.K. Gupta, Engineering Management, S. Chand, 2010

81
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
JNTUH COLLEGE OF ENGINEERING HYDERABAD

II Year MCA                        IV Semester

WEB PROGRAMMING LAB

Prerequisites
☞ A Course on “Computer Programming and Data Structures”
☞ A Course on “Objected Oriented Programming through Java”

Co-requisites
☞ A course on “Web Technologies”

Objectives
☞ To provide hands-on experience on web technologies
☞ To develop client-server application using web technologies
☞ To introduce server side programming with Java servlets and JSP
☞ To introduce client side scripting with Javascript and AJAX

Outcomes
☞ Design and develop interactive and dynamic web applications using HTML, CSS, JavaScript and XML
☞ Apply client-server principles to develop scalable and enterprise web applications.

List of Experiments:
1. Develop static pages (using Only HTML) of an online book store. The pages should resemble: www.amazon.com. The website should consist the following pages.
   a) Home page
   b) Registration and user Login
   c) User Profile Page
   d) Books catalog
   e) Shopping Cart
   f) Payment By credit card
   g) Order Conformation

2. Validate the Registration, user login, user profile and payment by credit card pages using JavaScript.

3. Create and save an XML document on 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.

4. Bean Assignments
   a. Create a JavaBean which gives the exchange value of INR (Indian Rupees) into equivalent American/Canadian/Australian Dollar value.
   b. Create a simple Bean with a label - which is the count of number of clicks. Then create a BeanInfo class, such that only the “count” property is visible in the Property Window.
   c. Create two Beans- a) KeyPad. b) DisplayPad. After that integrate the two Beans to make it work as a Calculator.
   d. 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.

<table>
<thead>
<tr>
<th>Light Transition</th>
<th>Automobile State</th>
</tr>
</thead>
<tbody>
<tr>
<td>Red ---&gt; Yellow</td>
<td>Ready</td>
</tr>
<tr>
<td>Yellow ---&gt; Green</td>
<td>Move</td>
</tr>
<tr>
<td>Green --&gt; Red</td>
<td>Stopped</td>
</tr>
</tbody>
</table>

5. 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.

6. 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.

TEXT BOOKS:
1. WEB TECHNOLOGIES: A Computer Science Perspective, Jeffrey C. Jackson, Pearson Education

REFERENCES:
4. Paul Dietel and Harvey Deitel,"Java How to Program", Prentice Hall of India, 8th Edition
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
JNTUH COLLEGE OF ENGINEERING HYDERABAD

II Year MCA                      IV Semester

ANDROID PROGRAMMING LAB

Prerequisites: --- NIL---

Objectives:
¬ To learn how to develop Applications in android environment.
¬ To learn how to develop user interface applications.
¬ To learn how to develop URL related applications.

Objectives:
¬ Student understands the working of Android OS Practically.
¬ Student will be able to develop userinterfaces.
¬ Student will be able to develop, deploy and maintain the Android Applications.

The student is expected to be able to do the following problems, though not limited.

1. Create an Android application that shows Hello + name of the user and run it on an emulator. (b) Create an application that takes the name from a text box and shows hello message along with the name entered in text box, when the user clicks the OK button.

2. Create a screen that has input boxes for User Name, Password, Address, Gender (radio buttons for male and female), Age (numeric), Date of Birth (Date Picket), State (Spinner) and a Submit button. On clicking the submit button, print all the data below the Submit Button. Use (a) Linear Layout (b) Relative Layout and (c) Grid Layout or Table Layout.

3. Develop an application that shows names as a list and on selecting a name it should show the details of the candidate on the next screen with a “Back” button. If the screen is rotated to landscape mode (width greater than height), then the screen should show list on left fragment and details on right fragment instead of second screen with back button. Use Fragment transactions and Rotation event listener.

4. Develop an application that uses a menu with 3 options for dialing a number, opening a website and to send an SMS. On selecting an option, the appropriate action should be invoked using intents.

5. Develop an application that inserts some notifications into Notification area and whenever a notification is inserted, it should show a toast with details of the notification.

6. Create an application that uses a text file to store user names and passwords (tab separated fields and one record per line). When the user submits a login name and password through a screen, the details should be verified with the text file data and if they match, show a dialog saying that login is successful. Otherwise, show the dialog with Login Failed message.
7. Create a user registration application that stores the user details in a database table.

8. Create a database and a user table where the details of login names and passwords are stored. Insert some names and passwords initially. Now the login details entered by the user should be verified with the database and an appropriate dialog should be shown to the user.

9. Create an admin application for the user table, which shows all records as a list and the admin can select any record for edit or modify. The results should be reflected in the table.

10. Develop an application that shows all contacts of the phone along with details like name, phone number, mobile number etc.

11. Create an application that saves user information like name, age, gender etc. in shared preference and retrieves them when the program restarts.

12. Create an alarm that rings every Sunday at 8:00 AM. Modify it to use a time picker to set alarm time.

13. Create an application that shows the given URL (from a text field) in a browser.
    Develop an application that shows the current location’s latitude and longitude continuously as the device is moving (tracking).

14. Create an application that shows the current location on Google maps.

Note:
Android Application Development with MIT App Inventor: For the first one week, the student is advised to go through the App Inventor from MIT which gives insight into the various properties of each component. The student should pay attention to the properties of each components, which are used later in Android programming. Following are useful links:
1. http://ai2.appinventor.mit.edu
2. https://drive.google.com/file/d/0B8rTw_91YclTWF4czdBMEpZcWs/view
III Year MCA                        V Semester

MOBILE COMMERCE

Prerequisites

---NIL---

Objectives

Explains topics like E-commerce and Mobile commerce
To understand BUSINESS– TO– BUSINESS MOBILE E– COMMERCE

Outcomes

Understand how to apply E-commerce in the business
Apply Mobile commerce theory in the business.

UNIT I

ELECTRONIC COMMERCE 9

UNIT II

MOBILE COMMERCE 9

UNIT III

MOBILE COMMERCE: TECHNOLOGY 9

UNIT IV

MOBILE COMMERCE: THEORY AND APPLICATIONS 9

UNIT V

BUSINESS– TO– BUSINESS MOBILE E– COMMERCE 9
Enterprise Enablement – Email and Messaging – Field Force Automation (Insurance, Real Estate, Maintenance, Healthcare) – Field Sales Support (Content Access, Inventory) – Asset

TEXT BOOKS

REFERENCES
III Year MCA V Semester

INFORMATION SECURITY

Prerequisites
1. A Course on “Computer Networks, Mathematics

Objectives
- Understand information security’s importance in our increasingly computer-driven world...
- Master the key concepts of information security and how they “work.”
- To understand the fundamentals of Cryptography
- To understand the various key distribution and management schemes
- To understand how to deploy encryption techniques to secure data in transit across data networks
- To apply algorithms used for secure transactions in real world applications

Outcomes
- Demonstrate the knowledge of cryptography and network security concepts and applications.
- Ability to design different public & private key cryptography algorithms.
- Ability to identify and investigate vulnerabilities and security threats and mechanisms to counter them.

UNIT I
Security Attacks (Interruption, Interception, Modification and Fabrication), Security Services (Confidentiality, Authentication, Integrity, Non-repudiation, access Control and Availability) and Mechanisms, A model for Internetwork security.

UNIT II
Message authentication and Hash Functions, Authentication Requirements and Functions, Message Authentication, Hash Functions and MACs Hash and MAC Algorithms SHA-512, HMAC.

UNIT III
Email Security: Pretty Good Privacy (PGP) and S/MIME.

UNIT IV
IP Security:

UNIT V

TEXT BOOKS:

REFERENCE BOOKS:
III Year MCA                        V Semester

BIG DATA ANALYTICS

Prerequisites
  ➤ Data Mining

Objectives
  ➤ The purpose of this course is to provide the students with the knowledge of Big data Analytics principles and techniques.
  ➤ This course is also designed to give an exposure of the frontiers of Big data

Outcomes
  ➤ Ability to explain the foundations, definitions, and challenges of Big Data and Analytics.
  ➤ Apply Big Data technologies in Text/Speech Analytics.
  ➤ Ability to program using HADOOP and Map reduce, NOSQL

UNIT 1:
Introduction to Big data, Challenges of BD, Benefits of BD, characteristics of BD, How to use BD in your Business, BD analytics, BD Enterprise Model, Building BD Platform, Data intelligence, Data Science, Data Scientist.

UNIT 2:
Big Data storage, Big Data Architecture, Big Data Computation, Relational Data Base, Google Big Data services, Open Stack, Microsoft AZURE, and Integrating Data source, NOSQL

UNIT 3:
Apache Hadoop, Core Hadoop, Hadoop Lower Levels, HDFS, PIG, Hive, HBase, Data and NO SQL Planning for BD, Hadoop Data Storage and Analysis, HDFS: design, concepts, java interface, Hadoop file system, Hadoop pipes, streaming.

UNIT 4:
Map reduces types, I/O formats, counter, storing, joining Map Reduce Library Classes, Setting up Hadoop cluster, cluster specifications, setup and installation.

UNIT 5:

Text Books:
1. BIG DATA and ANALYTICS, Seema Acharya, Subhasinin Chellappan, Wiley publications.

Reference:
2. Planning for Big Data, O'Reilly Radar team.
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
JNTUH COLLEGE OF ENGINEERING HYDERABAD

III Year MCA                        V Semester

CLOUD COMPUTING

Prerequisites
- A course on “Computer Networks”
- A course on “Operating Systems”

Objectives
- This course provides an insight into cloud computing
- Topics include- distributed system models, service oriented architectures, cloud programming and software environments, resource management and cloud service models.

Outcomes
- Ability to understand various service delivery models of a cloud computing architecture.
- Ability to understand the ways in which the cloud can be programmed and deployed.
- Understanding Cloud Service Providers.


TEXT BOOKS:

REFERENCE BOOKS:
III Year MCA V Semester

Dept. Elective-II
DISTRIBUTED SYSTEMS

Prerequisites
- A course on “Operating Systems”
- A course on “Network Security and Cryptography”

Objectives
- This course provides an insight into Distributed systems.
- Topics include- Peer to Peer Systems, Transactions and Concurrency control,
  Security and Distributed shared memory

Outcomes
- Ability to understand Transactions and Concurrency control.
- Ability to understand Security issues.
- Understanding Distributed shared memory.
- Ability to design distributed systems for basic level applications.

UNIT I
Characterization of Distributed Systems-Introduction, Examples of Distributed systems, Resource sharing and web, challenges, System models-Introduction, Architectural and Fundamental models, Networking and Internetworking, Interprocess Communication, Distributed objects and Remote Invocation-Introduction, Communication between distributed objects, RPC, Events and notifications, Case study-Java RMI.

UNIT II

UNIT III

UNIT IV
Transactions and Concurrency control-Introduction, Transactions, Nested Transactions, Locks, Optimistic concurrency control, Timestamp ordering, Comparison of methods for concurrency control, Distributed Transactions-Introduction, Flat and Nested Distributed Transactions, Atomic commit protocols, Concurrency control in distributed transactions,
Distributed deadlocks, Transaction recovery, Replication-Introduction, System model and group communication, Fault tolerant services, Transactions with replicated data.

UNIT V
Security-Introduction, Overview of Security techniques, Cryptographic algorithms, Digital signatures, Case studies-Kerberos, TLS, 802.11 WiFi.
Distributed shared memory, Design and Implementation issues, Sequential consistency and Ivy case study, Release consistency and Munin case study, Other consistency models, CORBA case study-Introduction, CORBA RMI, CORBA Services.

TEXT BOOKS:

REFERENCE BOOKS:
Prerequisites

- Data Structures
- Knowledge on statistical methods

Objectives

- This course explains machine learning techniques such as decision tree learning, Bayesian learning etc.
- To understand computational learning theory.
- To study the pattern comparison techniques.

Outcomes

- Understand the concepts of computational intelligence like machine learning
- Ability to get the skill to apply machine learning techniques to address the real time problems in different areas
- Understand the Neural Networks and its usage in machine learning application.

UNIT -I
Introduction - Well-posed learning problems, designing a learning system Perspectives and issues in machine learning

Concept learning and the general to specific ordering – Introduction, A concept learning task, concept learning as search, Find-S: Finding a Maximally Specific Hypothesis, Version Spaces and the Candidate Elimination algorithm, Remarks on Version Spaces and Candidate Elimination, Inductive Bias.

Decision Tree Learning – Introduction, Decision Tree Representation, Appropriate Problems for Decision Tree Learning, The Basic Decision Tree Learning Algorithm Hypothesis Space Search in Decision Tree Learning, Inductive Bias in Decision Tree Learning, Issues in Decision Tree Learning.

UNIT -II
Artificial Neural Networks Introduction, Neural Network Representation, Appropriate Problems for Neural Network Learning, Perceptions, Multilayer Networks and the Back propagation Algorithm.
Discussion on the Back Propagation Algorithm, An illustrative Example: Face Recognition


UNIT -III
Bayesian learning - Introduction, Bayes Theorem, Bayes Theorem and Concept Learning Maximum Likelihood and Least Squared Error Hypotheses, Maximum Likelihood Hypotheses for Predicting Probabilities, Minimum Description Length Principle, Bayes Optimal Classifier, Gibbs Algorithm, Naïve Bayes Classifier, An Example: Learning to Classify Text, Bayesian Belief Networks, EM Algorithm.

Instance-Based Learning – Introduction, k-Nearest Neighbor Learning, Locally Weighted Regression, Radial Basis Functions, Case-Based Reasoning, Remarks on Lazy and Eager Learning.

UNIT -IV

Pattern Comparison Techniques, Temporal patterns, Dynamic Time Warping Methods, Clustering, Codebook Generation, Vector Quantization


UNIT -V

Analytical Learning – Introduction, Learning with Perfect Domain Theories : PROLOG-EBG Remarks on Explanation-Based Learning, Explanation-Based Learning of Search Control Knowledge, Using Prior Knowledge to Alter the Search Objective, Using Prior Knowledge to Augment Search Operations.

Combining Inductive and Analytical Learning – Motivation, Inductive-Analytical Approaches to Learning, Using Prior Knowledge to Initialize the Hypothesis.

Text Books:
1. Machine Learning – Tom M.Mitchell,-MGH
2. Fundamentals of Speech Recognition By Lawrence Rabiner and Biing – Hwang Juang.

Reference Books:
III Year MCA  V Semester

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
JNTUH COLLEGE OF ENGINEERING HYDERABAD

Prerequisites:

- Data Mining and Data Warehousing

Objectives:

- It enables the student understand the concepts of Business Intelligence

Outcome:

- The student understands Data Ware Housing, Data Mining for Business Intelligence, Business Rules and Business Intelligence Implementation.

1. UNIT-I

   Introduction to Business Intelligence-the Business pressure-Responses and support model-Definition of BI- Architecture of BI- Styles of BI-event-Driven alerts, A cyclic process of Intelligence Creation. The value of Business intelligence-Value driven and Information use-Performance metrics and key performance indicators-horizontal use cases for BI.
   Case Study: 1 (Efraim Turban Pg no 26 & 107).

2. UNIT-II

   Case Studies: 2 (Efraim Turban Pg no 125).

3. UNIT-III

   Case Studies: 3 (Efraim Turban Pg no 228)

4. UNIT-IV


5. UNIT-V

   Business intelligence implementation-Business Intelligence and integration implementation-connecting in BI systems- Issues of legality- Privacy and ethics- Social networking and BI.

Text Book

References:
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING  
JNTUH COLLEGE OF ENGINEERING HYDERABAD  

III Year MCA                                   V Semester  
Dept. Elective-II  
REAL TIME OPERATING SYSTEMS

Prerequisites:  
- Computer Organization  
- Microprocessors  
- Operating Systems

Objectives:  
- To provide students with a thorough understanding of the principles behind the structure and operation of real-time operating systems.

Outcomes:  
- Distinguish a real-time system from other systems  
- Identify the functions of operating system  
- Evaluate the need for real-time operating system  
- Implement the real-time operating system principles

UNIT – I:  
**Introduction to Real Time Operating Systems:** Brief History of OS, Defining an RTOS, The Scheduler, Objects, Services, Key Characteristics of an RTOS  
**Tasks:** Defining a Task, Task States and Scheduling, Typical Task Operations, Typical task Structure, Synchronization, Communication and Concurrency.  
**Semaphores:** Defining Semaphores, Operations and Use, Defining Message Queue, States, Content, Storage, Operations and Use

UNIT - II:  
**Message Queues:** Defining Message Queues, Message Queue states, Message Queue content, Message Queue Storage, Typical Message Queue operations, Typical Message Queue use  
**Kernel Objects:** Pipes, Event Registers, Signals, Condition Variables  
**RTOS Services:** Building Blocks, Component Configuration

UNIT - III:  
**Exceptions and Interrupts:** What are Exceptions and Interrupts, Application of exceptions and Interrupts, A closer look at Exceptions and Interrupts, Processing General Exceptions, Nature of Spurious Exceptions  
**Timer and Timer Services:** Real Time Clocks, Programmable Interval Timers, Timer Interrupt Service Routines, A Model for Implementing the Soft-Timer Handling Capacity, Timing Wheels

UNIT IV:  
**I/O Subsystem:** Basic I/O Concepts, The I/O subsystem  
**Memory Management:** Dynamic Memory allocation In Embedded Systems, Fixed Size Memory management, Blocking Vs NON-Blocking Memory Functions
UNIT V:
Synchronization and Communication: Synchronization, Communication, Resource Synchronization Methods, Common Practical Design Patterns
Case Studies: RT Linux, MicroC/OS-II, Vx Works

TEXT BOOKS:

REFERENCE BOOKS:
1. Embedded Systems- Architecture, Programming and Design by Rajkamal, 2007, TMH.
2. Advanced UNIX Programming, Richard Stevens
3. Embedded Linux: Hardware, Software and Interfacing – Dr. Craig Hollabaugh
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
JNTUH COLLEGE OF ENGINEERING HYDERABAD

III Year MCA                        V Semester

Dept. Elective-III
DISTRIBUTED DATABASES

Prerequisites
1. A course on “Database Management Systems”

Objectives
1. To acquire knowledge on parallel and distributed databases and its applications.
2. To study the usage and applications of Object Oriented databases.
3. To learn the modeling and design of databases.
4. To acquire knowledge on parallel and distributed databases and its applications.
5. Equip students with principles and knowledge of parallel and object oriented databases.
6. Topics include distributed DBMS architecture and design; query processing and optimization; distributed transaction management and reliability; parallel and object database management systems.

Outcomes
1. Understand theoretical and practical aspects of distributed database systems.
2. Study and identify various issues related to the development of distributed database system.
3. Understand the design aspects of object oriented database system and related development.
4. Ability to write global queries for distributed databases.

UNIT I
Features of Distributed versus Centralized Databases, Principles of Distributed Databases, Levels Of Distribution Transparency, Reference Architecture for Distributed Databases, Types of Data Fragmentation, Integrity Constraints in Distributed Databases, Distributed Database Design

UNIT II
Translation of Global Queries to Fragment Queries, Equivalence transformations for Queries, Transforming Global Queries into Fragment Queries, Distributed Grouping and Aggregate Function Evaluation, Parametric Queries, Optimization of Access Strategies, A Framework for Query Optimization, Join Queries, General Queries

UNIT III
The Management of Distributed Transactions, A Framework for Transaction Management, Supporting Atomicity of Distributed Transactions, Concurrency Control for Distributed Transactions, Architectural Aspects of Distributed Transactions, Concurrency Control, Foundation of Distributed Concurrency Control, Distributed Deadlocks, Concurrency Control based on Timestamps, Optimistic Methods for Distributed Concurrency Control.
UNIT IV
Reliability, Basic Concepts, Nonblocking Commitment Protocols, Reliability and concurrency Control, Determining a Consistent View of the Network, Detection and Resolution of Inconsistency, Checkpoints and Cold Restart, Distributed Database Administration, Catalog Management in Distributed Databases, Authorization and Protection

UNIT V
Architectural Issues, Alternative Client/Server Architectures, Cache Consistency, Object Management, Object Identifier Management, Pointer Swizzling, Object Migration, Distributed Object Storage, Object Query Processing, Object Query Processor Architectures, Query Processing Issues, Query Execution, Transaction Management, Transaction Management in Object DBMSs, Transactions as Objects
Database Integration, Scheme Translation, Scheme Integration, Query Processing Query Processing Layers in Distributed Multi-DBMSs, Query Optimization Issues Transaction Management Transaction and Computation Model, Multidatabase Concurrency Control, Multidatabase Recovery, Object Orientation and Interoperability, Object Management Architecture CORBA and Database interoperability, Distributed Component Object Model, COM/OLE and Database Interoperability, PUSH-Based Technologies

TEXT BOOKS:
1. Distributed Databases Principles & Systems, Stefano Ceri, Giuseppe Pelagatti, TMH.

REFERENCE BOOKS:
Pre-requisites:
- The course assumes a reasonable comfort and background about Information Technology and Management Information Systems.

Objectives:
- To gain understanding of the basic principles of service orientation
- To learn service oriented analysis techniques
- To learn technology underlying the service design
- To learn the concepts such as SOAP, Registering and Discovering Services.

Outcomes:
At the end of this course, students are expected to gain the following learning:
- Get the foundations and concepts of service based computing
- Advocate the importance and means of technology alignment with business
- Understanding the basic operational model of web services,
- Gain the knowledge of key technologies in the service oriented computing arena
- Apply and practice the learning through a real or illustrative project/case study.

UNIT- I
Evolution and Emergence of Web Services – Evolution of distributed computing. Core distributed computing technologies – client/server, CORBA, JAVA RMI, Micro Soft DCOM, MOM, Challenges in Distributed Computing, role of J2EE and XML in distributed computing, emergence of Web Services and Service Oriented Architecture (SOA). Introduction to Web Services – The definition of web services, basic operational model of web services, tools and technologies enabling web services, benefits and challenges of using web services.

UNIT -II
Web Service Architecture – Web services Architecture and its characteristics, core building blocks of web services, standards and technologies available for implementing web services, web services communication, basic steps of implementing web services. Describing Web Services – WSDL introduction, non functional service description, WSDL1.1 Vs WSDL 2.0, WSDL document, WSDL elements, WSDL binding, WSDL tools, WSDL port type, limitations of WSDL.

UNIT III
UNIT – IV
Registering and Discovering Services: The role of service registries, Service discovery, Universal Description, Discovery, and Integration, UDDI Architecture, UDDI Data Model, Interfaces, UDDI Implementation, UDDI with WSDL, UDDI specification, Service Addressing and Notification, Referencing and addressing Web Services, Web Services Notification.

UNIT - V

Text Books:
2. Developing Java Web Services, R. Nagappan, R. Skoczylas, R.P. Sriganesh, Wiley India.
3. Developing Enterprise Web Services, S. Chatterjee, J. Webber, Pearson Education.

Reference Books:
1. XML, Web Services, and the Data Revolution, F.P. Coyle, Pearson Education.
3. Java Web Services, D.A. Chappell & T. Jewell, O'Reilly, SPD.
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING  
JNTUH COLLEGE OF ENGINEERING HYDERABAD  

III Year MCA                        V Semester  
Dept. Elective-III  
SIMULATION AND MODELING  

Prerequisites  
1. A course on “Computer Oriented Statistical Methods”  

Objectives  
1. The course is intended to provide a thorough understanding of methods, techniques and tools for modeling, simulation and performance analysis of complex systems.  
2. The topics include system models and studies; random number generation; simulation of continuous and discrete systems; simulation of queuing systems and pert networks.  

Outcomes  
1. Acquire proficiency in constructing a model for a given system/set of data.  
2. Ability to generate and test random number variates and employ them in developing simulation models.  
3. Ability to infer from the model and apply the results to resolve issues in a real world environment.  

Unit-I: System Models and Studies  

Unit-II: Random Numbers  

Unit-III: Simulation of Continuous and Discrete Systems  
Simulation of Continuous Systems: A chemical reactor, Numerical integration vs. continuous system simulation, Selection of an integration formula, Runge-Kutta integration formulas, Simulation of a servo system, Simulation of a water reservoir system, Analog vs. digital simulation.  
Discrete System Simulation: Fixed time-step vs. event-to-event model, On simulating randomness, Generation of random numbers, Generation of non-uniformly distributed random numbers, Monte-Carlo computation vs. stochastic simulation.  

Unit-IV: System Simulation  
Simulation of Queueing Systems: Rudiments of queueing theory, Simulation of a single-server queue, Simulation of a two-server queue, Simulation of more general queues.  

105

**Unit-V:** Simulation Experimentation
Design and Evaluation of Simulation Experiments: Length of simulation runs, Variance reduction techniques, Experimental layout, Validation.
Simulation Languages: Continuous and discrete simulation languages, Continuous simulation languages, Block-structured continuous simulation languages, Expression-based languages, Discrete-system simulation languages, GPSS.

**Text Books**

**Reference Books**
III Year MCA                        V Semester
Dept. Elective-III
INFORMATION RETRIEVAL SYSTEMS

Prerequisites:
- Data Structures

Course Objectives:
- to learn the important concepts and algorithms in Information Retrieval Systems.
- To understand the data/file structures that are necessary to design, and implement Information Retrieval (IR) systems.

Outcomes:
- Can be able to use different information retrieval techniques in various application areas
- Can apply IR principles to locate relevant information in large collections of data
- Able to design different document clustering Algorithms.
- Can be able to implement retrieval systems for web search tasks.

UNIT I
Introduction:
Motivation, Basic Concepts, Past-Present and Future, the Retrieval Process
Modeling:

UNIT II
Retrieval Evaluation
Introduction, retrieval Performance Evaluation, Reference Collections
Query languages
Introduction, Keyword-Based Querying, Pattern Matching, Structural Queries, Query Protocols
Query Operations
Introduction, User Relevance Feedback, Automatic Local Analysis, Automatic global Analysis
Text Operations
Introduction, Document Preprocessing, Document Clustering, Text Compression, Comparing text Compression Techniques

UNIT III
Indexing and Searching
Introduction, Inverted Files, Other Indices for Text, Boolean queries, Sequential Searching, pattern Matching, Structural Queries, Compression
Searching the Web
Introduction, Challenges, Characterizing the Web, Search Engines, Browsing, Metasearches, Finding the Needle in the Haystack, Searching using Hyperlinks
UNIT IV
User Interfaces and Visualization

UNIT V
Multimedia IR: Models and Languages
Introduction, Data Modeling, Query Languages
Multimedia IR: Indexing and Searching

Text Books
1. Modern Information Retrieval By Yates and Neto Pearson Education.

Reference:
III Year MCA                        V Semester

INFORMATION SECURITY LAB

Prerequisites
> 1. A Course on “Computer Networks, Mathematics

Objectives
> To understand the fundamentals of Cryptography
> To understand the various key distribution and management schemes
> To apply algorithms used for secure transactions in real world applications

Outcomes
> Demonstrate the knowledge of cryptography and network security concepts and applications.
> Ability to implement different public and private key algorithms.
> Ability to implement the programs to demonstrate the vulnerabilities and security threats and mechanisms to counter them.

1. Write a program that can encrypt and decrypt using Caesar Cipher. Assume the key value \( K = 5 \).
2. Write a program to encrypt and decrypt using a 2 X 2 Hill Cipher.
3. Write a program that can encrypt and decrypt using a transposition technique, key value \( k = 4312567 \).
4. Write a program that can encrypt and decrypt using simple DES.
5. Using any one of the key management technique, write a program so that the sender and the receiver have the same secret key (Symmetric key encryption).
6. Write an RSA program for generating the public and private key, and for encrypting and decrypting the given plain text.
7. Write a Diffie-Hellman key exchange program for generating the public and private key.
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
JNTUH COLLEGE OF ENGINEERING HYDERABAD

III Year MCA                        VI Semester

TECHNICAL SEMINAR

Prerequisites

None.

Objectives

Outcomes

At the end of the course the student will be able to:

- Analyze the selected topic, organize the content and communicate to audience in an effective manner
- Practice the learning by self study
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
JNTUH COLLEGE OF ENGINEERING HYDERABAD

III Year MCA VI Semester

Project Stage II

Prerequisites
- None.

Objectives
- To identify a problem, analyse, design and code
- To demonstrate with sufficient case studies

Outcomes
At the end of the course the student will be able to:
- Ability to Synthesize and apply prior knowledge to designing and implementing solutions to open-ended computational problems while considering multiple realistic constraints.
- Ability to design and develop the software with SE practices and standards
- Ability to analyze database, network and application design methods
- Ability to evaluate the various validation and verification methods
- Ability to practice CASE tools for solving case studies
- Ability to analyzing professional issues, including ethical, legal and security issues, related to computing projects.