

**ACADEMIC REGULATIONS,  
COURSE STRUCTURE & SYLLABI  
FOR**

***M. Tech. (DATA SCIENCES)***

Two Year PG Day-Time Program  
(with effect from 2021 – 22)



**JNTUH SCHOOL OF INFORMATION TECHNOLOGY  
(AUTONOMOUS)  
JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD  
Kukatpally, Hyderabad – 500 085 TELANGANA.**

**SCHOOL OF INFORMATION TECHNOLOGY  
(AUTONOMOUS)  
JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD  
Kukatpally, Hyderabad – 500 085, Telangana (India)**

**ACADEMIC REGULATIONS OF  
M.TECH. (REGULAR/FULL TIME) PROGRAMMES, 2021-2022(R-21)  
CHOICE BASED CREDIT SYSTEM (CBCS)  
(WITH EFFECT FROM THE ACADEMIC YEAR 2021 – 2022)**

The Master of Technology (M.Tech.) Post Graduate Degree of the Jawaharlal Nehru Technological University Hyderabad (JNTUH) shall be conferred on candidates who are admitted to the program and who fulfill all the requirements for the award of the Degree.

JNTUH offers 2 Years (4 Semesters) Master of Technology (M.Tech.) Post Graduate Degree program, under Choice Based Credit System (CBCS) at its constituent Autonomous College – **JNTUH, SCHOOL OF INFORMATION TECHNOLOGY (JNTUH SIT)**, Hyderabad in the following specializations

S.No.	Specialization
1	Computer Science(CS)
2	Software Engineering(SE)
3	Computer Networks & Information Security(CNIS)
4.	Data Sciences(DS)

### 1. ELIGIBILITY FOR ADMISSIONS

- 1.1 Admission to the above shall be made subject to eligibility, qualification and specialization as prescribed by the University for each Program, from time to time.
- 1.2 Admissions shall be made on the basis of merit rank obtained by the qualifying candidate on the basis of Valid GATE score or at an Entrance Test (TSPGECET) conducted by TELANGANA State Government, subject to reservations prescribed by the University time to time through Directorate of admissions JNTUH.

### 2. AWARD OF M. Tech. DEGREE

- 2.1 A candidate shall be declared eligible for the award of the M. Tech. Degree, if candidate pursues a course of study in not less than two and not more than four academic years.
- 2.2 A candidate who fails to fulfill all the academic requirements for the award of the degree within four academic years from the year of his/her admission shall forfeit his/her seat in M. Tech. course.

### 3. M.TECH. PROGRAMS STRUCTURE

- 3.1 The M.Tech. Programs of SIT-JNTUH are of Semester Pattern, with 4 Semesters constituting 2 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).

- 3.2** UGC/ AICTE specified Definitions/ Descriptions are adopted appropriately for various terms and abbreviations used in these Academic Regulations/ Norms, which are as listed below.
- 3.3 Semester Scheme:** Each PG program is of 2 Academic Years (4 Semesters), with the year being divided into two Semesters of 22 weeks ( $\geq 90$  working days) each, 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 by UGC, and Curriculum/ Course Structure as suggested by AICTE are followed.
- 3.4 Credit Courses:** Subjects/ Courses are to be registered by a candidate in a Semester to earn Credits. Credits shall be assigned to each Subject/ Course in a L: P: C (Lecture Periods: Practical Periods: Credits) Structure, based on the following general pattern.
- i. One Credit - for One hour/ Week/ for Theory/ Lecture (L) Courses
  - ii. One Credit - for Two hours/ Week/ for Laboratory/ Practical (P) Courses
  - iii. Other student activities like study tour, guest lecture, conference/workshop participations, technical paper presentations and mandatory courses (Audit Courses) will not carry any credits.

#### 4. COURSE WORK

- 4.1** A candidate after securing admission shall pursue the M.Tech. in a minimum period of 2 Academic Years, and a maximum period of 4 Academic Years (starting from the Date of Commencement of I Year).
- 4.2** Each candidate shall register for and secure the specified number of Credits required for the completion and award of the M.Tech. Degree in respective specialization.
- 4.3** Each of I Year I Semester and II Semester offers 19 Credits (I Year =  $2 \times 19 = 38C$ ) and II Year I Semester and II Semester offer 16 Credits (II Year =  $2 \times 16 = 32C$ ) each, totaling to 70 Credits (70C) for the entire M.Tech. Program.
- 4.4** **The candidate shall register for all 70 credits and secure all the 70 credits.**

#### 5. COURSE REGISTRATION

- 5.1** A ‘Course Coordinator or Faculty Advisor’ shall be assigned to each candidate, who will advise him/her about the M.Tech. Program, its Course Structure and Curriculum, Choice/Option for Subjects/ Courses, based on his/her competence, progress, pre-requisites and interest.
- 5.2** Academic Section of the College invites ‘Registration Forms’ from candidate’s apriori (before the beginning of the Semester). The Registration Requests for any ‘CURRENT SEMESTER’ shall be completed BEFORE the commencement of SEEs (Semester End Examinations) of the ‘PRECEDING SEMESTER’.
- 5.3** A candidate can register, ONLY AFTER obtaining the ‘WRITTEN APPROVAL’ from his Course Coordinator, which should be submitted to the College Academic Section.

- 5.4 A candidate may be permitted to register for Subjects/ Course of his/her CHOICE with a typical total of 19 Credits per Semester (I & II Semesters): **Minimum being 16 Credits and Maximum being 22 Credits**, based on his PROGRESS and SGPA/ CGPA, and completion of the 'PRE-REQUISITES' as indicated for various Subjects/ Courses, in the Department Course Structure and Syllabus contents. A candidate must register all the CORE subjects/courses.
- 5.5 The candidate has to register for the audit course(s) and must pass the audit course(s) for successful completion of the degree. However the credits earned in the audit courses are not included in the computation of the SGPA/CGPA
- 5.6 Choice for 'additional Subjects/ Courses' to reach the Maximum Permissible Limit of 22 Credits (above the typical 19 Credit norm) must be clearly indicated, which needs the specific approval and signature of the Course Coordinator.
- 5.7 If the Candidate submits ambiguous choices or multiple options or erroneous entries during Registration for the Subject(s) / Course(s) under a given/ specified Course Group/ Category as listed in the Course Structure, only the first mentioned Subject/ Course in that Category will be taken into consideration.
- 5.8 Subject/ Course Options exercised are final and CAN NOT be changed. However, if the Subject/ Course that has already been listed for Registration (by the Course Coordinator) in a Semester could not be offered due to any unforeseen or unexpected reasons, then the Candidate shall be allowed to have alternate choice - either for a new Subject (subject to offering of such a Subject), or for another existing Subject (subject to availability of seats), which may be considered. Such alternate arrangements will be made by the Course Coordinator, with due notification and time framed schedule, within the FIRST WEEK from the commencement of Class-work for that Semester.
- 5.9 Dropping of Subjects/ Courses may be permitted, ONLY AFTER obtaining prior approval from the Course Coordinator/Faculty Advisor (subject to retaining a minimum of 16 C), 'within 15 Days of Time' from the beginning of the current Semester.
- 5.10 Candidates may register for NPTEL/SWAYAM as per the university rules.

## 6. SUBJECTS/ COURSES TO BE OFFERED

- 6.1 A Subject/ Course may be offered to the Candidates, ONLY IF a Minimum of 1/3<sup>rd</sup> of the Section Strength opt for the same.
- 6.2 More than ONE TEACHER may offer the SAME SUBJECT (Lab/ Practicals may be included with the corresponding Theory Subject in the same Semester) in any Semester. However, selection choice for candidates will be based on - 'FIRST COME FIRST SERVE Basis and CGPA Criterion'.
- 6.3 If more entries for Registration of a Subject come into picture, then the concerned Course Coordinator shall take necessary action, whether to offer such a Subject/ Course for TWO (or multiple) SECTIONS or NOT .
- 6.4 In case of options coming from Candidates of other Departments/ Branches/ Disciplines (not considering OPEN ELECTIVES), PRIORITY shall be given to the candidate of the 'Parent Department' first.

## 7. ATTENDANCE

- 7.1 The candidate shall put in a minimum of 75% attendance per semester independently for each of the course/subject registered.
- 7.2 Condonation of shortage of attendance up to 10% in each course/subject registered (65% and above and less than 75%) may be given by the College/school Academic Committee.
- 7.3 Shortage of Attendance below 65% shall not be condoned. Condonation of shortage of attendance shall be granted only on genuine and valid reasons on representation by the candidate with supporting evidence and by paying stipulated fee.
- 7.4 Candidates whose shortage of attendance is not condoned in any course/subject registered are not eligible to write their end semester examination of that course/subject, they get **DETAINED** in that course/subject. The candidate will have to repeat that course/subject as and when offered; in case if there are any Professional Electives and/or Open Electives, the same may also be re-registered if offered, however, if those electives are not offered in later semesters, then alternate electives may be chosen from the SAME set of ELECTIVE subjects offered under that category. In such a case candidate has to pay tuition fee for that course/subject.
- 7.5 A Candidate shall put in a minimum required attendance in at least three (3) theory subjects in each semester for promoting to next Semester. In order to qualify for the award of the M.Tech Degree, the candidate shall complete all the academic requirements of the subjects, as per the course structure.

## 8. EVALUATION

The performance of the candidate in each semester shall be evaluated subject-wise, with a maximum of 100 marks for theory and 100 marks for practicals, on the basis of Internal Evaluation and End Semester Examination.

- 8.1 Evaluation of a candidate in a course/subject involves both external and internal components (except audit course). External evaluation will be in the form of Semester End Examination (SEE) in a course/subject for which 60 marks are allocated. The remaining 40 marks are allocated to Continuous Internal Evaluation (CIE).

### 8.2 For theory subjects,

8.2.1 The continuous internal evaluation has two Mid Term-Examinations (each of 40 marks). They are conducted as one in the middle of the Semester and the other immediately after the completion of instruction. Each midterm examination shall be conducted for a total duration of 120 minutes. The best one will be considered.

8.2.2 The End semester Examination will be conducted in each subject for 60 marks. The question paper consists of 8 questions. Each of these questions may contain sub-questions. Each question carries 12 marks. The candidate will be required to answer 5 questions. The questions are to be prepared to cover the entire range of prescribed syllabi of the subjects and units.

### 8.3 For practical subjects,

8.3.1 60 marks shall be awarded based on the performance in the End Semester Examinations

- 8.3.2** 40 marks shall be awarded based on the day-to-day performance in the lab and the performance in internal lab examination.
- 8.4 For audit subjects,**
- 8.4.1** The evaluation is based on INTERNAL EVALUATION only. The evaluation has two Mid Term-Examinations (each of 40 marks). They are conducted as one in the middle of the Semester and the other immediately after the completion of instruction.
- 8.4.2** Each midterm examinations shall be conducted for a total duration of 120 minutes. **The best one will be considered.**
- 8.5 A candidate shall be deemed to have secured the minimum academic requirement in a subject if he secures a minimum of 40% of marks in the Semester End Examination and a minimum aggregate of 50% of the total marks in the Semester End Examination and Internal Evaluation taken together. A candidate has to secure 50% of the marks allotted in the internal evaluation for passing the AUDIT Course.**
- 8.6** In case the candidate does not secure the minimum academic requirement in any subject (as specified in 8.5) he has to reappear for the Semester End Examination in that subject.
- 8.7** A candidate can re-register for the subjects, if the internal mark secured by a candidate is less than 50% and failed in that subject for maximum of two subjects and should register within four weeks of commencement of the class work. In such a case, the candidate must re-register for the subjects and secure the required minimum attendance. The candidate's attendance in the re-registered subject(s) shall be calculated separately to decide upon his eligibility for writing the Semester End Examination in those subjects. In the event of the candidate taking another chance, his Internal Evaluation (internal) marks and Semester End Examination marks obtained in the previous attempt stands cancelled.

## 9. Evaluation of Project / Dissertation Work

Every candidate shall be required to submit the thesis or dissertation after taking up a topic approved by the School/College.

- 9.1 Registration of Project Work:** A candidate is permitted to register for the project work after satisfying the attendance requirement of all the subjects (theory and practical subjects). A candidate has to choose the topic thesis in the first 2 weeks of the II Year I Semester in consultation with the Internal/External guides. After 2 weeks candidate has to submit an abstract of work to be carried out to the Project Review Committee (PRC), which in turn allows the candidate to register for thesis work if it is satisfied with the abstract submitted by the candidate.
- 9.2** A Project Review Committee (PRC) shall be constituted with Course Coordinator, Guide and a faculty member from the school.
- 9.3** Only after obtaining the approval of Project Review Committee (PRC), the candidate can initiate the Project work.
- 9.4** If a candidate wishes to change his supervisor or topic of the project he can do so with the approval of a committee appointed by the Director, SIT. However, the committee shall examine whether the change of topic/supervisor leads to a major

- change of his initial plans of project proposal. If so, his date of registration for the project work starts from the date of change of Supervisor or topic as the case may be.
- 9.5** The total duration of the project is for 44 weeks which is spread across 22 weeks in II Year I semester and 22 Weeks in II year II semester. The candidate can submit the final project only after 40 weeks from the date of registration after the approval of PRC.
- 9.6** Internal Evaluation of the project shall be on the basis of the seminars (Project reviews) conducted during the II Year I semester and II semesters by the Project Review Committee (PRC). In II Year I semester two seminars are conducted and in II Year II semester two seminars are conducted.
- 9.7** At the end of the II Year I semester, a candidate shall submit status report in a spiral bound copy form. Internal marks are for 40 marks which will be awarded by PRC. External marks are for 60 marks which will be awarded by an external examiner appointed by the Director, SIT. Evaluation of project by PRC (for 40 marks) and by the external examiner (for 60 marks) will be done simultaneously. **One publication is mandatory in Journal or conference for submission of the thesis.**
- 9.8** At the end of the II Year II semester, after approval from the PRC, the soft copy of the thesis should be submitted for ANTI-PLAGIARISM for the quality check and the plagiarism report should be included in the final thesis. If the copied information is less than 24%, then only thesis will be accepted for submission. A candidate shall submit the thesis/dissertation in a hard bound copy form. He will attend for the viva-voce. An external examiner appointed by the Director will evaluate the project for 100 marks. There will be no internal marks for phase II of the project.
- 9.9** The candidate has to submit two hard copies and one soft copy of Thesis/Dissertation, certified in the prescribed format by the supervisor to the school.
- 9.10** The Thesis/Dissertation will be adjudicated by one external examiner selected by the competent authority.
- 9.11** In case the candidate fails in viva-voce examination, based on the recommendation of the board the candidate has to retake the viva-voce examination after three months. If he fails in this viva-voce examination also, he will not be eligible for the award of the degree unless the candidate is asked to revise and resubmit.

## 10. Examinations and Assessment - The Grading System

- 10.1** Grades will be awarded to indicate the performance of each candidate 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 8 above, and a corresponding Letter Grade shall be given.
- 10.2** As a measure of the candidate's performance, a 10-point Absolute Grading System using the following Letter Grades (UGC Guidelines) and corresponding percentage of marks shall be followed:

Marks Obtained	Grade	Description of Grade	Grade Points (GP)
$\geq 90$	O	Outstanding	10
$\geq 80$ and $< 90$	A+	Excellent	9

>=70 and <80	A	Very Good	8
>=60 and <70	B+	Good	7
>=50 and <60	B	Average	6
<50	F	Fail	0
Not Appeared the Exam(s)	AB	Absent	0

- 10.3** A candidate obtaining F Grade in any Subject shall be considered ‘failed’ and will 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 candidate not appeared for examination then ‘AB’ Grade will be allocated in any Subject shall be considered ‘failed’ and will be required to reappear as ‘Supplementary Candidate’ in the Semester End Examination (SEE), as and when offered.
- 10.5** A Letter Grade does not imply any specific Marks percentage and it will be the range of marks percentage.
- 10.6** In general, a candidate shall not be permitted to repeat any Subject/ Course (s) only for the sake of ‘Grade Improvement’ or ‘SGPA/ CGPA Improvement’.
- 10.7** A candidate earns Grade Point (GP) in each Subject/ Course, on the basis of the Letter Grade obtained by him in that Subject/ Course. 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 Subject**

- 10.8** The Candidate passes the Subject/ Course only when he gets **GP ≥ 6 (B Grade or above)**.
- 10.9** The Grade Point Average (GPA) is calculated by dividing the Sum of Credit Points ( $\Sigma CP$ ) secured from ALL Subjects registered in a Semester or for the Exam appeared (like supplementary), by the Total Number of Credits registered during that Semester or for the Exam appeared (like supplementary). GPA is rounded off to FOUR Decimal Places. GPA is thus computed as

$$GPA = \frac{\sum_1^n C_i \times GP_i}{\sum_1^n C_i}$$

Where n is the number of subjects Registered in that semester / exam.

$C_i$  is Credits for the subjects.

$GP_i$  is the grade point obtained for the subject

where ‘i’ is the Subject indicator index (takes into account all Subjects in a Semester or for the Exam appeared), ‘N’ is the no. of Subjects ‘REGISTERED’ for the Semester or for the Exam appeared,  $C_i$  is the no. of Credits allotted to the  $i^{\text{th}}$  Subject,



and  $G_i$  represents the Grade Points (GP) corresponding to the Letter Grade awarded for that  $i^{\text{th}}$  Subject.

- 10.10** The Cumulative Grade Point Average (CGPA) is a measure of the overall cumulative performance of a candidate over all Subjects in all considered for registration. The CGPA is the ratio of the Total Credit Points secured by a candidate in ALL registered Courses in ALL Semesters, and the Total Number of Credits registered in ALL the Semesters. CGPA is rounded off to FOUR Decimal Places. CGPA is thus computed as per the formula

$$\text{CGPA} = \frac{\sum_{j=1}^m \text{GPA}_j \times \text{TC}_j}{\sum_{j=1}^m \text{TC}_j}$$

where m is the number of subjects registered in the course.

$\text{TC}_j$  the total number of credits for a  $j^{\text{th}}$  subject.

$\text{GPA}_j$  is the Grade point of the  $j^{\text{th}}$  subject.

- 10.11** For Calculations listed in Item 10.6 – 10.10, 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.
- 10.12** For Final % of Marks equivalent to the computed CGPA, the following formula may be used. % of

$$\text{Marks} = (\text{CGPA} - 0.5) \times 10$$

## 11. AWARD OF DEGREE AND CLASS

- 11.1** A Candidate 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 (PGP), and secures the required number of **70** Credits (with CGPA  $\geq 6.0$ ), 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.
- 11.2 Award of Class:** After a candidate has satisfied the requirements prescribed for the completion of the Degree and is eligible for the award of M. Tech. Degree, he shall be placed in one of the following three classes based on the CGPA:

Class Awarded	CGPA
First Class with Distinction	$\geq 7.75$
First Class	$6.75 \leq \text{CGPA} < 7.75$
Second Class	$6.00 \leq \text{CGPA} < 6.75$

- 11.3** A candidate with final CGPA (at the end of the PGP) < 6.00 will not be eligible for the award of Degree.

## 12. WITHHOLDING OF RESULTS

- 12.1** If the candidate has not paid the dues, if any, to the University or if any case of indiscipline is pending against him, the result of the candidate will be withheld and he will not be allowed into the next semester. His degree will be withheld in such cases.

## 13. TRANSITORY REGULATIONS

- 13.1** If any candidate is detained due to shortage of attendance in one or more subjects, they are eligible for admission to maximum of two earlier or equivalent subjects at a time as and when offered.
- 13.2** In case any candidate makes a re-registration then the academic regulations which were applicable for the year of his joining year will be applicable.

## 14. GENERAL

- 14.1 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.
- 14.2 Credit Point:** It is the product of grade point and number of credits for a course.
- 14.3** Wherever the words “he”, “him”, “his”, occur in the regulations, they include “she”, “her”.
- 14.4** The academic regulation should be read as a whole for the purpose of any interpretation.
- 14.5** In the case of any doubt or ambiguity in the interpretation of the above rules, the decision of the Vice-Chancellor is final.
- 14.6** The University may change or amend the academic regulations or syllabi at any time and the changes or amendments made shall be applicable to all the candidates with effect from the dates notified by the University.

## 15. MALPRACTICES RULES

### 15.1 Disciplinary Action For / Improper Conduct In Examinations

S.No.	Nature of Malpractices/Improper conduct	Punishment
	<i>If the candidate:</i>	
I. (a)	Possesses or keeps accessible in examination hall, any paper, note book, programmable calculators, Cell phones, pager, palm computers or any other form of material concerned with or related to the subject of the examination (theory or practical) in which he is appearing but has not made use of (material shall include any marks on the body of the candidate which can be used as an aid in the subject of the examination)	Expulsion from the examination hall and cancellation of the performance in that subject only.
(b)	Gives assistance or guidance or receives it from any other candidate orally or by	Expulsion from the examination hall and cancellation of the performance in that

	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.	subject only of all the candidates involved. In case of an outsider, he will be handed over to the police and a case is registered against him.
2.	Has copied in the examination hall from any paper, book, programmable calculators, palm computers or any other form of material relevant to the subject of the examination (theory or practical) in which the candidate is appearing.	Expulsion from the examination hall and cancellation of the performance in that subject and all other subjects the candidate has already appeared including practical examinations and project work and shall not be permitted to appear for the remaining examinations of the subjects of that Semester/year. The Hall Ticket of the candidate is to be cancelled and sent to the University.
3.	Impersonates any other candidate in connection with the examination.	The candidate who has impersonated shall be expelled from examination hall. The candidate is also debarred and forfeits the seat. The performance of the original candidate who has been impersonated, shall be cancelled in all the subjects of the examination (including practicals and project work) already appeared and shall not be allowed to appear for examinations of the remaining subjects of that semester/year. The candidate is also debarred for two consecutive semesters from class work and all University examinations. The continuation of the course by the candidate is subject to the academic regulations in connection with forfeiture of seat. If the imposter is an outsider, he will be handed over to the police and a case is registered against him.
4.	Smuggles in the Answer book or additional sheet or takes out or arranges to send out the question paper during the examination or answer book or additional sheet, during or after the examination.	Expulsion from the examination hall and cancellation of performance in that subject and all the other subjects the candidate has already appeared including practical examinations and project work and shall not be permitted for the remaining examinations of the subjects of that semester/year. The candidate is also debarred for two consecutive semesters from class work and all University examinations. The continuation of the course by the candidate is subject to the academic regulations in connection with forfeiture of seat.
5.	Uses objectionable, abusive or offensive language in the answer paper or in letters to the examiners or writes to the examiner requesting him to award pass marks.	Cancellation of the performance in that subject.
6.	Refuses to obey the orders of the Chief Superintendent/Assistant Superintendent / any officer on duty or misbehaves or creates disturbance of any kind in and around the examination hall or organizes a walk out or instigates others to walk out, or threatens the officer-in charge or any person on duty in or outside the examination hall of any injury to his person or to any of his relations whether	In case of candidates of the college, they shall be expelled from examination halls and cancellation of their performance in that subject and all other subjects the candidate(s) has (have) already appeared and shall not be permitted to appear for the remaining examinations of the subjects of that semester/year. The candidates also are debarred and forfeit their seats. In case of outsiders, they will be handed over to the police and a police case is registered

	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.	against them.
7.	Leaves the exam hall taking away answer script or intentionally tears of the script or any part thereof inside or outside the examination hall.	Expulsion from the examination hall and cancellation of performance in that subject and all the other subjects the candidate has already appeared including practical examinations and project work and shall not be permitted for the remaining examinations of the subjects of that semester/year. The candidate is also debarred for two consecutive semesters from class work and all University examinations. The continuation of the course by the candidate is subject to the academic regulations in connection with forfeiture of seat.
8.	Possess any lethal weapon or firearm in the examination hall.	Expulsion from the examination hall and cancellation of the performance in that subject and all other subjects the candidate has already appeared including practical Examinations and project work and shall not be permitted for the remaining examinations of the subjects of that semester/year. The candidate is also debarred and forfeits the seat.
9.	If candidate 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.	Candidate of the colleges expulsion from the examination hall and cancellation of the performance in that subject and all other subjects the candidate has already appeared including practical examinations and project work and shall not be permitted for the remaining examinations of the subjects of that semester/year. The candidate is also debarred and forfeits the seat. Person(s) who do not belong to the College will be handed over to police and, a police case will be registered against them.
10.	Comes in a drunken condition to the examination hall.	Expulsion from the examination hall and cancellation of the performance in that subject and all other subjects the candidate has already appeared including practical examinations and project work and shall not be permitted for the remaining examinations of the subjects of that semester/year.
11.	Copying detected on the basis of internal evidence, such as, during valuation or during special scrutiny.	Cancellation of the performance in that subject and all other subjects the candidate has appeared including practical examinations and project work of that

		semester/year examinations.
12.	If any malpractice is detected which is not covered in the above clauses 1 to 11 shall be reported to the University for further action to award suitable punishment.	

### 15.2 Malpractices identified by squad or special invigilators

1. Punishments to the candidates as per the above guidelines.
2. Punishment for institutions:  
(if the squad reports that the college is also involved in encouraging malpractices)
  - (i) A show cause notice shall be issued to the college.
  - (ii) Impose a suitable fine on the college.
  - (iii) Shifting the examination centre from the college to another college for a specific period of not less than one year

**JNTUH SCHOOL OF INFORMATION TECHNOLOGY**  
(Autonomous)  
**MASTER OF TECHNOLOGY-DATA SCIENCES**  
**I YEAR - I SEMESTER**

S.No	Subject Code	Subject	L	T	P	Total Marks (100)		Credits
						Int	Ext	
1.	DS1C10	<b>Core-1:</b> Advanced Algorithms through Python	3	-	-	40	60	3
2.	DS1C20	<b>Core-2:</b> Data Science	3	-	-	40	60	3
3	DS1C30	<b>Core-3:</b> Mathematics for Data Science	3	-	-	40	60	3
4.	DS1E10	<b>Elective-1:</b>	3	-	-	40	60	3
	DS1E11	Internet Technologies and Services						
	DS1E12	Internet of Things						
	DS1E13	Data Preparation and Analysis						
	DS1E14	Data Warehousing and Data Mining						
	DS1E15	Python Programming						
5.	DS1E20	<b>Elective-2:</b>	3	-	-	40	60	3
	DS1E21	Advanced Computer Networks						
	DS1E22	Information Retrieval Systems						
	DS1E23	Artificial Intelligence						
	DS1E24	Data Storage Technologies and Networks						
	DS1E25	Blockchain Technology						
DS1E26	Data Visualization							
6.	DS1L10	<b>Lab-1:</b> Advanced Algorithms through Python Lab	-	-	4	40	60	2
7.	DS1L20	<b>Lab-2: (Based on Elective-1)</b>	-	-	4	40	60	2
	DS1L21	Internet Technologies and Services Lab						
	DS1L22	Internet of Things Lab						
	DS1L23	Data Preparation and Analysis Lab						
	DS1L24	Data Warehousing and Data Mining Lab						
	DS1L25	Python Programming Lab						
8.	DS1A10	<b>Audit-1:</b>	2	-	-	100	-	0
	DS1A11	Foundations of Computer Science						
	DS1A12	Professional Communication Skills						
	DS1A13	Personality Development through Life Enlightenment Skills						
	DS1A14	Value Education						
	DS1A15	Constitution of India						
<b>Total</b>			17	-	8			19

**JNTUH SCHOOL OF INFORMATION TECHNOLOGY**  
(Autonomous)  
**MASTER OF TECHNOLOGY-DATA SCIENCES**  
**I YEAR - II SEMESTER**

S.No	Subject Code	Subject	L	T	P	Total Marks (100)		Credits
						Int	Ext	
1.	DS2C10	<b>Core-4:</b> Machine Learning	3	-	-	40	60	3
2.	DS2C20	<b>Core-5:</b> Big Data	3	-	-	40	60	3
3.	DS2C30	<b>Core-6:</b> Natural Language Processing	3	-	-	40	60	3
4.	DS2E10	<b>Elective-3:</b>	3	-	-	40	60	3
	DS2E11	Mobile Application Development						
	DS2E12	Computer Forensics						
	DS2E13	Network programming						
	DS2E14	Advanced Programming						
	DS2E15	Cryptography and Network Security						
5.	DS2E20	<b>Elective-4:</b>	3	-	-	40	60	3
	DS2E21	Cloud Computing						
	DS2E22	Multimodal Analytics						
	DS2E23	Data Security and Access Control						
	DS2E24	Web Analytics and Development						
	DS2E25	Knowledge Discovery						
	DS2E26	Predictive Analytics						
6.	DS2L10	<b>Lab-3:</b> Machine Learning Lab	-	-	4	40	60	2
7.	DS2L20	<b>Lab-4: (Based on Elective-3)</b>	-	-	4	40	60	2
	DS2L21	Mobile Application Development Lab						
	DS2L22	Computer Forensics Lab						
	DS2L23	Network programming Lab						
	DS2L24	Advanced Programming Lab						
	DS2L25	Cryptography and Network Security Lab						
8.	DS2A10	<b>Audit-2:</b>	2	-	-	100	-	0
	DS2A11	English for Research Paper Writing						
	DS2A12	Disaster Management						
	DS2A13	Soft Skills						
	DS2A14	Stress Management by Yoga						
	DS2A15	Sanskrit for Technical Knowledge						
	DS2A16	Research Methodology						
	<b>Total</b>							

**JNTUH SCHOOL OF INFORMATION TECHNOLOGY**  
(Autonomous)  
**MASTER OF TECHNOLOGY-DATA SCIENCES**

**II YEAR - I SEMESTER**

S.No	Subject Code	Subject	L	T	P	Total Marks (100)		Credits
						Int	Ext	
1.	DS3E10	<b>Core-Elective 5:</b>	3	-	-	40	60	3
	DS3E11	Recommender Systems						
	DS3E12	Deep Learning						
	DS3E13	Cloud Security						
	DS3E14	GPU Computing						
	DS3E15	Distributed Databases						
	DS3E16	Adhoc Networks						
	DS3E17	Reinforcement Learning						
	DS3E18	MultiAgent Systems						
2.	DS3O10	<b>Open Elective-1:</b>	3	-	-	40	60	3
	DS3O11	Business Analytics						
	DS3O12	Industrial Safety						
	DS3O13	Operations Research						
	DS3O14	Cost management and Engineering Projects						
	DS3O15	Cyber Security						
	DS3O16	Composite Materials						
3.	DS3P10	<b>Project Work: Project Phase-1</b>	-	-	20	40	60	10
		<b>Total</b>	6	-	20			16

**II YEAR – II SEMESTER**

S.No	Subject Code	Subject	L	T	P	Total Marks (100)		Credits
						Int	Ext	
1.	DS4P10	<b>Project work: Project Phase-2</b>	-	-	32	-	100	16
		<b>TOTAL</b>	-	-	32			16



**MASTER OF TECHNOLOGY  
(DATA SCIENCES)  
I YEAR I SEMESTER**

**DS1C10**

**ADVANCED ALGORITHMS THROUGH PYTHON  
(CORE –1)**

**UNIT I**

Python Overview, Objects in Python, Expressions, Operators, Precedence, Control flow, Functions, Input and Output(Console Input and Output, Files), Exception handling, Iterators and Generators, Conditional Expressions, List comprehension, Packing and Unpacking of Sequences, Scopes and Namespaces, Modules and the Import Statement, Object Oriented Programming-

Object oriented design goals, Object oriented design Principles, Class Definitions, Inheritance, Namespaces and Object Orientation, Shallow and Deep Copying.

Algorithms, Algorithm analysis- time complexity and space complexity, Asymptotic Notation-Big Oh, Omega and Theta notations, Algorithm Analysis Examples.

Data structures-Linear and non linear data structures, ADT concept, Array based Sequences and Link based sequences, Singly linked lists and Doubly linked lists -Insertion, and Deletion operations, Circularly linked lists.

**UNIT II**

The Stack ADT, Array-Based Implementation(Python list),Singly linked list Implementation, infix to postfix conversion using Stack, Recursion-Examples, analyzing recursive algorithms, Examples of Recursion-Linear recursion, Binary recursion, and Multiple recursion, Eliminating tail recursion, Queue ADT, array and linked list Implementations, Circular queue-insertion and deletion, Deque ADT, Doubly linked list Implementation.

**UNIT III**

Searching–Linear and binary search methods, Maps and Dictionaries, Hash Tables-Hash functions, Collision Handling Schemes-Open Addressing, Separate Chaining, Python Hash Table Implementation, Sets, Multisets, and Multimaps.

Sorting –Bubble sort, Selection sort, Insertion sort, Quick sort, Merge sort, Comparison of sorting methods.

**UNIT IV**

Trees- Tree Definitions and Properties, Binary tree ADT, Properties of Binary trees, Array-Based representation, Linked structure for Binary trees, Binary tree traversal algorithms, Implementing Binary tree traversals in Python. Priority queue ADT, implementation using Heaps, Python Heap implementation, Sorting with a Priority Queue-Heap sort.

Graphs- Graphs terminology, Graph ADT, Data Structures for Graphs, Graph traversals-dfs and bfs, Applications of Graphs-Minimum cost spanning tree using Kruskal’s algorithm, Dijkstra’s algorithm for Single Source Shortest Path Problem.

**UNIT V**

Search trees- Binary search trees-Binary search tree ADT, insertion, deletion and searching operations, Python implementation, Balanced search trees - AVL trees-Definition and examples only, Red Black trees –Definition and examples only, B-Trees-definition and examples only, Tries (examples only), Comparison of Search trees.

Text Compression-Huffman coding and decoding, Pattern matching algorithms-KMP algorithm and Boyer-Moore Algorithm.

**TEXT BOOKS:**

1. Data structures and Algorithms in Python, M.T.Goodrich, R.Tomassia and Michael H. Goldwasser, An Indian Adaptation,Wiley India Pvt. Ltd.
2. Data structures and Algorithms using Python, Rance D.Necaise, Wiley Student Edition.

**REFERENCES:**

1. Introduction to Programming in Python, Robert Sedgewick, Kevin Wayne and Robert Dondero, Pearson.
2. Python Programming, Sheetal Taneja and Naveen Kumar, Pearson.
3. Exploring Python, Timothy A.Budd, Tata McGraw-Hill Edition.
4. Think Python, Allen B.Downey, O'Reilly,SPD.
5. Python Programming, Reema Thareja, Oxford University Press.

**MASTER OF TECHNOLOGY  
(DATA SCIENCES)  
I YEAR I SEMESTER**

**DS1C20**

**DATA SCIENCE  
(CORE – 2)**

**Objectives:**

- To understand about Data Science
- To understand big data, to learn the analytics of Big Data how data is stored and processed in Hadoop
- To learn about Machine Learning Algorithms
- To learn model evaluation and how data is analyzed using R features

**UNIT I**

**Introduction to Data, Data Science, Data Process:** Introduction to Data Science and data science process – Evolution of Data Science – Data Science Roles – Stages in a Data Science Project – Applications of Data Science in various fields ,data science profile , Types of Digital data: Classification of Digital Data,

**Introduction to Big Data:** What is big data, Evolution of Big Data, Traditional Business Intelligence vs Big Data, Coexistence of Big Data and Data Warehouse.

**UNIT II**

**Data Collection and Data Preprocessing:** Processing data with hadoop, interfacing with hadoop ecosystem.

Hadoop: Features of Hadoop, Key advantages of hadoop, versions of hadoop, overview of hadoop ecosystem, Hadoop distributions.

Why hadoop? RDBMS vs Hadoop, Distribution computing challenges, History of hadoop, Hadoop overview, HDFS NoSQL: Where it is used? What is it? Types of NoSQL Databases, Why NoSQL? Advantages of NoSQL, What we miss with NoSQL? Use of NoSQL in industry, SQL vs NoSQL.

**UNIT III**

**Exploratory Data Analytics:** Descriptive Statistics – Mean, Standard Deviation, dispersion, Skewness and Kurtosis , statistical-interference-Correlation Statistics – ANOVA.

**UNIT IV**

**Algorithms/Model Development:** Basic machine learning algorithms, Simple and Multiple Regression – naivebayes, k-.means ,KNN ,decision tree, random forest, LDA ,Prediction and Decision Making, Evaluation Metrics – Cross Validation – Overfitting.

**UNIT V**

**Data Visualization:** using R, What is R? Why use R for analytics? How to run R? First R example, functions a short programming example, some important R data structures, vectors, matrices, lists, R programming structures, Charts, pie –charst, Barchart, boxplots, scatterplots ,linechart, Histograms, scatterplots ,Box plot.

**TEXT BOOKS:**

1. BIG DATA and ANALYTICS, Seema Acharya, Subhashini Chellappan, Wiley Publications.
2. Cathy O'Neil and Rachel Schutt , "Doing Data Science", O'Reilly, 2015.

**MASTER OF TECHNOLOGY  
(DATA SCIENCES)  
I YEAR I SEMESTER**

**DS1C30**

**MATHEMATICS FOR DATA SCIENCE  
(CORE – 3)**

**Course Objective:**

- To understand the mathematical fundamentals that is prerequisite for a variety of courses like Data Science, Machine Learning, Deep Learning, Natural Language Processing and Reinforcement Learning.
- To study various mathematical concepts of Linear Algebra, Calculus, Probability, Statistics, Linear Regression.

**UNIT I**

**Introduction to Data Science:** Data Science, Process of Data Science: Collecting data, Storing data, processing data, Describing data, Statistical modeling, Algorithmic modeling. Why Data science is so popular today? Data science vs. Artificial Intelligence, the myths of Data Science, Engineering Data science Systems: CRISP-DM model, Programming tools.

**UNIT II**

**Linear Algebra:** Vector, Linear combinations, span, linear transformations and matrices, matrix multiplication, inverse matrices, dot product, cross product, Cramer's rule. Matrix decomposition- Determinant and trace, Eigenvectors and Eigenvalues, singular value decomposition. **Calculus:** Derivatives basic formulas, Chain rule and product rule, Euler's number e. Integration and the fundamental theorem of calculus, area with slope, Taylor series. Partial differentiation and gradients, Gradient of vector-valued functions, Gradient of matrices.

**UNIT III**

**Probability:** Sample space, events, properties, joint probability, conditional probability, multiplication principle, total probability theorem, Baye's theorem. **Discrete Random Variables:** Probability mass function, discrete distributions - Bernouli, Binomial, Poisson, Geometric, Expectation and Variance. **Continuous Random Variables:** Probability density function, Continuous distributions- uniform, normal, exponential distributions. Maximum likelihood estimate, Markov models.

**UNIT IV**

**Descriptive Statistics:** Types of Data, Describe qualitative data- frequency plots, relative frequency plots. Describing quantitative data- histograms, stem and leaf plots. Describing relationship between variables- scatter plots. **Measure of centrality** - mean, median, mode, sensitivity to outliers, effect of transformations. Percentiles -quartiles, quintiles, deciles. **Measure of spread** - range, IQR, variance and standard deviation, sensitivity to outliers, effect of transformations. Box plots.

**UNIT V**

**Inferential Statistics:** Distribution of sample statistics, parameter, Central limit theorem and implications. Likelihood of sample mean: chisquare distribution, find expected value of the error, variance of proposition, point estimator, interval estimator of  $\mu$  with known  $\sigma$ , unknown  $\sigma$ , T-distribution plots, z and t- variables. **Hypothesis Testing:** Hypothesis testing proportion (P), Type1 and type2 error plotting distribution. Linear Regression, Correlation and Covariance.

**TEXT BOOKS:**

1. Marc Peter Deisenroth, A. Aldo Faisal, Chen Soon Ong, *Mathematics for Machine Learning*, Cambridge University Press, 2020.
2. Trevor Hastie, Robert Tibshirani and Jerome Friedman, *The Elements of Statistical Learning*, Springer Verlag, 2001.

**REFERENCES:**

1. Introduction to Statistics and Data Analysis with Exercises, Solutions and Application in R  
Authors: Heumann, Christian, Schomaker, Michael, Shalabh, Publisher: Springer 2016
2. Applied Statistics and Probability for Engineers, Douglas C. Montgomery, George C. Runger, 2018, Willey.
3. Linear Algebra and its Applications, Gilbert Strang, 4<sup>th</sup> edition, Cengage Learning.
4. Probability and Statistics for Engineers (E Miller and John E. Freund), R.A. Johnson, C.B. Gupta, 7<sup>th</sup> edition, PHI.
5. Essence of Linear Algebra - [https://www.youtube.com/playlist?list=PLZHQObOWTQDPD3MizzM2xVFitgF8hE\\_ab](https://www.youtube.com/playlist?list=PLZHQObOWTQDPD3MizzM2xVFitgF8hE_ab)
6. Essence of Calculus - <https://www.youtube.com/playlist?list=PLZHQObOWTQDMsr9K-rj53DwVRMYO3t5Yr>
7. Advanced Engineering Mathematics, 10<sup>th</sup> edition, Erwin Kreyszig, Wiley.

**M.TECH (DATA SCIENCES)  
I YEAR I SEMESTER**

**DS1E11**

**INTERNET TECHNOLOGIES AND SERVICES  
(ELECTIVE – 1)**

**Objective:**

The student who has knowledge of programming with java should be able to develop web based solutions using multi-tier architecture. S/he should have good understanding of different technologies on client and server side components as Follows:

- Client Side: HTML5, CSS3, Javascript, Ajax, JQuery and JSON
- Server Side: Servlets, JSP
- Database: MySQL with Hibernate and Connection Pooling
- Framework: Struts with validation framework, Internationalization (I18N)
- SOA: Service Oriented Architecture, Web services fundamentals, Axis framework for WS

**UNIT I**

**Client Side Technologies:** Overview of HTML - Common tags, XHTML, capabilities of HTML5 Cascading Style sheets, CSS3 enhancements, linking to HTML Pages, Classes in CSS Introduction to JavaScripts, variables, arrays, methods and string manipulation, BOM/DOM (Browser/Document Object Model), accessing elements by ID, Objects in JavaScript Dynamic HTML with JavaScript and with CSS, form validation with JavaScript, Handling Timer Events Simplifying scripting with JQuery, JASON for Information exchange.

**UNIT II**

**Introduction to Java Servlets:** Introduction to Servlets: Lifecycle of a Servlet, Reading request and initialization parameters, Writing output to response, MIME types in response, Session Tracking: Using Cookies and Sessions, Steps involved in Deploying an application Database Access with JDBC and Connection Pooling Introduction to XML, XML Parsing with DOM and SAX Parsers in Java Ajax - Ajax programming with JSP/Servlets, creating XML Http Object for various browsers, Sending request, Processing response data and displaying it. Introduction to Hibernate

**UNIT III**

**Introduction to JSP:** JSP Application Development: Types of JSP Constructs (Directives, Declarations, Expressions, Code Snippets), Generating Dynamic Content, Exception Handling, Implicit JSP Objects, Conditional Processing, Sharing Data Between JSP pages, Sharing Session and Application Data, Using user defined classes with jsp:useBean tag, Accessing a Database from a JSP.

**UNIT IV**

**Introduction to Struts Framework:** Introduction to MVC architecture, Anatomy of a simple struts2 application, struts configuration file, Presentation layer with JSP, JSP bean, html and logic tag libraries, Struts Controller class, Using form data in Actions, Page Forwarding, validation framework, Internationalization.

**UNIT V**

**Service Oriented Architecture and Web Services:** Overview of Service Oriented Architecture – SOA concepts, Key Service Characteristics, Technical Benefits of a SOA Introduction to Web Services– The definition of web services, basic operational model of web services, basic steps of implementing web services. Core fundamentals of SOAP – SOAP Message Structure, SOAP encoding, SOAP message exchange models, Describing Web Services –Web Services life cycle, anatomy of WSDL Introduction to Axis– Installing axis web service framework, deploying a java web service on axis. Web Services Interoperability – Creating java and .Net client applications for an Axis Web Service

(Note: The Reference Platform for the course will be open source products Apache Tomcat Application Server, MySQL database, Hibernate and Axis)

**TEXT BOOKS:**

1. Web Programming, building internet applications, Chris Bates 3<sup>rd</sup> edition, WILEY Dreamtech .
2. The complete Reference Java 7<sup>th</sup> Edition , Herbert Schildt., TMH.
3. Java Server Pages, Hans Bergsten, SPD, O'Reilly.
4. Professional Jakarta Struts - James Goodwill, Richard Hightower, Wrox Publishers.
5. Developing Java Web Services, R. Nagappan, R. Skoczylas, R.P. Sriganesh, Wiley India, rp – 2008.
6. Understanding SOA with Web Services, Eric Newcomer and Greg Lomow, Pearson Edition – 2009
7. Java Web Service Architecture, James McGovern, Sameer Tyagi et al., Elsevier - 2009

**REFERENCES:**

1. Programming the world wide web, 4<sup>th</sup> edition, R.W. Sebesta, Pearson
2. Core SERVLETS AND JAVASERVER PAGES VOLUME 1: CORE TECHNOLOGIES , Marty Hall and Larry Brown Pearson
3. Internet and World Wide Web – How to program , Dietel and Nieto PHI/Pearson.
4. Jakarta Struts Cookbook , Bill Siggelkow, S P D O'Reilly.
5. Professional Java Server Programming, S. Allamaraju & others Apress (dreamtech).
6. Java Server Programming , Ivan Bayross and others, The X Team, SPD
7. Web Warrior Guide to Web Programming - Bai/Ekedaw - Cengage Learning.
8. Beginning Web Programming - Jon Duckett , WROX.



**M.TECH (DATA SCIENCES)  
I YEAR I SEMESTER**

**DS1E12**

**INTERNET OF THINGS  
(ELECTIVE – 1)**

**Objectives:**

- To introduce the terminology, technology and its applications
- To introduce the concept of M2M (machine to machine) with necessary protocols
- To introduce the hardware and working principles of various sensors used for IoT
- To introduce the Python Scripting Language which is used in many IoT devices
- To introduce the Raspberry PI platform, that is widely used in IoT applications
- To introduce the implementation of web based services on IoT devices

**UNIT I**

Introduction to Internet of Things –Definition and Characteristics of IoT, Physical Design of IoT – IoT Protocols, IoT communication models, IoT Communication APIs, IoT enabled Technologies – Wireless Sensor Networks, Cloud Computing, Big data analytics, Communication protocols, Embedded Systems, IoT Levels and Templates, Domain Specific IoTs – Home, City, Environment, Energy, Retail, Logistics, Agriculture, Industry, health and Lifestyle

**UNIT II**

Introduction to Python - Language features of Python, Data types, data structures, Control of flow, functions, modules, packaging, file handling, data/time operations, classes, Exception handling Python packages - JSON, XML, HTTPLib, URLLib, SMTPLib

**UNIT III**

IoT Physical Devices and Endpoints - Introduction to Raspberry Pi- Installation, Interfaces (serial, SPI, I2C), Programming – Python program with Raspberry PI with focus on interfacing external gadgets, controlling output, reading input from pins.

**UNIT IV**

Controlling Hardware- Connecting LED, Buzzer, Switching High Power devices with transistors, Controlling AC Power devices with Relays, Controlling servo motor, speed control of DC Motor, Using unipolar and bipolar Stepper motors

Digital input- Sensing push switch, pull-up and pull-down resistors, Rotary encoder, Using keypad, Using RTC

Sensors: Light sensor, temperature sensor with thermistor, voltage sensor, ADC and ADC, Temperature and Humidity Sensor DHT11, Read Switch, Distance Measurement with ultrasound sensor

**UNIT V**

IoT Physical Servers and Cloud Offerings – Introduction to Cloud Storage models and communication APIs Webservice – Web server for IoT, Cloud for IoT, Python web application framework Designing a RESTful web API

**TEXT BOOK:**

1. Internet of Things - A Hands-on Approach, Arshdeep Bahga and Vijay Madisetti, Universities Press, 2015, ISBN: 9788173719547
2. Getting Started with Raspberry Pi, Matt Richardson & Shawn Wallace, O'Reilly (SPD), 2014, ISBN: 9789350239759
3. Raspberry Pi Cookbook, Software and Hardware Problems and solutions, Simon Monk, O'Reilly (SPD), 2016, ISBN 7989352133895

**M.TECH (DATA SCIENCES)  
I YEAR I SEMESTER**

**DS1E13**

**DATA PREPARATION AND ANALYSIS  
(ELECTIVE – 1)**

**COURSE OBJECTIVE**

To prepare the data for analysis and develop meaningful Data Visualizations

**UNIT I:**

**Data Gathering and Preparation:**

Data formats, parsing and transformation, Scalability and real-time issues

**UNIT II:**

**Data Cleaning:**

Consistency checking, Heterogeneous and missing data, Data Transformation and Segmentation

**UNIT III:**

**Exploratory Analysis:**

Descriptive and comparative statistics, Clustering and association, Hypothesis Generation

**UNIT IV:**

**Visualization:**

Designing visualizations, Time series, Geolocated data, Correlations and connections, Hierarchies and networks, interactivity

**COURSE OUTCOMES**

**After completion of course, students would be:**

Able to extract the data for performing the Analysis.

**REFERENCES:**

1. Making sense of Data : A practical Guide to Exploratory Data Analysis and Data Mining, by Glenn J. Myatt

**M.TECH (DATA SCIENCES)  
I YEAR I SEMESTER**

**DS1E14**

**DATA WAREHOUSING AND DATA MINING  
(ELECTIVE – 1)**

**Objectives**

- To understand the principles of Data warehousing and Data Mining.
- To be familiar with the Data warehouse architecture and its Implementation.
- To know the Architecture of a Data Mining system.
- To understand the various Data preprocessing Methods.
- To perform classification and prediction of data.

**UNIT I**

Data Warehousing and Business Analysis: - Data warehousing Components –Building a Data warehouse – Data Warehouse Architecture – DBMS Schemas for Decision Support – Data Extraction, Cleanup, and Transformation Tools –Metadata – reporting – Query tools and Applications – Online Analytical Processing (OLAP) – OLAP and Multidimensional Data Analysis.

**UNIT II**

Data Mining: - Data Mining Functionalities – Data Preprocessing – Data Cleaning – Data Integration and Transformation – Data Reduction – Data Discretization and Concept Hierarchy Generation- Architecture Of A Typical Data Mining Systems- Classification Of Data Mining Systems.

Association Rule Mining: - Efficient and Scalable Frequent Item set Mining Methods – Mining Various Kinds of Association Rules – Association Mining to Correlation Analysis – Constraint-Based Association Mining.

**UNIT III**

Classification and Prediction: - Issues Regarding Classification and Prediction – Classification by Decision Tree Introduction – Bayesian Classification – Rule Based Classification – Classification by Back propagation – Support Vector Machines – Associative Classification – Lazy Learners – Other Classification Methods – Prediction – Accuracy and Error Measures – Evaluating the Accuracy of a Classifier or Predictor – Ensemble Methods – Model Section.

**UNIT IV**

Cluster Analysis: - Types of Data in Cluster Analysis – A 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 Object, Spatial, Multimedia, Text and Web Data:

Multidimensional Analysis and Descriptive Mining of Complex Data Objects – Spatial Data Mining – Multimedia Data Mining – Text Mining – Mining the World Wide Web.

**TEXT BOOK:**

1. Jiawei Han, Micheline Kamber and Jian Pei “Data Mining Concepts and Techniques”, Third Edition, Elsevier, 2011.

**REFERENCES:**

1. Alex Berson and Stephen J. Smith “Data Warehousing, Data Mining & OLAP”, Tata McGraw – Hill Edition, Tenth Reprint 2007.
2. K.P. Soman, Shyam Diwakar and V. Ajay “Insight into Data mining Theory and Practice”, Easter Economy Edition, Prentice Hall of India, 2006.
3. G. K. Gupta “Introduction to Data Mining with Case Studies”, Easter Economy Edition, Prentice Hall of India, 2006.
4. Pang-Ning Tan, Michael Steinbach and Vipin Kumar “Introduction to Data Mining”, Pearson Education, 2007.

**M.TECH (DATA SCIENCES)  
I YEAR I SEMESTER**

**DS1E15**

**PYTHON PROGRAMMING  
(ELECTIVE – 1)**

**Objectives:**

- To learn the basics of Python programming language
- To learn the object oriented programming concepts through python
- To learn GUI programming with different tools available in python
- To learn Django Framework

**UNIT I**

Basic features of Python-Interactive execution, comments, types, variables, operators, expressions, Statements-assignment, input, print, Control flow-Conditionals, Loops, break statement, continue statement, pass statement, Functions, definition, call, scope and lifetime of variables, keyword arguments, default parameter values, variable length arguments, recursive functions, Sequences-Strings, Lists and Tuples-basic operations and functions, iterating over sequences, Sets and Dictionaries- operations and functions, Python program examples.

**UNIT II**

Files-operations-opening, reading, writing, closing,file positions. Exceptions – raising and handling exceptions, try/except statements, finally clause, standard exceptions, custom exceptions. Functional programming-mapping, filtering and reduction, Lambda functions, List comprehensions. Scope, namespaces and modules, import statement, creating own modules, avoiding namespace collisions when importing modules, iterators and generators, Python program examples.

**UNIT III**

Object oriented programming- classes, constructors, objects, class variables, class methods, static methods, operator overloading. Inheritance-is-a relationship, composition, polymorphism, overriding, multiple inheritance, abstract classes, multithreaded programming, Python program examples.

**UNIT IV**

GUI Programming with Tkinter, Widgets (Buttons, Canvas, Frame, Label, Menu, Entry, Text, Scrollbar, Combobox, Listbox, Scale), event driven programming-events, callbacks, binding, layout managementgeometry managers: pack and grid, creating GUI based applications in Python.

**UNIT V**

Introduction to Django Framework Model Template View (MTV) framework, Creating a Project and Application, Configuring database, Defining a model, Defining a view, Defining a template, Defining a URL pattern, Enabling Admin site, Designing a RESTful API

**TEXT BOOKS:**

1. Exploring Python, Timothy A. Budd, McGraw Hill Publications.
2. Introduction to Programming using Python, Y.Daniel Liang, Pearson.
3. Python Programming, R.Thareja, Oxford University Press.
4. PythonProgramming, Sheetal Taneja and Naveen Kumar, Pearson.
5. Internet of Things - A hands on approach, Arshdeep Bahga and Vijay Madisetti, Universities Press, 2015.

**REFERENCE BOOKS:**

1. Introduction to Computer Science using Python, Charles Dierbach, Wiley India Edition.
2. Fundamentals of Python, K. A. Lambert, B.L. Juneja, Cengage Learning.
3. Think Python, how to think like a computer scientist, Allen B. Downey,SPD, O'Reilly.
4. www.python.org web site.
5. Official Django Document  
(<https://buildmedia.readthedocs.org/media/pdf/django/1.5.x/django.pdf>)

**M.TECH (DATA SCIENCES)  
I YEAR I SEMESTER**

**DS1E21**

**ADVANCED COMPUTER NETWORKS  
(ELECTIVE – 2)**

**Objectives:**

- The objective of this course is to build a solid foundation in computer networks concepts and design
- To understand computer network architectures, protocols, and interfaces.
- The OSI reference model and the Internet architecture network applications.
- The course will expose students to the concepts of traditional as well as modern day computer networks - wireless and mobile, multimedia-based.
- Students completing this course will understand the key concepts and practices employed in modern computer networking

**UNIT I**

**Computer Networks and the Internet:** What Is the Internet?, The Network Edge , The Network Core , Delay, Loss, and Throughput in Packet-Switched Networks , Protocol Layers and Their Service Models, Networks Under Attack

**Network Management:** What is Network Management?, The Infrastructure for Network Management, The Internet-Standard Management Framework

**UNIT II**

**Application Layer:** Principles of Network Applications, Electronic Mail in the Internet, DNS—The Internet's Directory Service

**Transport Layer:** Introduction and Transport-Layer Services, Multiplexing and Demultiplexing, Connectionless Transport: UDP, Principles of Reliable Data Transfer, Connection-Oriented Transport: TCP, Principles of Congestion Control, TCP Congestion Control.

**UNIT III**

**The Network Layer:** Introduction, Virtual Circuit and Datagram Networks, What's Inside a Router?, The Internet Protocol (IP): Forwarding and Addressing in the Internet , Routing Algorithms, Routing in the Internet, Broadcast and Multicast Routing.

**UNIT IV**

**The Link Layer:** Links, Access Networks, and LANs, Introduction to the Link Layer, Error-Detection and -Correction Techniques, Multiple Access Links and Protocols, Switched Local Area Networks, Link Virtualization: A Network as a Link Layer, Data Center Networking,



**UNIT V**

**Wireless and Mobile Networks:** Introduction, Wireless Links and Network Characteristics, WiFi: 802.11 Wireless LANs, Cellular Internet Access, **Mobility Management:** Principles, Mobile IP, Managing Mobility in Cellular Networks, **Wireless and Mobility:** Impact on Higher-Layer Protocols.

**TEXT BOOKS:**

1. Computer Networking: A Top Down Approach , *James F. Kurose, Keith W. Ross*, 6<sup>th</sup> Edition.
2. Data Communications and Networking, *Behrouz A. Forouzan*, Fourth Edition, Tata McGraw Hill
3. High Speed Networks and Internets – Performance and Quality of Service, *William Stallings*, Second Edition, Pearson Education.
4. Top-Down Network Design, *Priscilla Oppenheimer*, Second Edition, Pearson Education (CISCO Press)

**REFERENCES:**

1. Computer Networks by Mayank Dave, Cengage.
2. Guide to Networking Essentials, *Greg Tomsho, Ed Tittel, David Johnson*, Fifth Edition, Thomson.
3. Computer Networks, *Andrew S. Tanenbaum*, Fourth Edition, Prentice Hall.
4. An Engineering Approach to Computer Networking, *S.Keshav*, Pearson Education.
5. Campus Network Design Fundamentals, *Diane Teare, Catherine Paquet*, Pearson Education (CISCO Press)
6. Computer Communications Networks, Mir, Pearson Education.

**M.TECH (DATA SCIENCES)  
I YEAR I SEMESTER**

**DS1E22**

**INFORMATION RETRIEVAL SYSTEMS  
(ELECTIVE – 2)**

**UNIT I**

Introduction to Information Retrieval Systems : Definition of Information Retrieval System, Objectives of Information Retrieval System, Functional Overview, Relationship to Database Management Systems, Digital Libraries and Data Warehouses; Boolean retrieval. The term vocabulary and postings lists. Dictionaries and tolerant retrieval. Index construction. Index compression.

**UNIT II**

Scoring, term weighting and the vector space model. Computing scores in a complete search system. Evaluation in information retrieval. Relevance feedback and query expansion.

**UNIT III**

XML retrieval. Probabilistic information retrieval. Language models for information retrieval. Text classification. Vector space classification.

**UNIT IV**

Support vector machines and machine learning on documents. Flat clustering. Hierarchical clustering. Matrix decompositions and latent semantic indexing.

**UNIT V**

Web search basics. Web crawling and indexes. Link analysis.

**TEXT BOOKS:**

1. Introduction to Information Retrieval , Christopher D. Manning and Prabhakar Raghavan and Hinrich Schütze, Cambridge University Press, 2008.
2. Information Storage and Retrieval Systems: Theory and Implementation, Kowalski, Gerald, Mark T Maybury, Springer.

**REFERENCS :**

1. Modern Information Retrieval , Ricardo Baeza-Yates, Pearson Education, 2007.
2. Information Retrieval: Algorithms and Heuristics, David A Grossman and Ophir Frieder, 2nd Edition, Springer, 2004.
3. Information Retrieval Data Structures and Algorithms, William B Frakes, Ricardo Baeza- Yates, Pearson Education, 1992.
4. Information Storage & Retrieval , Robert Korfhage , John Wiley & Sons.

**M.TECH (DATA SCIENCES)  
I YEAR I SEMESTER**

**DS1E23**

**ARTIFICIAL INTELLIGENCE  
(ELECTIVE – 2)**

**Objectives:**

- To learn the difference between optimal reasoning Vs human like reasoning
- To understand the notions of state space representation, exhaustive search, heuristic search along with the time and space complexities
- To learn different knowledge representation techniques
- To understand the applications of AI: namely Game Playing, Theorem Proving, Expert Systems, Machine Learning and Natural Language Processing

**UNIT I**

Introduction: What is AI? Foundations of AI, History of AI, Agents and environments, The nature of the Environment, Problem solving Agents, Problem Formulation, Search Strategies

**UNIT II**

Knowledge and Reasoning: Knowledge-based Agents, Representation, Reasoning and Logic, Propositional logic, First-order logic, Using First-order logic, Inference in First-order logic, forward and Backward Chaining

**UNIT III**

Learning: Learning from observations, Forms of Learning, Inductive Learning, Learning decision trees, why learning works, Learning in Neural and Belief networks

**UNIT IV**

Practical Natural Language Processing: Practical applications, Efficient parsing, Scaling up the lexicon, Scaling up the Grammar, Ambiguity, Perception, Image formation, Image processing operations for Early vision, Speech recognition and Speech Synthesis

**UNIT V**

Robotics: Introduction, Tasks, parts, effectors, Sensors, Architectures, Configuration spaces, Navigation and motion planning, Introduction to AI based programming Tools

**TEXT BOOKS:**

1. Stuart Russell, Peter Norvig: “Artificial Intelligence: A Modern Approach”, 2nd Edition, Pearson Education, 2007

**REFERENCES:**

1. Artificial Neural Networks B. Yagna Narayana, PHI
2. Artificial Intelligence , 2nd Edition, E.Rich and K.Knight (TMH).
3. Artificial Intelligence and Expert Systems – Patterson PHI.
4. Expert Systems: Principles and Programming- Fourth Edn, Giarrantana/ Riley, Thomson.
5. PROLOG Programming for Artificial Intelligence. Ivan Bratka- Third Edition – Pearson Education.
6. Neural Networks Simon Haykin PHI

**M.TECH (DATA SCIENCES)  
I YEAR I SEMESTER**

**DS1E24**

**DATA STORAGE TECHNOLOGIES AND NETWORKS  
(ELECTIVE – 2)**

**COURSE OBJECTIVE**

to provide learners with a basic understanding of Enterprise Data Storage and Management Technologies

**UNIT I**

**Storage Media and Technologies** – Magnetic, Optical and Semiconductor Media, Techniques for read/write Operations, Issues and Limitations.

**UNIT II**

**Usage and Access** – Positioning in the Memory Hierarchy, Hardware and Software Design for Access, Performance issues.

**UNIT III**

**Large Storages** – Hard Disks, Networked Attached Storage, Scalability issues, 7 Model Curriculum of Engineering & Technology PG Courses [Volume-I], Networking issues.

**Architecture** - Storage Partitioning, Storage System Design, Caching, Legacy Systems.

**UNIT IV**

**Storage Area Networks** – Hardware and Software Components, Storage Clusters/Grids.

**Storage QoS**–Performance, Reliability, and Security issues.

**UNIT V**

Recent Trends related to Copy data management, Erasure coding, and Softwaredefined storage appliances.

**COURSE OUTCOMES**

**After completion of course, students would be:**

Learn Storage System Architecture  
Overview of Virtualization Technologies, Storage Area Network

**REFERENCES:**

1. The Complete Guide to Data Storage Technologies for Network-centric Computing Paperback–Import, Mar 1998 by Computer Technology Research Corporation
2. Data Storage Networking: Real World Skills for the CompTIA Storage by Nigel Poulton

**M.TECH (DATA SCIENCES)  
I YEAR I SEMESTER**

**DS1E25**

**BLOCKCHAIN TECHNOLOGY  
(ELECTIVE – 2)**

**Objectives:**

- Familiarise the functional/operational aspects of cryptocurrency ECOSYSTEM.
- Understand Blockchain Technology Architecture and components.
- Smart Contracts and Bitcoins
- Understand Blockchain and other technologies

**UNIT I**

**Basics of Blockchain:** Introduction, Concept of Blockchain, History, Definition of Blockchain, Fundamentals of Blockchain, Characteristics of Blockchain, Consensus in Trust-Building Exercise, Public, Private, and Hybrid Blockchains, Distributed Ledger Technologies, DLT Decentralized Applications and Databases, Architecture of Blockchain, Transactions, Chaining Blocks, Value Proposition of Blockchain Technology

**UNIT II**

**Architecture of Blockchain:** Architecture of Blockchain, Transactions, Chaining Blocks, Value Proposition of Blockchain Technology, **Consensus:** Introduction, Consensus Approach, Consensus Algorithms, Byzantine Agreement Methods

**UNIT III**

**Blockchain Components:** Introduction, Ethereum, History, Ethereum Virtual Machine, Working of Ethereum, Ethereum Clients, Ethereum Key Pairs, Ethereum Addresses, Ethereum Wallets, Ethereum Transactions, Ethereum Languages, Ethereum Development Tools

**UNIT IV**

**Smart Contracts:** Introduction, Smart Contracts, Absolute and Immutable, Contractual Confidentiality, Law Implementation and Settlement, Characteristics, Internet of Things, **Bitcoins:** Introduction, Working of Bitcoin, Merkle Trees, Bitcoin Block Structure, Bitcoin Address, Bitcoin Transactions, Bitcoin Network, Bitcoin Wallets, Bitcoin Payments, Bitcoin Clients, Bitcoin Supply

**UNIT V**

**Blockchain and Allied Technologies:** Blockchain and Cloud Computing, Characteristics of Blockchain Cloud, Blockchain and Artificial Intelligence, Blockchain and IoT, Blockchain and Machine Learning, Blockchain and Robotic Process Automation

**TEXTBOOKS:**

1. Blockchain Technology: Concepts and Applications. Kumar Saurabh, Ashutosh Saxena, Wiley

**REFERENCES:**

1. Blockchain Technology, Chandramouli Subramanian, Asha A George, Abhilash K A and Meena Karthikeyan, Universities Press
2. Blockchain Basics: A Non-Technical Introduction in 25 Steps by Daniel Drescher, Apress
3. Joseph Bonneau et al, SoK: Research perspectives and challenges for Bitcoin and cryptocurrency, IEEE Symposium on security and Privacy, 2015 ( article available for free download) { curtain raiser kind of generic article, written by seasoned experts and pioneers}.
4. J.A.Garay et al, The bitcoin backbone protocol - analysis and applications EUROCRYPT 2015 LNCS VOI 9057, ( VOLII ), pp 281-310. ( Also available at [eprint.iacr.org/2016/1048](http://eprint.iacr.org/2016/1048)) . ( serious beginning of discussions related to formal models for bitcoin protocols).
5. R.Pass et al, Analysis of Blockchain protocol in Asynchronous networks , EUROCRYPT 2017, ( [eprint.iacr.org/2016/454](http://eprint.iacr.org/2016/454)) . A significant progress and consolidation of several principles).
6. R.Pass et al, Fruitchain, a fair blockchain, PODC 2017 ( [eprint.iacr.org/2016/916](http://eprint.iacr.org/2016/916)).

**M.TECH (DATA SCIENCES)  
I YEAR I SEMESTER**

**DS1E26**

**DATA VISUALIZATION  
(ELECTIVE – 2)**

**COURSE OBJECTIVE**

- Familiarize students with the basic and advanced techniques of information visualization and scientific visualization,
- To learn key techniques of the visualization process
- A detailed view of visual perception, the visualized data and the actual visualization, interaction and distorting techniques

**UNIT I**

Introduction of visual perception, visual representation of data, Gestalt principles, information overloads. Creating visual representations, visualization reference model, visual mapping, visual analytics, Design of visualization applications.

**UNIT II**

Classification of visualization systems, Interaction and visualization techniques misleading, Visualization of one, two and multi-dimensional data, text and text documents.

**UNIT III**

Visualization of groups, trees, graphs, clusters, networks, software, Metaphorical visualization

**UNIT IV**

Visualization of volumetric data, vector fields, processes and simulations, Visualization of maps, geographic information, GIS systems, collaborative visualizations, Evaluating visualizations

**UNIT V**

Recent trends in various perception techniques, various visualization techniques, data structures used in data visualization.

**REFERENCES:**

1. WARD, GRINSTEIN, KEIM,.Interactive Data Visualization: Foundations, Techniques, and Applications. Natick : A K Peters, Ltd.
2. E. Tufte, The Visual Display of Quantitative Information, Graphics Press.

**M.TECH (DATA SCIENCES)  
I YEAR I SEMESTER**

**DS1L10**

**ADVANCED ALGORITHMS THROUGH PYTHON LAB  
(LAB-1)**

**Sample Problems :**

1. Write Python programs that use both recursive and non-recursive functions for implementing the following searching methods:
  - a) Linear search
  - b) Binary search
2. Write Python program to implement the following using arrays(Lists) and linked lists
  - a) List ADT
3. Write Python programs to implement the following using an array(List).
  - a) Stack ADT
  - b) Queue ADT
4. Write a Python program that reads an infix expression and converts the expression to postfix form. (use stack ADT).
5. Write a Python program to implement circular queue ADT .
6. Write a Python program that uses both a stack and a queue to test whether the given string is a palindrome or not.
7. Write Python programs to implement the following using a singly linked list.
  - a) Stack ADT
  - b) Queue ADT
8. Write Python programs to implement the deque (double ended queue) ADT using Doubly linked list.
9. Write a Python program to implement priority queue ADT.
10. Write a Python program to perform the following operations:
  - a) Construct a binary search tree of elements.
  - b) Search for a key element in the above binary search tree.
  - c) Delete an element from the above binary search tree.
11. Write a Python program to implement all the functions of a dictionary (ADT) using Hashing.
12. Write a Python to implement Dijkstra's algorithm for Single source shortest path problem.
13. Write Python programs that use recursive and non-recursive functions to traverse the given binary tree in
  - a) Preorder
  - b) Inorder
  - c) Postorder.
14. Write Python programs for the implementation of bfs and dfs for a given graph.
15. Write Python programs for implementing the following sorting methods:
  - a) Bubble sort
  - b) Insertion sort
  - c) Quick sort
  - d) Merge sort
  - e) Heap sort
  - g) Binary tree sort



16. Write a Python program that implements Kruskal's algorithm to generate minimum cost spanning tree.
17. Write a Python program that implements KMP algorithm for pattern matching.
18. Write a Python Program that reads lines of text from a file and makes frequency count of words. It should then display words followed by their frequency count.

**REFERENCES:**

1. Data structures and Algorithms using Python, Rance D.Necaise, Wiley Student Edition.
2. Introduction to Programming in Python, Robert Sedgewick, Kevin Wayne and Robert Dondero, Pearson.
3. Python Programming, Sheetal Taneja and Naveen Kumar, Pearson.
4. Exploring Python, Timothy A.Budd, Tata McGraw-Hill Edition.
5. Think Python, Allen B.Downey, O'Reilly,SPD.
6. Python Programming, Reema Thareja, Oxford University Press.  
Introduction to Programming using Python, Y. Daniel Liang, Pearson

**M.TECH (DATA SCIENCES)  
I YEAR I SEMESTER**

**DS1L21**

**INTERNET TECHNOLOGIES AND SERVICES LAB  
(ELECTIVE -1 LAB)  
(LAB-2)**

**Objectives:**

- Write syntactically correct HTTP messages and describe the semantics of common HTTP methods and header fields
- Discuss differences between URIs, URNs, and URLs, and demonstrate a detailed understanding of http-scheme URLs, both relative and absolute
- Describe the actions, including those related to the cache, performed by a browser in the process of visiting a Web address
- Install a web server and perform basic administrative procedures, such as tuning communication parameters, denying access to certain domains, and interpreting an access log
- Write a valid standards-conformant HTML document involving a variety of element types, including hyperlinks, images, lists, tables, and forms
- Use CSS to implement a variety of presentation effects in HTML and XML documents, including explicit positioning of elements
- Demonstrate techniques for improving the accessibility of an HTML document

**List of Sample Problems:**

**i) Internet Technologies**

1. Develop static pages (using Only HTML) of an online Book store. The pages should resemble: [www.amazon.com](http://www.amazon.com) the website should consist the following pages.

Home page, Registration and user Login

User Profile Page, Books catalog

Shopping Cart, Payment By credit card

Order Conformation

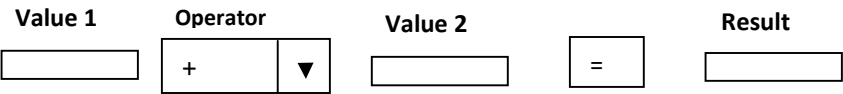
2. Validate the Registration, user login, user profile and payment by credit card pages using JavaScript.

3. Create and save an XML document at the server, which contains 10 users information. Write a program, which takes User Id as an input and returns the user details by taking the user information from the XML document.

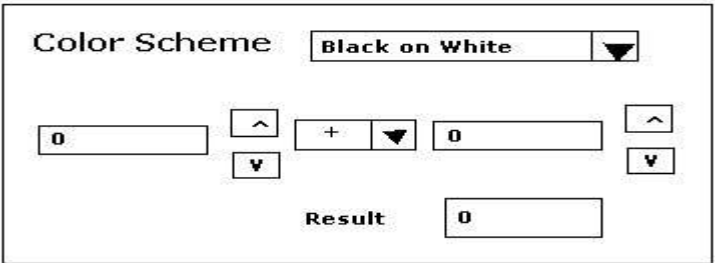
4. Install TOMCAT web server. Convert the static web pages of assignments 2 into dynamic web pages using Servlets and cookies. Hint: Users information (user id, password, credit card number) would be stored in web.xml. Each user should have a separate Shopping Cart.
5. Redo the previous task using JSP 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.
6. Implement the “Hello World!” program using JSP Struts Framework.

## ii) Additional Assignment Problems

1. Write an HTML page including any required Javascript that takes a number from one text field in the range of 0 to 999 and shows it in another text field in words. If the number is out of range, it should show “out of range” and if it is not a number, it should show “not a number” message in the result box.
2. Write a java swing application that takes a text file name as input and counts the characters, words and lines in the file. Words are separated with white space characters and lines are separated with new line character.
3. Write a simple calculator servlet that takes two numbers and an operator (+, -, /, \* and %) from an HTML page and returns the result page with the operation performed on the operands. It should check in a database if the same expression is already computed and if so, just return the value from database. Use MySQL or PostgreSQL.
4. Write an HTML page that contains a list of 5 countries. When the user selects a country, its capital should be printed next to the list. Add CSS to customize the properties of the font of the capital (color, bold and font size).
5. Write a servlet that takes name and age from an HTML page. If the age is less than 18, it should send a page with “Hello <name>, you are not authorized to visit this site” message, where <name> should be replaced with the entered name. Otherwise it should send “Welcome <name> to this site” message.
6. Write a calculator program in HTML that performs basic arithmetic operations (+, -, /, \* and %). Use CSS to change the foreground and background color of the values, buttons and result display area separately. Validate the input strings using JavaScript regular expressions. Handle any special cases like division with zero reasonably. The screen may look similar to the following:



7. Write a Java program that creates a calculator GUI, as shown in figure. Extra components may be added for convenience:



The Color Scheme may be Black on White or Blue on Yellow (selectable) and accordingly all components colors must be changed. The values can be either entered or increased or decreased by a step of 10. The operators are

+, -, / and \* (selectable). Once any change takes place, the result must be automatically computed by the program.

8. Write a Java Application that will read an XML file that contains personal information (Name, Mobile Number, age and place. It reads the information using SAX parser. After reading the information, it shows two input Text Fields in a window, one for tag name and the other for value. Once these two values are given, it should list all the records in the XML file that match the value of the given field in a text area (result box). For example, if the two text boxes are entered with “name” and “ABCD” then it should show all the records for which name is “ABCD”? An Illustration is given below that takes a mobile number and lists all the records that have the same mobile number.



9. Consider the following web application for implementation:

The user is first served a login page which takes user's name and password. After submitting the details the server checks these values against the data from a database and takes the following decisions.

- a. If name and password matches, serves a welcome page with user's full name.
- b. If name matches and password doesn't match, then serves “password mismatch” page

c. If name is not found in the database, serves a registration page, where users full name, present user name (used to login) and password are collected. Implement this application in:

1. Pure JSP
2. Pure Servlets
3. Struts Framework

10. Implement a simple arithmetic calculator with +, -, /, \*, % and = operations using Struts Framework. The number of times the calculator is used should be displayed at the bottom (use session variable).

### Internet Technologies and Services Lab – Additional Problems

1. Create a web Service in Java that takes two city names from the user and returns the distance between these two from data available from a table in MySQL.
2. Write a java and a C# client which use the above service
3. Write a Java program that takes a file as input and encrypts it using DES encryption. The program should check if the file exists and its size is not zero.
4. Write a Java program that generates a key pair and encrypts a given file using RSA algorithm.
5. Write a Java program that finds digest value of a given string.  
Consider the following xml file for encryption  
abc123 Replace and values with the RSA encrypted values represented with base64 encoding assuming that the public key is available in a file in local directory “pubkey.dat”. Encrypt with AES algorithm with a password ‘secret’. The checksum of all the field values concatenated with a delimiter character ‘+’ will be inserted in the checksum and the xml file is written to encrypted.xml file.
6. Write an HTML page that gives 3 multiple choice (a,b,c and d) questions from a set of 5 preloaded questions randomly. After each question is answered change the color of the question to either green or blue using CSS.
7. Finally on clicking OK button that is provided, the score should be displayed as a pop-up window. Use Java Script for dynamic content.
8. Write an HTML page that has 3 countries on the left side (“USA”, “UK” and “INDIA”) and on the right side of each country, there is a pull-down menu that contains the following entries: (“Select Answer”, “New Delhi”, “Washington” and “London”). The user will match the Countries with their respective capitals by selecting an item from the menu. The user chooses all the three answers (whether right or wrong). Then colors of the countries should be changed either to green or to red depending on the answer. Use CSS for changing color.
9. Write an HTML Page that can be used for registering the candidates for an entrance test. The fields are: name, age, qualifying examination (diploma or 10+2), stream in qualifying examination. If qualifying examination is “diploma”, the stream can be “Electrical”, “Mechanical” or “Civil”. If the qualifying examination is 10+2, the stream can be “MPC” or “BPC”. Validate the name to accept only characters and spaces.
10. Write an HTML page that has two selection menus. The first menu contains the states (“AP”, “TN” and “KN”) and depending on the selection the second menu should show the following items: “Hyderabad”, “Vijayawada”, “Kurnool” for AP, “Chennai”, “Salem”, “Madurai” for TN and “Bangalore”, “Bellary”, “Mysore” for KN.

11. Write an HTML page that has phone buttons 0 to 9 and a text box that shows the dialed number. If 00 is pressed at the beginning, it should be replaced with a + symbol in the text box. If the number is not a valid international number (+ followed by country code and 10 digit phone number) the color of the display should be red and it should turn to green when the number is valid. Consider only "+91, +1 and +44 as valid country codes. Use CSS for defining colors.
12. Write an HTML page that has a text box for phone number or Name. If a number is entered in the box the name should be displayed next to the number. If 00 is pressed at the beginning, it should be replaced with a + symbol in the text box. If a name is entered in the text box, it should show the number next to the name. If the corresponding value is not found, show it in red and show it in green otherwise. Use CSS for colors. Store at least 5 names and numbers in the script for testing.
13. A library consists of 10 titles and each title has a given number of books initially. A student can take or return a book by entering his/her HTNo as user ID and a given password. If there are at least two books, the book is issued and the balance is modified accordingly.
  - Use RDBMS and implement it with JSP.
  - Use XML File for data and Implement it with JSP
  - Use RDBMS and implement it with Servlets
  - Use XML File for data and Implement it with Servlets
14. A Bus Reservation System contains the details of a bus seat plan for 40 seats in 2×2 per row arrangement, where the seats are numbered from 1 to 40 from first row to last row. The customer can visit the website and can reserve a ticket of his choice if available by entering his details (Name, Address, Gender and Age). The customer can cancel the ticket by entering the seat number and his name as entered for reservation.
  - Use RDBMS and implement it with JSP.
  - Use XML File for data and Implement it with JSP
  - Use RDBMS and implement it with Servlets
  - Use XML File for data and Implement it with Servlets.
15. Implement a simple messaging system with the following details:  
When a student logs in with his/her HTNO and a given password, they should get all the messages posted to him/her giving the ID of sender and the actual message. Each message may be separated with a ruler. There should be a provision for the user to send a message to any number of users by giving the IDs separated with commas in the "To" text box.
  - Use RDBMS and implement it with JSP.
  - Use XML File for data and Implement it with JSP
  - Use RDBMS and implement it with Servlets
  - Use XML File for data and Implement it with Servlets.
16. There is an image of 600×100 size which can be logically divided into 12 button areas with labels (0-9, +, =).
17. Write a javascript calculator program that uses this image as input virtual keyboard and three text areas for two input numbers and result of sum of these numbers. Add a CSS that can be used to change the colors of text and background of text areas and the page. The input numbers can be up to 4 digits each. Develop a web application that takes user name and password as input and compares them with those available in an xml user database. If they match, it should display the welcome page that contains the user's full name and last used date and time retrieved from a client cookie. On logout it stores new time to the cookie and

displays a goodbye page. If authentication fails, it should store the attempt number to the client cookie and displays an error page. Add necessary CSS that takes care of the font, color of foreground and background.

18. A web application has the following specifications:

The first page (Login page) should have a login screen where the user gives the login name and password. Both fields must be validated on client side for a minimum length of 4 characters, name should be lower case az characters only and password should contain at least one digit. On submitting these values, the server should validate them with a MySQL database and if failed, show the login page along with a message saying “Login Name or Password Mismatch” in Red color below the main heading and above the form. If successful, show a welcome page with the user’s full name (taken from database) and a link to Logout. On logout, a good bye page is displayed with the total time of usage (Logout time – login time). Specify the Schema details of table and web.xml file contents. Implement it using (a) JSP Pages (b) Servlets (c) Struts

19. Design a struts based web portal for an international conference with following specifications:

The welcome page should give the details of the conference and a link to login. If login fails, direct them back for re-login and also provide a link for registration. On successful registration/login, the user will be directed to a page where s/he can see the status (accepted/rejected) of their already submitted papers followed by a form for submitting a doc file to the conference. Provide a logout button on all pages including the home page, once the user logs in. Implement validation framework to check that the user name is in the form of CCDDCC and password is in the form of (CCSDDD) (C for character, S for special character (one of @, #, \$, %, ^, & and !)) and D for digit)., Database should be accessed through Connection Pool for MySql for user information. Provide scope for internationalization in future. Assume any missing information and mention it first.

**M.TECH (DATA SCIENCES)  
I YEAR I SEMESTER**

**DS1L22**

**INTERNET OF THINGS LAB  
(ELECTIVE - 1 LAB)  
(LAB-2)**

***Python Basic exercises***

1. Write a Python program that reads 10 integers from keyboard and prints the average of even numbers and odd numbers separately
2. Write a Python program that prints the grade of a student when internal and external marks are given. A candidate is declared Failed (Grade = F), if Total marks < 50 or External marks < 25.

If a candidate is passed, then Grade is given as follows:

Condition	Grade
50 <= total marks < 60	E
60 <= total marks < 70	D
70 <= total marks < 80	C
80 <= total marks < 90	B
total marks >= 90	A

3. Create a table in MySQL that stores the status of devices in a house with the following data (Device ID, Device Name and Device State, last altered date and time). Now write a Python program that reads and alters the state of a given device. The date format is “YYYY-MM-DD:HH-mm-ss” where mm is minutes and ss is seconds.
4. Write a Python program that loads all the states of the devices into a dictionary from the table mentioned above.
5. Write a Python program that sorts the device states based on the last altered time
6. Write a Python program that reads a string from keyboard and prints the count of each alphabet in the string.
7. Write a Python program that reads a page from internet and prints it on the screen.
8. Write a Python program that reads and modifies an XML file
9. Write a Python program that reads and alters JSON data from a database table
10. Write a client-server Python program that uses socket connection to implement a time server. The client will connect to the server and the server sends the current time as “YYYY-MM-DD:HH-mm-ss” format. This value should be printed on the client side.



11. Write a Python program that generates 10 random numbers and stores them in a text file one per line. Now write another Python program that reads this data into a list and shows them
12. Write a program that reads key-value pair data from a file and stores them in a database table
13. Write a Python program that reads a time string in the format of “YYYY-MM-DD:HH-mm-ss” and prints its components separately.
14. Write a Python program that reads data from a table and writes it to a text file using tab as field separator and new line as record separator and vice versa.

### ***Raspberry Pi Experiments:***

*Use Raspberry Pi for all the experiments*

1. Connect an LED to GPIO pin 25 and control it through command line
2. Connect an LED to GPIO pin 24 and a Switch to GPIO 25 and control the LED with the switch. The state of LED should toggle with every press of the switch
3. Use DHT11 temperature sensor and print the temperature and humidity of the room with an interval of 15 seconds
4. Use joystick and display the direction on the screen
5. Use Light Dependent Resistor (LDR) and control an LED that should switch-on/off depending on the light.
6. Create a traffic light signal with three colored lights (Red, Orange and Green) with a duty cycle of 5-2-10 seconds.
7. User rotary encoder and print the position of the shaft on the console
8. Control a servo motor angle that is taken from the keyboard
9. Switch on and switch of a DC motor based on the position of a switch
10. Convert an analog voltage to digital value and show it on the screen.
11. Create a door lock application using a reed switch and magnet and give a beep when the door is opened.
12. Control a 230V device (Bulb) with Raspberry Pi using a relay
13. Control a 230V device using a threshold temperature, using temperature sensor.
14. Simulate an earthquake alarm using vibration sensor and give an alarm when vibration is detected.
15. Create an application that has three LEDs (Red, Green and white). The LEDs should follow the cycle (All Off, Red On, Green On, White On) for each clap (use sound sensor).
16. Create a web application for the above applications wherever possible with suitable modifications to get input and to send output.

**M.TECH (DATA SCIENCES)  
I YEAR I SEMESTER**

**DS1L23**

**DATA PREPARATION AND ANALYSIS LAB  
(ELECTIVE -1 LAB)  
(LAB-2)**

**Objectives:**

- To make students understand learn about a Big Data –R Programming , way of solving problems.
- To teach students to write programs in Rstudio to solve problems with predefined R Packages. On synthetic datasets

**Introduction to R Programming:**

What is R and RStudio? R is a statistical software program. It has extremely useful tools for data exploration, data analysis, and data visualization. It is flexible and also allows for advanced programming. RStudio is a user interface for R, which provides a nice environment for working with R.

1. Write an R program to evaluate the following expression  $ax+b/ax-b$ .
2. Write an R program to read input from keyboard (hint: readLine()).
3. Write an R program to find the sum of n natural numbers:  $1+2+3+4+\dots+n$
4. Write an R program to read n numbers.
  - (i) Sum of all even numbers (ii) Total number of even numbers.
5. Write an R program to read n numbers.
  - (i) Total number of odd numbers (ii) Sum of all odd numbers
6. Write an R program to obtain
  - (i) sum of two matrices A and B (ii) subtraction of two matrices A and B
  - (iii) Product of two matrices.
7. Write an R program for “ declaring and defining functions “
8. Write an R program that uses functions to add n numbers reading from keyboard
9. Write an R program uses functions to swap two integers.
10. Write an R program that use both recursive and non-recursive functions for implementing the Factorial of a given number, n .
11. Write an R program to reverse the digits of the given number . { example 1234 to
12. Write an R program to implement
  - (i) Linear search (ii) Binary Search.
13. Write an R program to implement
  - (i) Bubble sort (ii) selection sort .
14. Write a R program to implement the data structures
  - (i) Vectors (ii) Array (iii) Matrix (iv) Data Frame (v) Factor
15. **Graphical Analysis:** Creating a simple graph, Modifying the plots and lines of a graph. Modifying Title and Subtitle, Modifying Axes of a Graph, Adding Legend on a Graph, Special Graphs (Using pie charts, barplot, histograms scatter plots), Multiple Plots.
16. **Descriptive Statistics:** Measure of Central Tendency (Mean, Median and Mode), Measure of Positions (Quartiles, Deciles, Percentiles and Quantiles), Measure of Dispersion (Range, Median,

Absolute deviation about median, Variance and Standard deviation), Measure of Distribution (Skewness and Kurtosis), Box and Whisker Plot (Box Plot and its parts, Using Box Plots to compare distribution).

**17.**Construct classification and regression trees

**18.**Implement K-Means and Hierarchical

**19.** Experiments based on Linear Regression and Multiple Linear Regression Methods.

**20.**Implement K-nearest neighbors and Linear Discriminant Analysis.

**21.** Implement Leave-One-Out Cross-Validation and k-Fold cross-validation on any simulated data set.

**M.TECH (DATA SCIENCES)  
I YEAR I SEMESTER**

**DS1L24**

**DATA WAREHOUSING AND DATA MINING LAB  
(ELECTIVE -1 LAB)  
(LAB-2)**

**List of Sample Problems:**

Task 1: Credit Risk Assessment

**Description:**

The business of banks is making loans. Assessing the credit worthiness of an applicant is of crucial importance. You have to develop a system to help a loan officer decide whether the credit of a customer is good, or bad. A bank's business rules regarding loans must consider two opposing factors. On the one hand, a bank wants to make as many loans as possible. Interest on these loans is the banks profit source. On the other hand, a bank cannot afford to make too many bad loans. Too many bad loans could lead to the collapse of the bank. The bank's loan policy must involve a compromise: not too strict, and not too lenient.

To do the assignment, you first and foremost need some knowledge about the world of credit. You can acquire such knowledge in a number of ways.

1. Knowledge Engineering. Find a loan officer who is willing to talk. Interview her and try to represent her knowledge in the form of production rules.
2. Books. Find some training manuals for loan officers or perhaps a suitable textbook on finance. Translate this knowledge from text form to production rule form.
3. Common sense. Imagine yourself as a loan officer and make up reasonable rules which can be used to judge the credit worthiness of a loan applicant.
4. Case histories. Find records of actual cases where competent loan officers correctly judged when, and when not to, approve a loan application.

**The German Credit Data:**

Actual historical credit data is not always easy to come by because of confidentiality rules. Here is one such dataset, consisting of 1000 actual cases collected in Germany. credit dataset (original) Excel spreadsheet version of the German credit data.

In spite of the fact that the data is German, you should probably make use of it for this assignment. (Unless you really can consult a real loan officer !)

A few notes on the German dataset

- DM stands for Deutsche Mark, the unit of currency, worth about 90 cents Canadian (but looks and acts like a quarter).
- owns\_telephone. German phone rates are much higher than in Canada so fewer people own telephones.
- foreign\_worker. There are millions of these in Germany (many from Turrkey). It is very hard to get German citizenship if you were not born of German parents.

- There are 20 attributes used in judging a loan applicant. The goal is to classify the applicant into one of two categories, good or bad.

**Subtasks : (Turn in your answers to the following tasks)**

1. List all the categorical (or nominal) attributes and the real-valued attributes separately.
2. What attributes do you think might be crucial in making the credit assessment ? Come up with some simple rules in plain English using your selected attributes.
3. One type of model that you can create is a Decision Tree - train a Decision Tree using the complete dataset as the training data. Report the model obtained after training.
4. Suppose you use your above model trained on the complete dataset, and classify credit good/bad for each of the examples in the dataset. What % of examples can you classify correctly ? (This is also called testing on the training set) Why do you think you cannot get 100 % training accuracy ?
5. Is testing on the training set as you did above a good idea ? Why or Why not ?
6. One approach for solving the problem encountered in the previous question is using cross-validation ? Describe what is cross-validation briefly. Train a Decision Tree again using cross-validation and report your results. Does your accuracy increase/decrease ? Why ?
7. Check to see if the data shows a bias against "foreign workers" (attribute 20), or "personal-status" (attribute 9). One way to do this (perhaps rather simple minded) is to remove these attributes from the dataset and see if the decision tree created in those cases is significantly different from the full dataset case which you have already done. To remove an attribute you can use the preprocess tab in Weka's GUI Explorer. Did removing these attributes have any significant effect? Discuss.
8. Another question might be, do you really need to input so many attributes to get good results? Maybe only a few would do. For example, you could try just having attributes 2, 3, 5, 7, 10, 17 (and 21, the class attribute (naturally)). Try out some combinations. (You had removed two attributes in problem 7. Remember to reload the arff data file to get all the attributes initially before you start selecting the ones you want.)
9. Sometimes, the cost of rejecting an applicant who actually has a good credit (case 1) might be higher than accepting an applicant who has bad credit (case 2). Instead of counting the misclassifications equally in both cases, give a higher cost to the first case (say cost 5) and lower cost to the second case. You can do this by using a cost matrix in Weka. Train your

Decision Tree again and report the Decision Tree and cross-validation results. Are they significantly different from results obtained in problem 6 (using equal cost)?

10. Do you think it is a good idea to prefer simple decision trees instead of having long complex decision trees? How does the complexity of a Decision Tree relate to the bias of the model?
11. You can make your Decision Trees simpler by pruning the nodes. One approach is to use Reduced Error Pruning - Explain this idea briefly. Try reduced error pruning for training your Decision Trees using cross-validation (you can do this in Weka) and report the Decision Tree you obtain? Also, report your accuracy using the pruned model. Does your accuracy increase?
12. (Extra Credit): How can you convert a Decision Trees into "if-then-else rules". Make up your own small Decision Tree consisting of 2-3 levels and convert it into a set of rules. There also exist different classifiers that output the model in the form of rules - one such classifier in Weka is rules.PART, train this model and report the set of rules obtained. Sometimes just one attribute can be good enough in making the decision, yes, just one! Can you predict what attribute that might be in this dataset? OneR classifier uses a single attribute to make decisions (it chooses the attribute based on minimum error). Report the rule obtained by training a one R classifier. Rank the performance of j48, PART and oneR.

### Task Resources:

- Mentor lecture on Decision Trees
- Andrew Moore's Data Mining Tutorials (See tutorials on Decision Trees and Cross Validation)
- Decision Trees (Source: Tan, MSU)
- Tom Mitchell's book slides (See slides on Concept Learning and Decision Trees)
- Weka resources:
  - Introduction to Weka (html version) (download ppt version)
  - Download Weka
  - Weka Tutorial
  - ARFF format
  - Using Weka from command line

### Task 2: Hospital Management System

Data Warehouse consists Dimension Table and Fact Table.

REMEMBER The following

Dimension

The dimension object (Dimension):

- \_ Name
- \_ Attributes (Levels) , with one primary key
- \_ Hierarchies

One time dimension is must.

#### About Levels and Hierarchies

Dimension objects (dimension) consist of a set of levels and a set of hierarchies defined over those levels. The levels represent levels of aggregation. Hierarchies describe parent-child relationships among a set of levels.

For example, a typical calendar dimension could contain five levels. Two hierarchies can be defined on these levels:

H1: YearL > QuarterL > MonthL > WeekL > DayL

H2: YearL > WeekL > DayL

The hierarchies are described from parent to child, so that Year is the parent of Quarter, Quarter the parent of Month, and so forth.

#### About Unique Key Constraints

When you create a definition for a hierarchy, Warehouse Builder creates an identifier key for each level of the hierarchy and a unique key constraint on the lowest level (Base Level)

Design a Hospital Management system data warehouse (TARGET) consists of Dimensions Patient, Medicine, Supplier, Time. Where measures are ' NO UNITS', UNIT PRICE.

Assume the Relational database (SOURCE) table schemas as follows

TIME (day, month, year),

PATIENT (patient\_name, Age, Address, etc.,)

MEDICINE ( Medicine\_Brand\_name, Drug\_name, Supplier, no\_units, Uunit\_Price, etc.,)

SUPPLIER :( Supplier\_name, Medicine\_Brand\_name, Address, etc., )

If each Dimension has 6 levels, decide the levels and hierarchies, Assume the level names suitably.

Design the Hospital Management system data warehouse using all schemas. Give the example 4-D cube with assumption names.

#### **Similar Tasks Can Be Framed**

**M.TECH (DATA SCIENCES)  
I YEAR I SEMESTER**

**DS1L25**

**PYTHON PROGRAMMING LAB  
(ELECTIVE -1 LAB)  
(LAB-2)**

**Note: The problems given below are only sample problems.**

1. Write a Python program that reads a list of names and ages, then prints the list sorted by age.
2. Write a Python program that will prompt the user for a file name, read all the lines from the file into a list, sort the list, and then print the lines in sorted order.
3. Write a Python program that asks the user for a file name, and then prints the number of characters, words, and lines in the file.
4. Write a Python program that will prompt the user for a string and a file name, and then print all lines in the file that contain the string.
5. Create a class Rectangle. The constructor for this class should take two numeric arguments, which are the length and breadth. Add methods to compute the area and perimeter of the rectangle, as well as methods that simply return the length and breadth. Add a method isSquare that returns a Boolean value if the Rectangle is a Square.
6. Write a class Complex for performing arithmetic with complex numbers. The constructor for this class should take two floating-point values. Add methods for adding, subtracting, and multiplying two complex numbers.
7. Write a Python program that converts a fully parenthesized arithmetic expression from infix to postfix.
8. Write a Python program that reads a postfix expression from standard input, evaluates it, and writes the value to standard output.
9. Write a Python program that implements binary search method to search for a key in a sorted list.

**Sample problems covering data structures:**

10. Write Python program to implement the List ADT using a linked list.
11. Write Python programs to implement the deque (double ended queue) ADT using a) Array b) Singly linked list c) Doubly linked list.
12. Write a Python program that counts the occurrences of words in a text file and displays the words in decreasing order of their occurrence counts.
13. Write a Python program that implements insertion sort for sorting a list of elements in ascending order



**GUI applications:**

1. Write a Python program that works as a simple calculator. Use a grid to arrange buttons for the digits and for the +, -, \*, % operations. Add a text field to display the result.
2. Develop a Python GUI application that receives an integer in one text field, and computes its factorial Value and fills it in another text field, when the button named “Compute” is clicked.
3. Write a Python program that creates a user interface to perform integer divisions. The user enters two numbers in the text fields, Num1 and Num2. The division of Num1 and Num2 is displayed in the Result field when the Divide button is clicked. If Num1 or Num2 were not an integer Num2 is Zero, the program should Display an appropriate message in the result field in Red color.
4. Write a Python program that simulates a traffic light. The program lets the user select one of three lights: red, yellow, or green. When a radio button is selected, the light is turned on, and only one light can be on at a time.No light is on when the program starts.

**Advanced Problems:**

1. Create a table in MySQL that stores the status of devices in a house with the following data (Device ID, Device Name and Device State, last altered date and time). Now write a Python program that reads and alters the state of a given device. The date format is “YYYY-MM-DD:HH-mm-ss” where mm is minutes and ss is seconds
2. Write a Python program that loads all the states of the devices into a dictionary from the table mentioned above.
3. Write a Python program that sorts the device states based on the last altered time
4. Write a Python program that reads a string from keyboard and prints the count of each alphabet in the string.
5. Write a Python program that reads a page from internet and prints it on the screen
6. Write a Python program that reads and modifies an XML file
7. Write a Python program that reads and alters JSON data from a database table
8. Write a client-server Python program that uses socket connection to implement a time server. The client will connect to the server and the server sends the current time as “YYYY-MM-DD:HH-mm-ss” format. This value should be printed on the client side.
9. Write a Python program that generates 10 random numbers and stores them in a text file one per line. Now write another Python program that reads this data into a list and shows them
10. Write a program that reads key-value pair data from a file and stores them in a database table
11. Write a Python program that reads a time string in the format of “YYYY-MM-DD:HH-mm-ss” and prints its components separately.

12. Write a Python program that reads data from a table and writes it to a text file using tab as field separator and new line as record separator and vice versa.
13. Create a Django web application for a simple calculator with basic operations (+, -, \* and /) with two numbers.
14. Create a Django web application that implements Library MIS, which has the features like
  - Add/Delete a book
  - Issue a book to a person
  - Collect a book from a person
  - Search for a title or author
15. Create a Django web application that implements a bus reservation system, where a new bus can be added/removed with a given source and destination. A user should be able to reserve or cancel a seat.

**TEXT BOOKS:**

1. Exploring Python, Timothy A. Budd, McGraw Hill Publications.
2. Introduction to Programming using Python, Y.Daniel Liang, Pearson.
3. Python Programming, R.Thareja, Oxford University Press.
4. Python Programming, Sheetal Taneja and Naveen Kumar, Pearson.
5. Internet of Things - A hands on approach, Arshdeep Bahga and Vijay Madiseti, Universities Press, 2015.
6. [www.python.org](http://www.python.org) web site.
7. Official Django Document  
(<https://buildmedia.readthedocs.org/media/pdf/django/1.5.x/django.pdf>)

**M.TECH (DATA SCIENCES)  
I YEAR I SEMESTER**

**DS1A11**

**FOUNDATIONS OF COMPUTER SCIENCE  
(AUDIT – 1)**

**Objectives**

- To understand the fundamental concepts underlying OOP and apply them in solving problems.
- To learn to write programs in Java for solving problems.
- To understand the functions of Operating systems.
- To understand the Operating system services.
- To learn to use SQL to create, query and update the data in databases.

**UNIT I**

**Java Basics** - Java buzzwords, comments, data types, variables, constants, scope and life time of variables, operators, operator hierarchy, expressions, type conversion and casting, control flow- block scope, conditional statements, loops, break and continue statements, simple java program, arrays, input and output, formatting output, Key attributes of Object Oriented Programming- Encapsulation, Inheritance, Polymorphism, classes, objects, constructors, methods, parameter passing, static fields and methods, access control, this reference, overloading methods and constructors, recursion, garbage collection, String handling-building strings, operating on strings, StringBuffer and StringBuilder, Enumerations, autoboxing and unboxing, Generics.

**UNIT II**

**Inheritance** – Inheritance concept, benefits of inheritance , Super classes and Sub classes, Member access rules, Inheritance hierarchies, super uses, preventing inheritance: final classes and methods, casting, polymorphism- dynamic binding, method overriding, abstract classes and methods, the Object class and its methods. **Interfaces** – Interfaces vs. Abstract classes, defining an interface, implementing interfaces, accessing implementations through interface references, extending interface. **Packages**-defining, creating and accessing a Package, understanding CLASSPATH, importing packages.

**UNIT III**

**Exception handling** – Dealing with errors, benefits of exception handling, the classification of exceptions- exception hierarchy, checked exceptions and unchecked exceptions, usage of try, catch, throw, throws and finally, exception specification, built in exceptions, creating own exception sub classes. **Files** – streams- byte streams, character streams, text Input/output, binary input/output, random access file operations, file management using File class , java.io.

**UNIT IV**

**Operating system Introduction-** Operating system objectives and functions, Computer system organization, Computer system Architecture, Operating system structure, Operating system operations (Refer to Chapter 1 of 2<sup>nd</sup> text book).

Process Management, Memory Management, Storage Management, Protection and Security, Computing Environments (Refer to Chapter 1 of 2<sup>nd</sup> text book).

**Operating system structures-** Operating system services, User and Operating system interface, System calls, types of System calls, System programs (Refer to Chapter 2 of 2<sup>nd</sup> text book).

**UNIT V**

**MySQL(Database)-** Introduction to SQL, Data types, Creating Database objects , Querying and Updating Data- Data Definition language (DDL) statements-CREATE,ALTER,DROP and Data Manipulation Language statements(DML)-SELECT,INSERT, UPDATE,DELETE((Refer to Chapter 4 and Chapter 5 of 3<sup>rd</sup> text book).

**TEXT BOOKS :**

1. Java: the complete reference, 10<sup>th</sup> edition, Herbert Schildt, Oracle Press, Mc-Graw Hill Education, Indian Edition.
2. Operating System Concepts, 9<sup>th</sup> edition, Abraham Silberschatz, P.B. Galvin, G. Gagne, Wiley Student Edition (Refer to Chapter 1 and Chapter 2 only), 2016 India edition.
3. Database Systems, 6<sup>th</sup> edition, R. Elamasri and S. B. Navathe, Pearson(Refer to Chapter 4 and Chapter 5 only)

**REFERENCE BOOKS :**

1. Java: How to Program P.Deitel and H.Deitel ,10<sup>th</sup> edition, Pearson.
2. Java Programming, D.S.Malik, Cengage Learning.
3. Core Java, Volume 1-Fundamentals, 9<sup>th</sup> edition, Cay S.Horstmann and Gary Cornell, Pearson.
4. Programming in Java, S.Malhotra and S.Choudhary, Oxford Univ. Press.
5. Data base system concepts, A.Silberschatz, H.F. Korth and S.Sudarshan, Mc Graw Hill, VI th edition.
6. Introduction to SQL, Rick F. Van der Lans,4<sup>th</sup> edition, Pearson.
7. An introduction to programming and OO design using Java, J.Nino, F.A.Hosch, John Wiley&Sons.
8. Operating Systems – Internals and Design Principles, W. Stallings, Pearson.
9. Modern Operating Systems, Andrew S Tanenbaum, 3rd Edition, Pearson.
10. Operating Systems A concept-based Approach, 3<sup>rd</sup> Edition, D.M. Dhamdhare, TMH.

**M.TECH (DATA SCIENCES)  
I YEAR I SEMESTER**

**DS1A12**

**PROFESSIONAL COMMUNICATION SKILLS  
(AUDIT – 1)**

**Course Objectives:**

- To teach 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.

**Course Outcomes:**

- Students are trained to convert the conceptual understanding of communication into every day practice.
- Students are expected to be ready for placements.
- Students are prepared to communicate their ideas relevantly and coherently in professional writing.

**UNIT I**

**INTRODUCTION**

Basics of Communication - Principles of Communication - Types of Communication – Stages of Communication – Verbal and Non-verbal Communication – Channels of Communication – Barriers to Effective Communication – Formal and Informal Expressions in Various Situations.

**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**

Difference between Spoken and Written Communication- Features of Effective Writing - Formation of a Sentence – SVOs and SVOC patterns – Types of sentences- Common errors in Writing - Writing coherent sentences using connectives and conjunctions- Written Presentation Skills – Tenses – Concord – Question Tags - Practice Exercises - One Word Substitutes – Words Often Confused and Misspelt.

**UNIT IV**

**PROFESSIONAL WRITING**

Letter writing – Types, Parts and Styles of Formal Letters – Language to be used in Formal Letters – Letters of Enquiry, Complaint, and Apology with Replies – Letter of Application -Resume – E-mail – Active and Passive Voice.

**UNIT V****REPORT WRITING**

Types of Reports – Formats of Reports – Memo Format – Letter Format and Manuscript Format-  
Parts of Technical Report – Informational, Analytical and Project Reports – Idioms and Phrases.

**REFERENCE BOOKS:**

1. Meenakshi Raman & Sangeetha Sharma. 2012. *Technical Communication*. New Delhi
2. Rizvi, M. A. 2005. *Effective Technical Communication*. New Delhi: Tata McGraw Hill
3. Sanjay Kumar & Pushp Latha. 2012. *Communication Skills*. New Delhi: OUP
4. Er. A. K. Jain, Dr. Pravin S. R. Bhatia & Dr. A. M. Sheikh. 2013. *Professional Communication Skills*. S. Chand Publishers. New Delhi.
5. Farhathullah, T.M. 2009. *English for Business Communication*. Bangalore: Prism
6. Bikram K Das. 2011. *Functional Grammar and Spoken and Written Communication in English*. Kolkata: Orient Blackswan
7. Kiranmai Dutt, P *et al.* 2011. *A Course in Communication Skills*. New Delhi: CUP India
8. Krishnaswamy, N. 2000. *Modern English – A Book of Grammar, Vocabulary and Usage*. Macmillan India Pvt. Ltd
9. Ramachandran, K K. *et al.* 2007. *Business Communication*. New Delhi: Macmillan
10. Taylor, Ken. 2011. *50 ways to improve your Business English*. Hyderabad: Orient Blackswan

**M.TECH (DATA SCIENCES)  
I YEAR I SEMESTER**

**DS1A13**

**PERSONALITY DEVELOPMENT THROUGH LIFE ENLIGHTENMENT SKILLS  
(AUDIT - 1)**

**Course Objectives**

1. To learn to achieve the highest goal happily
2. To become a person with stable mind, pleasing personality and determination
3. To awaken wisdom in students

**UNIT I**

Neetisatakam-Holistic development of personality

- Verses- 19,20,21,22 (wisdom), Verses- 29,31,32 (pride & heroism), Verses- 26,28,63,65 (virtue), Verses- 52,53,59 (dont's), Verses- 71,73,75,78 (do's)

**UNIT II**

- Approach to day to day work and duties
- Shrimad BhagwadGeeta : Chapter 2-Verses 41, 47,48,
- Chapter 3-Verses 13, 21, 27, 35, Chapter 6-Verses 5,13,17, 23, 35,
- Chapter 18-Verses 45, 46, 48.

**UNIT III**

- Statements of basic knowledge.
- Shrimad BhagwadGeeta: Chapter2-Verses 56, 62, 68
- Chapter 12 -Verses 13, 14, 15, 16,17, 18
- Personality of Role model. Shrimad BhagwadGeeta:Chapter2-Verses 17, Chapter 3-Verses 36,37,42,
- Chapter 4-Verses 18, 38,39
- Chapter18 – Verses 37,38,63

**SUGGESTED READING**

1. “Srimad Bhagavad Gita” by Swami SwarupanandaAdvaita Ashram (Publication Department), Kolkata
2. Bhartrihari's Three Satakam (Niti-sringar-vairagya) by P.Gopinath, 4. Rashtriya Sanskrit Sansthanam, New Delhi.

**M.TECH (DATA SCIENCES)  
I YEAR I SEMESTER**

**DS1A14**

**VALUE EDUCATION  
(AUDIT – 1)**

**Course Objectives**

Students will be able to

1. Understand value of education and self- development
2. Imbibe good values in students
3. Let the should know about the importance of character

**UNIT I**

Values and self-development –Social values and individual attitudes. Work ethics, Indian vision of humanism. Moral and non- moral valuation. Standards and principles. Value judgements

**UNIT II**

Importance of cultivation of values. Sense of duty. Devotion, Self-reliance. Confidence, Concentration. Truthfulness, Cleanliness. Honesty, Humanity. Power of faith, National Unity. Patriotism.Love for nature,Discipline

**UNIT III**

Personality and Behavior Development - Soul and Scientific attitude. Positive Thinking. Integrity and discipline. Punctuality, Love and Kindness. Avoid fault Thinking. Free from anger, Dignity of labour. Universal brotherhood and religious tolerance.

**UNIT IV**

True friendship. Happiness Vs suffering, love for truth. Aware of self-destructive habits. Association and Cooperation. Doing best for saving nature

**UNIT V**

Character and Competence –Holy books vs Blind faith. Self-management and Good health. Science of reincarnation. Equality, Nonviolence,Humility, Role of Women. All religions and same message. Mind your Mind, Self-control.Honesty, Studying effectively

**TEXTBOOK:**

1. Chakroborty, S.K. “Values and Ethics for organizations Theory and practice”, Oxford University Press, New Delhi



**M.TECH (DATA SCIENCES)  
I YEAR I SEMESTER**

**DS1A15**

**CONSTITUTION OF INDIA  
(AUDIT-1)**

**Course Objectives:** Students will be able to:

1. Understand the premises informing the twin themes of liberty and freedom from a civil rights perspective.
2. To address the growth of Indian opinion regarding modern Indian intellectuals' constitutional role and entitlement to civil and economic rights as well as the emergence of nationhood in the early years of Indian nationalism.
3. To address the role of socialism in India after the commencement of the Bolshevik Revolution in 1917 and its impact on the initial drafting of the Indian Constitution

**UNIT I**

History of Making of the Indian Constitution: History Drafting Committee, ( Composition & Working) Philosophy of the Indian Constitution:Preamble Salient Features

**UNIT II**

Contours of Constitutional Rights & Duties: Fundamental Rights Right to Equality Right to Freedom Right against Exploitation Right to Freedom of Religion Cultural and Educational Rights Right to Constitutional Remedies Directive Principles of State Policy Fundamental Duties.

**UNIT III**

Organs of Governance: Parliament Composition Qualifications and Disqualifications Powers and Functions Executive President Governor Council of Ministers Judiciary, Appointment and Transfer of Judges, Qualifications Powers and Functions

**UNIT IV**

Local Administration: District's Administration head: Role and Importance, Municipalities: Introduction, Mayor and role of Elected Representative, CEO of Municipal Corporation. Pachayati raj: Introduction, PRI: ZilaPachayat. Elected officials and their roles, CEO ZilaPachayat: Position and role. Block level: Organizational Hierarchy (Different departments), Village level: Role of Elected and Appointed officials, Importance of grass root democracy

**UNIT V**

Election Commission: Election Commission: Role and Functioning. Chief Election Commissioner and Election Commissioners. State Election Commission: Role and Functioning. Institute and Bodies for the welfare of SC/ST/OBC and women.

**SUGGESTED READING:**

1. The Constitution of India, 1950 (Bare Act), Government Publication.
2. Dr. S. N. Busi, Dr. B. R. Ambedkar framing of Indian Constitution, 1st Edition, 2015.
3. M. P. Jain, Indian Constitution Law, 7th Edn., Lexis Nexis, 2014.
4. D.D. Basu, Introduction to the Constitution of India, Lexis Nexis, 2015

**MASTER OF TECHNOLOGY  
(DATA SCIENCES)  
I YEAR II SEMESTER**

DS2C10

**MACHINE LEARNING  
(CORE-4)**

**Objectives:**

- To be able to formulate machine learning problems corresponding to different applications.
- To understand a range of machine learning algorithms along with their strengths and weaknesses.
- To understand the basic theory underlying machine learning.

**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, Perceptrons, Multilayer Networks and the Back propagation Algorithm.

**Evaluation Hypotheses:** Motivation, Estimation Hypothesis Accuracy, Basics of Sampling Theory, A General Approach for Deriving Confidence Intervals, Difference in Error of Two Hypotheses, Comparing Learning Algorithms.

**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.

**UNIT III**

**Support Vector Machines:** Introduction, The Support Vector Classifier, Support Vector Machines and Kernels, Markov models, Hidden Markov Models.

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

**Clustering and Unsupervised Learning:** K-means clustering, Gaussian mixture density estimation, model selection

**UNIT IV**

**Genetic Algorithms:** Motivation, Genetic Algorithms, An Illustrative Example, Hypothesis Space Search, Genetic Programming, Models of Evolution and Learning, Parallelizing Genetic Algorithms.

**Learning Sets of Rules:** Introduction, Sequential Covering Algorithms, Learning Rule Sets: Summary, Learning First-Order Rules, Learning Sets of First-Order Rules:FOIL, Induction as Inverted Deduction, Inverting Resolution.

**Reinforcement Learning:** Introduction, The Learning Task,  $Q$ -learning, Non-Deterministic, Rewards And Actions, Temporal Difference Learning, Generalizing from examples, Relationship to Dynamic Programming.

**UNIT V**

**Dimensionality reduction:** Feature Selection, Principal Component Analysis, Linear Discriminant Analysis, Factor Analysis, Independent Component Analysis, Multidimensional Scaling, Manifold Learning.

**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, McGraw Hill, 1997
2. The Elements of Statically Learning - Trevor Has Tie, Robert Tibshirani & Jerome Friedman., Springer Verlag, 2001

**REFERENCE BOOKS:**

1. Machine Learning Methods in the Environmental Sciences, Neural Networks, William W Hsieh, Cambridge Univ Press.
2. Pattern Classification - Richard o. Duda, Peter E. Hart and David G. Stork, John Wiley & Sons Inc.,2001
3. Neural Networks for Pattern Recognition - Chris Bishop, Oxford University Press, 1995

**MASTER OF TECHNOLOGY  
(DATA SCIENCES)  
I YEAR II SEMESTER**

DS2C20

**BIG DATA  
(CORE-5)**

**Objectives:**

To understand about big data, to learn the analytics of Big Data

To understand how data is stored and processed in Hadoop

To learn about NoSQL databases

To learn R tool and understand how data is analyzed using R features

To learn about spark and to understand what features of it are making it to overtake hadoop

**UNIT I**

**Types of Digital data:** Classification of Digital Data, **Introduction to Big Data:** What is big data, Evolution of Big Data, Traditional Business Intelligence vs Big Data, Coexistence of Big Data and Data Warehouse. **Big Data Analytics:** What is Big Data Analytics, What Big Data Analytics Isn't, Why this sudden Hype Around Big Data Analytics, Classification of Analytics, Greatest Challenges that Prevent Business from Capitalizing Big Data, Top Challenges Facing Big Data, Why Big Data Analytics Important, Data Science, Terminologies used in Big Data Environments.

**UNIT II**

**Hadoop:** Features of Hadoop, Key advantages of hadoop, versions of hadoop, overview of hadoop ecosystem, Hadoop distributions. Why hadoop? RDBMS vs Hadoop, Distribution computing challenges, History of hadoop, Hadoop overview, HDFS

**UNIT III**

Processing data with hadoop, interfacing with hadoop ecosystem.

**NoSQL:** Where it is used? What is it? Types of NoSQL Databases, Why NoSQL? Advantages of NoSQL, What we miss with NoSQL? Use of NoSQL in industry, SQL vs NoSQL.

**UNIT IV**

What is R? Why use R for analytics? How to run R? First R example, functions a short programming example, some important R data structures, vectors, matrices, lists, R programming structures.

**UNIT V**

Introduction to Spark, Scala language: values, data types, variables, expressions, conditional expressions, evaluation order, compound expressions, functions, tuple with functions, List, Length, ++, ::, sorted, reverse, sum. slice, mkString, contains, map, filter, leftfold, reduce, Map, Contains, getOrElse, WithDefault, Keys and Values, groupBy, set, mapValues, keys and values, Option(Some and None), Objects, classes, inheritance, traits

**TEXT BOOKS:**

1. BIG DATA and ANALYTICS, Seema Acharya, Subhashini Chellappan, Wiley publications.(Unit I, II, III)
2. BIG DATA, Black Book™, DreamTech Press, 2015 Edition
3. “The art of R programming” by Norman matloff, 2009.(Unit IV)
4. “Atomic Scala”, 2<sup>nd</sup> edition, Bruce Eckel, Dianne Marsh. (Unit V)

**MASTER OF TECHNOLOGY  
(DATA SCIENCES)  
I YEAR II SEMESTER**

**DS2C30                      NATURAL LANGUAGE PROCESSING  
(CORE-6)**

**Objectives:**

- To acquire basic understanding of linguistic concepts and natural language complexity variability.
- To acquire basic understanding of machine learning techniques as applied to language.
- To implement N-grams Models.

**UNIT I**

**Introduction and Overview:** What is Natural Language Processing, hands-on demonstrations. Ambiguity and uncertainty in language. The Turing test. **Regular Expressions** Chomsky hierarchy, regular languages, and their limitations. Finite-state automata. Practical regular expressions for finding and counting language phenomena. A little morphology. Exploring a large corpus with regex tools. **Programming in Python** An introduction to programming in Python. Variables, numbers, strings, arrays, dictionaries, conditionals, iteration. The NLTK (Natural Language Toolkit) **String Edit Distance and Alignment** Key algorithmic tool: dynamic programming, a simple example, use in optimal alignment of sequences. String edit operations, edit distance, and examples of use in spelling correction.

**UNIT II**

**Context Free Grammars** Constituency, CFG definition, use and limitations. Chomsky Normal Form. Top-down parsing, bottom-up parsing, and the problems with each. The desirability of combining evidence from both directions **Non-probabilistic Parsing** Efficient CFG parsing with CYK, another dynamic programming algorithms. Early parser. Designing a little grammar, and parsing with it on some test data. **Probability** Introduction to probability theory Joint and conditional probability, independence, Bayes rule. Examples of applications in natural language. **Information Theory** The "Shannon game"--motivated by language! Entropy, crossentropy, information gain. Its application to some language phenomena.

**UNIT III**

**Language modeling and Naive Bayes** Probabilistic language modeling and its applications. Markov models. N-grams. Estimating the probability of a word, and smoothing. Generative models of language. Part of Speech Tagging and Hidden Markov Models, Viterbi Algorithm for Finding Most Likely HMM Path Dynamic programming with Hidden Markov Models, and its use for part-of-speech tagging.

**UNIT IV**

**Probabilistic Context Free Grammars** Weighted context free grammars. Weighted CYK. Pruning and beam search. **Parsing with PCFGs** A tree bank and what it takes to create one. The probabilistic version of CYK. **Maximum Entropy Classifiers** The maximum entropy principle and its relation to maximum likelihood. Maximum entropy classifiers and their application to document classification, sentence segmentation, and other language tasks

**UNIT V**

**Maximum Entropy Markov Models & Conditional Random Fields** Part-of-speech tagging, noun-phrase segmentation and information extraction models that combine maximum entropy and finite-state machines. **Lexical Semantics** Mathematics of Multinomial and Dirichlet distributions. Information Extraction & Reference Resolution- Various methods, including HMMs. Models of anaphora resolution. Machine learning methods for co reference.

**TEXT BOOKS:**

1. "Speech and Language Processing": Jurafsky and Martin, Prentice Hall
2. "Statistical Natural Language Processing" - Manning and Schutze, MIT Press
3. "Natural Language Understanding". James Allen. The Benajmins/Cummings Publishing Company

**REFERENCES BOOKS:**

1. Cover, T. M. and J. A. Thomas: Elements of Information Theory. Wiley.
2. Charniak, E.: Statistical Language Learning. The MIT Press.
3. Jelinek, F.: Statistical Methods for Speech Recognition. The MIT Press.
4. Lutz and Ascher - "Learning Python", O'Reilly





**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
2. Android Application Development for Java Programmers, James C Sheusi, Cengage Learning, 2013

**REFERENCES:**

1. Beginning Android 4 Application Development, Wei-Meng Lee, Wiley India (Wrox), 2013

**M.TECH (DATA SCIENCES)  
I YEAR II SEMESTER**

**DS2E12**

**COMPUTER FORENSICS  
(ELECTIVE – 3)**

**Objectives:**

- To understand the cyberspace
- To understand the forensics fundamentals
- To understand the evidence capturing process.
- To understand the preservation of digital evidence.

**UNIT I**

**Computer Forensics Fundamentals:** Introduction to Computer Forensics, Use of Computer Forensics in Law Enforcement, Computer Forensics Assistance to Human Resources/Employment Proceedings, Computer Forensics Services, Benefits of Professional Forensics Methodology, Steps Taken by Computer Forensics Specialists, Who Can Use Computer Forensic Evidence?. **Types of Computer Forensics Technology :** Types of Military Computer Forensic Technology, Types of Law Enforcement Computer Forensic Technology, Types of Business Computer Forensics Technology.

**UNIT II**

**Computer Forensics Evidence and Capture: Data Recovery:** Data Recovery Defined, Data Backup and Recovery, The Role of Backup in Data Recovery, The Data-Recovery Solution, Case Histories. **Evidence Collection and Data Seizure:** Why Collect Evidence?, Collection Options, Obstacles, Types of Evidence, The Rules of Evidence, Volatile Evidence, General Procedure, Collecting and Archiving, Methods of Collection, Artifacts, Collection Steps, Controlling Contamination: The Chain of Custody.

**UNIT III**

**Duplication and Preservation of Digital Evidence:**

Preserving the Digital Crime Scene, Computer Evidence Processing Steps, Legal Aspects of Collecting And Preserving Computer Forensic Evidence. **Computer Image Verification and Authentication :** Special Needs of Evidential Authentication, Practical Considerations, Practical Implementation.

**UNIT IV**

**Computer Forensics Analysis: Discovery of Electronic Evidence:**

Electronic Document Discovery: A Powerful New Litigation Tool, **Identification of Data:** Timekeeping, Time Matters, Forensic Identification and Analysis of Technical Surveillance Devices. **Reconstructing Past Events:** How to Become a Digital Detective, Useable File Formats, Unusable File Formats, Converting Files. **Networks:** Network Forensics Scenario, A Technical Approach, Destruction of Email, Damaging Computer Evidence, International Principles Against Damaging of Computer Evidence, Tools Needed for Intrusion Response to the Destruction of Data, Incident Reporting and Contact Forms

**UNIT V**

**Current Computer Forensics Tools:** Evaluating Computer Forensics Tool Needs, Computer Forensics Software Tools, Computer Forensics Hardware Tools, Validating and Testing Forensics Software.

**TEXT BOOKS:**

- 
1. “Computer Forensics : Computer Crime Scene Investigation”, JOHN R. VACCA, Firewall Media.
  2. “Guide to Computer Forensics and Investigations”4e, Nelson, Phillips Enfinger, Stuart, Cengage Learning.
- 

**REFERENCES:**

- 
1. “Computer Forensics and Cyber Crime”, Marjie T Britz, Pearson Education.
  2. “Computer Forensics”, David Cowen, Mc Graw Hill.
- 
3. Brian Carrier , "File System Forensic Analysis" , Addison Wesley, 2005
  4. Dan Farmer & Wietse Venema , "Forensic Discovery", Addison Wesley, 2005
  5. Eoghan Casey , —Digital Evidence and Computer Crime —, Edition 3, Academic Press, 2011
  6. Chris Pogue, Cory Altheide, Todd Haverkos , Unix and Linux Forensic Analysis DVD ToolKit, Syngress Inc. , 2008
  7. Harlan Carvey , Windows Forensic Analysis DVD Toolkit, Edition 2, Syngress Inc. , 2009
  8. Harlan Carvey , Windows Registry Forensics: Advanced Digital Forensic Analysis of the Windows Registry , Syngress Inc, Feb 2011
  9. Eoghan Casey, Handbook of Digital Forensics and Investigation, Academic Press, 2009
  10. Gonzales/ Woods/ Eddins, Digital Image Processing using MATLAB, 2nd edition, Gatesmark Publishing, ISBN 9780982085400
  11. N.Efford, Digital Image Processing, Addison Wesley 2000, ISBN 0-201-59623-7
  12. M Sonka, V Hlavac and R Boyle, Image Processing, Analysis and Machine Vision, PWS 1999, ISBN 0-534-95393-
  13. Pratt.W.K., Digital Image Processing, John Wiley and Sons, New York, 1978

**M.TECH (DATA SCIENCES)  
I YEAR II SEMESTER**

**DS2E13**

**NETWORK PROGRAMMING  
(ELECTIVE – 3)**

**Objectives:**

- To understand Linux utilities
- To understand file handling, signals
- To understand IPC, network programming in Java
- To understand processes to communicate with each other across a Computer Network.

**UNIT I**

Linux Utilities- File handling utilities, Security by file permissions, Process utilities, Disk utilities, Networking utilities, Filters, Text processing utilities and Backup utilities. Bourne again shell(bash) - Introduction, pipes and redirection, here documents, running a shell script, the shell as a programming language, shell meta characters, file name substitution, shell variables, command substitution, shell commands, the environment, quoting, test command, control structures, arithmetic in shell, shell script examples. Review of C programming concepts-arrays, strings (library functions), pointers, function pointers, structures, unions, libraries in C.

**UNIT II**

Files- File Concept, File types File System Structure, Inodes, File Attributes, file I/O in C using system calls, kernel support for files, file status information-stat family, file and record locking-lockf and fcntl functions, file permissions- chmod, fchmod, file ownership-chown, lchown, fchown, links-soft links and hard links – symlink, link, unlink. File and Directory management – Directory contents, Scanning Directories- Directory file APIs. Process- Process concept, Kernel support for process, process attributes, process control – process creation, replacing a process image, waiting for a process, process termination, zombie process, orphan process.

**UNIT III**

Signals- Introduction to signals, Signal generation and handling, Kernel support for signals, Signal function, unreliable signals, reliable signals, kill, raise, alarm, pause, abort, sleep functions. Interprocess Communication - Introduction to IPC mechanisms, Pipes- creation, IPC between related processes using unnamed pipes, FIFOs-creation, IPC between unrelated processes using FIFOs(Named pipes), differences between unnamed and named pipes, popen and pclose library functions, Introduction to message queues, semaphores and shared memory. Message Queues- Kernel support for messages, UNIX system V APIs for messages, client/server example. Semaphores- Kernel support for semaphores, UNIX system V APIs for semaphores.

**UNIT IV**

Shared Memory- Kernel support for shared memory, UNIX system V APIs for shared memory, client/server example. Network IPC - Introduction to Unix Sockets, IPC over a network, Client-Server model, Address formats (Unix domain and Internet domain), Socket system calls for Connection Oriented - Communication, Socket system calls for Connectionless-Communication, Example-Client/Server Programs- Single Server-Client connection, Multiple simultaneous clients, Socket options – setsockopt, getsockopt,fcntl.

**UNIT V**

Network Programming in Java-Network basics, TCP sockets, UDP sockets (datagram sockets), Server programs that can handle one connection at a time and multiple connections (using multithreaded server), Remote Method Invocation (Java RMI)-Basic RMI Process, Implementation details-Client-Server Application.

**TEXT BOOKS:**

1. Unix System Programming using C++, T.Chan, PHI.(Units II,III,IV)
2. Unix Concepts and Applications, 4th Edition, Sumitabha Das, TMH.(Unit I)
3. An Introduction to Network Programming with Java, Jan Graba, Springer, 2010.(Unit V)
4. Unix Network Programming, W.R. Stevens, PHI.(Units II,III,IV)
5. Java Network Programming, 3<sup>rd</sup> edition, E.R. Harold, SPD, O'Reilly.(Unit V)

**REFERENCES:**

1. Linux System Programming, Robert Love, O'Reilly, SPD.
2. Advanced Programming in the UNIX environment, 2<sup>nd</sup> Edition, W.R.Stevens, Pearson Education.
3. UNIX for programmers and users, 3<sup>rd</sup> Edition, Graham Glass, King Ables, Pearson Education.
4. Beginning Linux Programming, 4<sup>th</sup> Edition, N.Matthew, R.Stones, Wrox, Wiley India Edition.
5. Unix Network Programming The Sockets Networking API, Vol.-I, W.R.Stevens, Bill Fenner, A.M.Rudoff, Pearson Education.
6. Unix Internals, U.Vahalia, Pearson Education.
7. Unix shell Programming, S.G.Kochan and P.Wood, 3<sup>rd</sup> edition, Pearson Education.
8. C Programming Language, Kernighan and Ritchie, PHI

**M.TECH (DATA SCIENCES)  
I YEAR II SEMESTER**

**DS2E14**

**ADVANCED PROGRAMMING  
(ELECTIVE – 3)**

**UNIT I**

Basic features of Python-Interactive execution, comments, types, variables, operators, expressions, Statements-assignment, input, print, Control flow-Conditionals, Loops, break statement, continue statement, pass statement, Functions, definition, call, scope and lifetime of variables, keyword arguments, default parameter values, variable length arguments, recursive functions, Functional programming-mapping, filtering and reduction, Lambda functions, Scope, namespaces and modules-import statement, creating own modules, avoiding namespace collisions when importing modules, module reload, LEGB rule, dir( ) function, iterators and generators, Sequences-Strings ,Lists and Tuples-basic operations and functions, iterating over sequences, List comprehensions, Packing and Unpacking of Sequences, Sets and Dictionaries- operations, regular expressions, Python program examples.

**UNIT II**

Files-operations-opening, reading, writing, closing, file positions, file names and paths, functions for accessing and manipulating files and directories on disk, os module, Exceptions – raising and handling exceptions, try/except statements, finally clause, standard exceptions, Object oriented programming- classes, constructors, objects, class variables, class methods, static methods, Inheritance-is-a relationship, composition, polymorphism, overriding, multiple inheritance, abstract classes, multithreaded programming, time and calendar modules, Python program examples.

**UNIT III**

GUI Programming with Tkinter , Widgets(Buttons, Canvas, Frame, Label, Menu, Entry, Text, Scrollbar, Combobox, Listbox, Scale),event driven programming-events, callbacks, binding, layout management-geometry managers:pack and grid, creating GUI based applications in Python.

**UNIT IV**

Network Programming-Sockets, Socket addresses, Connection-oriented and Connectionless Sockets,socket module,urllib module,Socket object methods,Client/Server applications(TCP/IP and UDP/IP),Socketserver module, handling multiple clients, Client side scripting-Transferring files-FTP, ftplib module,ftplib.FTP class methods, sending and receiving emails- smtplib module, smtplib.SMTP class methods, poplib module, poplib.POP3 methods, Python program examples.

**UNIT V**

Database Programming-SQL Databases,SQLite,sqlite3 module, connect function( ),DB-API 2.0 Connection object methods, Cursor object Attributes and methods, creating Database applications in Python, Web programming-Simple web client, urllib, urlparse modules, Server side scripting-Building CGI applications-Setting up a web server, Creating the form page, Generating the results page, Saving state information in CGI Scripts, HTTP Cookies, Creating a cookie, Using cookies in CGI scripts, Handling cookies with urllib2 module, cgi module.

**TEXT BOOKS :**

1. Exploring Python, Timothy A. Budd, McGraw Hill Publications.
2. Core Python Programming, 2<sup>nd</sup> edition, W.J.Chun, Pearson.
3. Python Programming, R.Thareja, Oxford University Press.
4. Programming Python, 3<sup>rd</sup> edition, Mark Lutz, SPD,O'Reilly.

**REFERENCE BOOKS :**

1. Introduction to Computer Science using Python, Charles Dierbach, Wiley India Edition.
2. Fundamentals of Python, K. A. Lambert, B.L. Juneja, Cengage Learning.
3. Beginning Python, 2<sup>nd</sup> edition, Magnus Lie Hetland, Apress, dreamtech press.
4. Starting out with Python, 3<sup>rd</sup> edition, Tony Gaddis, Pearson.
5. Python Essential Reference, D.M.Beazley, 3<sup>rd</sup> edition, Pearson.
6. Programming in Python3, Mark Summerfield, Pearson.
7. Think Python, How to think like a computer scientist, Allen B. Downey, SPD, O'Reilly.
8. [www.python.org](http://www.python.org) web site.

**M.TECH (DATA SCIENCES)  
I YEAR II SEMESTER**

**DS2E15**

**CRYPTOGRAPHY AND NETWORK SECURITY  
(ELECTIVE – 3)**

**Objectives:**

- Understand the basic categories of threats to computers and networks
- Understand various cryptographic algorithms.
- Describe public-key cryptosystem.
- Describe the enhancements made to IPv4 by IPSec
- Understand Intrusions and intrusion detection
- Generate and distribute a PGP key pair and use the PGP package to send an encrypted e-mail message.
- Discuss Web security and Firewalls

**UNIT I**

**Attacks on Computers and Computer Security:** Introduction, The need for security, Security approaches, Principles of security, Types of Security attacks, Security services, Security Mechanisms, A model for Network Security **Cryptography: Concepts and Techniques:** Introduction, plain text and cipher text, substitution techniques, transposition techniques, encryption and decryption, symmetric and asymmetric key cryptography, steganography, key range and key size, possible types of attacks.

**UNIT II**

**Symmetric key Ciphers:** Block Cipher principles & Algorithms(DES, AES,Blowfish), Differential and Linear Cryptanalysis, Block cipher modes of operation, Stream ciphers, RC4,Location and placement of encryption function, Key distribution **Asymmetric key Ciphers:** Principles of public key cryptosystems, Algorithms(RSA, Diffie-Hellman,ECC), Key Distribution

**UNIT III**

**Message Authentication Algorithms and Hash Functions:** Authentication requirements, Functions, Message authentication codes, Hash Functions, Secure hash algorithm, Whirlpool, HMAC, CMAC, Digital signatures, knapsack algorithm.

**UNIT IV**

**E-Mail Security:** Pretty Good Privacy, S/MIME **IP Security:** IP Security overview, IP Security architecture, Authentication Header, Encapsulating security payload, combining security associations, key management



**UNIT V**

**Web Security:** Web security considerations, Secure Socket Layer and Transport Layer Security, Secure electronic transaction **Intruders, Virus and Firewalls:** Intruders, Intrusion detection, password management, Virus and related threats, Countermeasures, Firewall design principles, Types of firewalls **Case Studies on Cryptography and security:** Secure Inter-branch Payment Transactions, Cross site Scripting Vulnerability, Virtual Elections.

**TEXT BOOKS:**

1. Cryptography and Network Security : William Stallings, Pearson Education, 5<sup>th</sup> Edition
2. Cryptography and Network Security: Atul Kahate, Mc Graw Hill, 2<sup>nd</sup> Edition.
3. Network Security and Cryptography: Bernard Menezes, CENGAGE Learning

**REFERENCES:**

1. Cryptography and Network Security: C K Shyamala, N Harini, Dr T R Padmanabhan, Wiley India, 1<sup>st</sup> Edition.
2. Cryptography and Network Security : Forouzan Mukhopadhyay, Mc Graw Hill, 2<sup>nd</sup> Edition
3. Information Security, Principles and Practice : Mark Stamp, Wiley India.
4. Principles of Computer Security: WM.Arthur Conklin, Greg White, TMH
5. Introduction to Network Security: Neal Krawetz, CENGAGE Learning.
6. Principles of Information security by Michael E Whitman and Herbert J.Mattord.

**M.TECH (DATA SCIENCES)  
I YEAR II SEMESTER**

**DS2E21**

**CLOUD COMPUTING  
(ELECTIVE – 4)**

**Objectives:**

- Cloud computing has evolved as a very important computing model, which enables information, software, and shared resources to be provisioned over the network as services in an on-demand manner.
- This course provides an insight into what is cloud computing and the various services cloud is capable.

**UNIT I**

**Introduction:** Computing, Distributed Computing, Cluster Computing, Grid Computing,  
**Fundamentals:** Motivation for Cloud Computing, Defining Cloud Computing, Definition of Cloud computing, cloud architecture layers, Principles of Cloud computing, cloud ecosystem, requirements for cloud services, cloud architecture ,cloud application, benefits and drawbacks.

**UNIT II**

**Technological drivers for cloud computing:** Introduction, SOA and Cloud, Virtualization multicore technology memory and storage technologies, Networking technologies, Web 2.0,web3.0 Agile SDLC for cloud computing, how cloud meets agile process?, Application Environments.

**UNIT III**

**Cloud Deployment Models:** Introduction private cloud, public cloud, community cloud, hybrid cloud.

**Cloud Service Models:** Infrastructure as a Service, Platform as a Service, Software as a Service, Other Cloud Service Models.

**UNIT IV**

**Programming models for cloud computing:** Introduction, extended programming models for cloud , new programming models proposed for cloud; **software development in cloud:** Introduction Different perspectives on SaaS Development, new challenges, cloud-aware software development using Paas technology,

**UNIT V**

**Cloud Service Providers:**EMC, ,Google, ,Amazon Web Services, Microsoft, IBM, ,SAP Labs, ,Sales force, Rackspace,VMware,Manjrasoft, **Service level agreements(SLAs) in cloud,**

**TEXT BOOKS:**

1. Essentials of cloud Computing : K.Chandrasekhran , CRC press, 2014

**REFERENCES:**

1. Cloud Computing: Principles and Paradigms by Rajkumar Buyya, James Broberg and Andrzej M. Goscinski, Wiley, 2011.
2. Distributed and Cloud Computing, Kai Hwang, Geoffery C.Fox, Jack J.Dongarra, Elsevier, 2012.
3. Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance, Tim Mather, Subra Kumaraswamy, Shahed Latif, O'Reilly, SPD, rp2011.

**M.TECH (DATA SCIENCES)  
I YEAR II SEMESTER**

**DS2E22**

**MULTIMODAL ANALYTICS  
(ELECTIVE – 4)**

**Objectives:**

- To learn NLP techniques and their applications
- To explore clustering and classification approaches on Text and Image data
- To design face detection and recognition algorithms using image analytics
- To apply text and image analytics across various industry application

**UNIT I**

**Introduction to Natural language processing:** Text Tokenization, Text Normalization, Understanding Text syntax and Structure, part-of-speech tagging, chunking, syntax parsing and named entity recognition. Text and Document Representation – format and structure, TF-IDF, Bag of words, n-grams, Word2Vec. Working with NLP toolkits.

**UNIT II**

**Text Classification and Clustering:** Basic supervised text categorization algorithms - Naive Bayes, k Nearest Neighbor (kNN) and Logistic Regression. Support Vector Machines and Decision Trees. Text Clustering - Partition and Hierarchical clustering algorithms for text clustering; Topic modelling Basics- topic models - Probabilistic Latent Semantic Indexing (pLSI) and Latent Dirichlet Allocation (LDA), and their variants for different application scenarios.

**Document Summarization:** Extraction-based Summarization techniques, Machine Learning based Summarization techniques; **Opinion Mining and Sentiment Analysis** - Sentiment polarity prediction, review mining, and aspect-based sentiment analysis.

**UNIT III**

**Image Processing Basics-** Visual Perception- Sampling and Quantization- Basic Relations between Pixels- Mathematical Tools Used in Digital Image Processing: Fundamental Operations – Vector and Matrix Operations- Image Transforms (DFT, DCT, DWT, Hadamard). Image features – colour, shape, texture, etc. Local Features, Edge Detection, Histograms and basic statistical models of image. Fundamentals of spatial filtering: spatial correlation and convolution-smoothing blurring-sharpening- edge detection - Basics of filtering in the frequency domain: smoothing-blurring-sharpening. Segmentation techniques.

**UNIT IV**

**Object Detection, Recognition and Tracking:** Eigen Faces, Sparse Representation Texture models Image and Video Classification models-Object tracking in Video. Tracking & Video Analysis: Object Tracking using Active Contours Tracking and Motion Understanding – Kalman filters, condensation, particle, Bayesian filters, hidden Markov models, change detection and model based tracking, Face detection and recognition, Convolutional features for visual recognition.

**UNIT V**

**Image classification and clustering Techniques-** Deep learning techniques, image categorisation, fine-grained recognition, Content-based Image Retrieval.

**Text and Image Analytics Applications** – Question and Answer Systems, Chatbots, Image captioning, Gesture Recognition, Video Surveillance, Traffic Monitoring, Intelligent Transport System, etc.

**TEXT BOOKS:**

1. Mining Text Data. Charu C. Aggarwal and ChengXiangZhai, Springer, 2012.
2. Sarkar Dipanjan, *Text Analytics with Python*, apress, 2016
3. Richard Szeliski, “Computer Vision: Algorithms and Applications”, Springer, 2011.

**REFERENCES:**

1. Rafael Gonzalez and Richard Woods, Digital Image Processing



**M.TECH (DATA SCIENCES)  
I YEAR II SEMESTER**

**DS2E24**

**WEB ANALYTICS AND DEVELOPMENT  
(ELECTIVE – 4)**

**Objective:**

The course explores use of social network analysis to understand growing connectivity and complexity in the world ranging from small groups to WWW.

**UNIT I**

Introduction – Social network and Web data and methods, Graph and Matrices, Basic measures for individuals and networks, Information Visualization

**UNIT II**

Web Analytics tools: Click Stream Analysis, A/B testing, Online Surveys

**UNIT III**

Web Search and Retrieval: Search Engine Optimization, Web Crawling and indexing, Ranking Algorithms, Web traffic models

**UNIT IV**

Making Connection: Link Analysis, Random Graphs and Network evolution, Social Connects: Affiliation and identity

**UNIT V**

Connection: Connection Search, Collapse, Robustness Social involvements and diffusion of innovation

**REFERENCES:**

1. Hansen, Derek, Ben Shneiderman, Marc Smith. 2011. Analyzing Social Media Networks with NodeXL: Insights from a Connected World. Morgan Kaufmann, 304.
2. Avinash Kaushik. 2009. Web Analytics 2.0: The Art of Online Accountability.
3. Easley, D. & Kleinberg, J. (2010). Networks, Crowds, and Markets: Reasoning About a Highly Connected World. New York: Cambridge University Press.  
<http://www.cs.cornell.edu/home/kleinber/networks-book/>
4. Wasserman, S. & Faust, K. (1994). Social network analysis: Methods and applications. New York: Cambridge University Press. Monge, P. R. & Contractor, N. S. (2003). Theories of communication networks. New York: Oxford University Press.

**M.TECH (DATA SCIENCES)  
I YEAR II SEMESTER**

**DS2E25**

**KNOWLEDGE DISCOVERY  
(ELECTIVE – 4)**

**COURSE OBJECTIVE**

- Conduct case studies on real data mining examples

**UNIT I**

**Introduction KDD and Data Mining** - Data Mining and Machine Learning, Machine Learning and Statistics, Generalization as Search, Data Mining and Ethics

**UNIT II**

**Knowledge Representation** - Decision Tables, Decision Trees, Classification Rules, Association Rules, Rules involving Relations, Trees for Numeric Predictions, Neural Networks, Clusters

**UNIT III**

**Decision Trees** - Divide and Conquer, Calculating Information, Entropy, Pruning, Estimating Error Rates, The C4.5 Algorithm

**Evaluation of Learned Results**- Training and Testing, Predicting Performance, Cross-Validation

**UNIT IV**

**Classification Rules** - Inferring Rudimentary Rules, Covering Algorithms for Rule Construction, Probability Measure for Rule Evaluation, Association Rules, Item Sets, Rule Efficiency

**Numeric Predictions** - Linear Models for Classification and Numeric Predictions, Numeric Predictions with Regression Trees, Evaluating Numeric Predictions

**UNIT V**

**Artificial Neural Networks** – Perceptrons, Multilayer Networks, The Backpropagation Algorithm

**Clustering** - Iterative Distance-based Clustering, Incremental Clustering, The EM Algorithm

**REFERENCES:**

1. Data mining and knowledge discovery handbook by Maimon, oded(et al.)
2. Data Cleansing : A Prelude to knowledge Discovery



**M.TECH (DATA SCIENCES)  
I YEAR II SEMESTER**

**DS2E26**

**PREDICTIVE ANALYTICS  
(ELECTIVE – 4)**

**Objectives:**

- To learn, how to develop models to predict categorical and continuous outcomes, using such techniques as neural networks, decision trees, logistic regression, support vector machines and Bayesian network models.
- To advice on when and how to use each model. Also learn how to combine two or more models to improve prediction

**UNIT I**

**Linear Methods for Regression and Classification:** Overview of supervised learning, Linear regression models and least squares, Multiple regression, Multiple outputs, Subset selection , Ridge regression, Lasso regression , Linear Discriminant Analysis , Logistic regression , Perceptron learning algorithm.

**UNIT II**

**Model Assessment and Selection:** Bias, Variance, and model complexity, Bias-variance trade off, Optimism of the training error rate , Estimate of In-sample prediction error, Effective number of parameters, Bayesian approach and BIC, Cross- validation , Boot strap methods, conditional or expected test error.

**UNIT III**

**Additive Models, Trees and Boosting:** Generalized additive models, Regression and classification trees , Boosting methods-exponential loss and AdaBoost, Numerical Optimization via gradient boosting, Examples ( Spam data, California housing , New Zealand fish, Demographic data)

**UNIT IV**

**Neural Networks(NN) , Support Vector Machines(SVM) and K-nearest Neighbor:** Fitting neural networks, Back propagation, Issues in training NN, SVM for classification, Reproducing Kernels, SVM for regression, K-nearest –Neighbour classifiers( Image Scene Classification)

**UNIT V**

**Unsupervised Learning and Random forests:** Association rules, Cluster analysis, Principal Components, Random forests and analysis.

**TEXT BOOKS:**

1. Trevor Hastie, Robert Tibshirani, Jerome Friedman , The Elements of Statistical Learning- Data Mining, Inference, and Prediction , Second Edition , Springer Verlag, 2009.
2. E. Alpaydin, Introduction to Machine Learning, Prentice Hall Of India, 2010
3. C.M. Bishop –Pattern Recognition and Machine Learning, Springer, 2006

**M.TECH (DATA SCIENCES)  
I YEAR II SEMESTER**

**DS2L10**

**MACHINE LEARNING LAB  
(LAB-3)**

**Objectives:**

- To implement Mathematical and statistical concepts for inferences using R/Python
- To interpret the data in high dimensional space
- To Perform descriptive analytics over massive data
- To work on statistical machine learning based Classification, Graphical and computational methods

**Implement the following methods using R or Python**

1. **Graphical Analysis:** Creating a simple graph, Modifying the plots and lines of a graph. Modifying Title and Subtitle, Modifying Axes of a Graph, Adding Legend on a Graph, Special Graphs (Using pie charts, barplot, histograms scatter plots), Multiple Plots.
2. **Descriptive Statistics:** Measure of Central Tendency (Mean, Median and Mode), Measure of Positions (Quartiles, Deciles, Percentiles and Quantiles), Measure of Dispersion (Range, Median, Absolute deviation about median, Variance and Standard deviation), Measure of Distribution (Skewness and Kurtosis), Box and Whisker Plot (Box Plot and its parts, Using Box Plots to compare distribution).
3. **Comparing Population:** Test of Hypothesis, Cross Tabulations (Contingency table and their use, Chi-Square test, Fisher's exact test). One Sample t test, Independent t test, Paired Sample t test and one way ANOVA.
4. Experiments based on Linear Regression and Multiple Linear Regression Methods.
5. Experiments based on Logistic Regression and Multiple Logistic Regression Methods.
6. Implement K-nearest neighbors and Linear Discriminant Analysis.
7. Implement Leave-One-Out Cross-Validation and k-Fold cross-validation on any simulated data set.
8. Experiment the bootstrap to estimate the standard errors of the coefficients from a linear regression fit.
9. Implement Lasso, Ridge and Principal Components regression.
10. Construct classification and regression trees
11. Create a synthesis data set with 1000 observations. Construct random forest and apply bagging to the training set. Estimate test set MSE for this approach.
12. Implement Stochastic Gradient Boosting (using the Gradient Boosting Modeling implementation) algorithm.
13. Implement Support Vector Machine with a Radial Basis Kernel Function.
14. Implement K-Means and Hierarchical Clustering.

**REFERENCES:**

- 1.The Elements of Statistical Learning (Data Mining,Inference,and Prediction), by Trevor Hastie, Robert Tibshirani, Jerome Friedman, Springer.
- 2.Pattern Recognition and Machine Learning by Bishop, Christopher M.
- 3.Kernel Methods for Pattern Analysis by John Shawe-Taylor, NelloCristianini.

**M.TECH (DATA SCIENCES)  
I YEAR II SEMESTER**

**DS2L21**

**MOBILE APPLICATION DEVELOPMENT LAB  
(ELECTIVE -3)  
(LAB-4)**

**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.

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.
14. Develop an application that shows the current location's latitude and longitude continuously as the device is moving (tracking).
15. 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/0B8rTtW\\_91YclTWF4czdBMEpZcWs/view](https://drive.google.com/file/d/0B8rTtW_91YclTWF4czdBMEpZcWs/view)

**M.TECH (DATA SCIENCES)  
I YEAR II SEMESTER**

**DS2L22**

**COMPUTER FORENSICS LAB  
(ELECTIVE -3)  
(LAB-4)**

To perform the following tasks for the lab, Internet facility and open source tools should be provided.

1. Use a Web search engine, such as Google or Yahoo!, and search for companies specializing in computer forensics. Select three and write a two-to three-page paper comparing what each company does.(Project 1-1)
2. Search the Internet for articles on computer crime prosecutions. Find at least two. Write one to two pages summarizing the two articles and identify key features of the decisions you find in your search. (Project 1-5)
3. Use a Web search engine, search for various computer forensics tools.
4. Preparing and processing of investigations. Try to examine and identify the evidences from the drives. (Project 2-1)
5. Extracting of files that have been deleted.( Project 2-4)
6. Illustrate any Data acquisition method and validate. Use an open source data acquisition tool.
7. You're investigating an internal policy violation when you find an e-mail about a serious assault for which a police report needs to be filed. What should you do? Write a two-page paper specifying who in your company you need to talk to first and what evidence must be turned over to the police.(Project 5-2)
8. Create a file on a USB drive and calculate its hash value in FTK Imager. Change the file and calculate the hash value again to compare the files.(Project 5-4)
9. Compare two files created in Microsoft Office to determine whether the files are different at the hexadecimal level. Keep a log of what you find. (Project 6-1)
10. Illustrate the analysis of forensic data.
11. Illustrate the validating of forensic data.
12. Locate and extract Image (JPEG) files with altered extensions.(Project 10-1)
13. Examine or Investigate an E-mail message.

**TEXT BOOKS:**

1. "Computer Forensics and Investigations", Nelson, Phillips Enfinger, Steuart, 3<sup>rd</sup> Edition, Cengage Learning.

**M.TECH (DATA SCIENCES)  
I YEAR II SEMESTER**

**DS2L23**

**NETWORK PROGRAMMING LAB  
(ELECTIVE -3)  
(LAB-4)**

**Objectives:**

- To gain hands-on experiences in installing and administering computer systems and networks, in particular, the UNIX version.
- To implement networking and Internet protocols via programming and TCP/IP protocol architecture; user datagram protocol.
- TO implement shell script that accepts a list of files.

**LIST OF SAMPLE PROBLEMS/EXPERIMENTS:**

1. Write a shell script that accepts a file name, starting and ending line numbers as arguments and displays all the lines between the given line numbers.
2. Write a shell script that deletes all lines containing a specified word in one or more files supplied as arguments to it.
3. Write a shell script that displays a list of all the files in the current directory to which the user has read, write and execute permissions.
4. Write a shell script that receives any number of file names as arguments checks if every argument supplied is a file or a directory and reports accordingly. Whenever the argument is a file, the number of lines on it is also reported.
5. Write a shell script that accepts a list of file names as its arguments, counts and reports the occurrence of each word that is present in the first argument file on other argument files.
6. Write a shell script that accepts any number of arguments and prints them in the reverse order.
7. Write a shell script that determines the period for which a specified user is working on the system.
8. Write a shell script to list all of the directory files in a directory.
9. Write an interactive file-handling shell program- Let it offer the user the choice of copying, removing or linking files. Once the user has made a choice, have the program ask him for the necessary information such as the file name, new name and so on.
10. Write a shell script to find factorial of a given integer.
11. Write a shell script to find the G.C.D. of two integers.
12. Write a shell script to generate a multiplication table.
13. Write a shell script that copies multiple files to a directory.
14. Write a shell script that counts the number of lines and words present in a given file. \*15. Write a shell script that displays the list of all files in the given directory.

15. Write a shell script (small calculator) that adds, subtracts, multiplies and divides the given two integers. There are two division options: one returns the quotient and the other returns remainder. The script requires 3 arguments: The operation to be used and two integer numbers. The options are add (-a), subtract (-s), multiply (-m), quotient (-c) and remainder (-r).
16. Write a shell script to reverse the rows and columns of a matrix.
17. Write a sed command that deletes the first character in each line in a file.
18. Write sed command that deletes the character before the last character in each line a file.
19. Write a sed command that swaps the first and second words in each line of a file.
20. Write an awk script that reads a file of which each line has 5 fields – ID, NAME, MARKS1, MARKS2, MARKS3 and finds out the average for each student. Print out the average marks with appropriate messages.
21. Write an awk script to find the factorial of a user supplied number.
22. ls -l command produces long listing of files.
23. Write an awk script 1) to print the selected fields (Ex: size and name of the files) from the file listing. 2) to print the size of all files and number of files.
24. Write an awk script to count the number of lines in a file that do not contain vowels.
25. Write an awk script to find the number of characters, words and lines in a file.
26. Write a c program that makes a copy of a file using
  - a. Standard I/O
  - b. System calls.
27. Write a C program that counts the number of blanks in a text file
  - a. Using standard I/O
  - b. Using system calls
28. Implement in C the following UNIX commands using system calls
  - a. cat
  - b. ls
  - c. mv
29. Write a program that takes one or more file/directory names as command line input and reports the following information on the file.
  - i. File type.
  - ii. Number of links.
  - iii. Time of last access.
  - iv. Read, Write and Execute permissions.
30. Write a c program to emulate the UNIX ls -l command.
31. Write a c program that creates a directory, puts a file into it, and then removes it.
32. Write a c program that searches for a file in a directory and reports whether the file is present in the directory or not.
33. Write a c program to list for every file in a directory, its inode number and file name.
34. Write a c program that creates a file containing hole which is occupying some space but having nothing.
35. Write a c program that demonstrates redirection of standard output to a file.  
Ex: ls > fl.
36. Write a c program to create a child process and allow the parent to display “parent” and the child to display “child” on the screen.
37. Write a c program to create a Zombie process.
38. Write a c program that illustrates how an orphan is created.



39. Write a c program that creates a child process to execute a command. The command to be executed is passed on the command line.
40. Write a c program that accepts two small numbers as arguments and then sums the two numbers in a child process. The sum should be returned by child to the parent as its exit status and the parent should print the sum.
41. Write a c program that illustrates how to execute two commands concurrently with a command pipe.  
Ex:- ls -l | sort
42. Write c programs that illustrate communication between two unrelated processes using named pipe.
43. Write a c program in which a parent writes a message to a pipe and the child reads the message.
44. Write a c program that illustrates suspending and resuming processes using signals.
45. Write a c program that displays the real time of a day every 60 seconds, 10 times.
46. Write a c program that runs a command that is input by the user and prints the exit status if the command completes in 5 seconds. If it doesn't, then the parent uses kill to send a SIGTERM signal to kill the child process.
47. Write a C program that illustrates file-locking using semaphores.
48. Write a C program that implements a producer-consumer system with two processes. (Using semaphores).
49. Write client and server programs (using C) for
  - a. Interaction between server and client processes using Unix Domain Sockets.
  - b. Interaction between server and client processes using Internet Domain Sockets.
50. Write a C program (sender.c)
  - i. To create a message queue with read and write permissions.
  - ii. To write 3 messages to it with different priority numbers.
51. Write a C program (receiver.c) that receives the messages (from the above message queue as specified in 63.a) and displays them.
52. Write C program that illustrates two processes communicating via shared memory.
53. Design TCP iterative Client and server application to reverse the given input sentence
54. Design TCP iterative Client and server application to reverse the given input sentence
55. Design TCP client and server application to transfer file
56. Design a TCP concurrent server to convert a given text into upper case using multiplexing system call "select"
57. Design a TCP concurrent server to echo given set of sentences using poll functions
58. Design UDP Client and server application to reverse the given input sentence
59. Design UDP Client server to transfer a file
60. Design using poll client server application to multiplex TCP and UDP requests for converting a given text into upper case.
61. Design a RPC application to add and subtract a given pair of integers

**M.TECH (DATA SCIENCES)  
I YEAR II SEMESTER**

**DS2L24**

**ADVANCED PROGRAMMING LAB  
(ELECTIVE -3)  
(LAB-4)**

**Note: The problems given below are only sample problems.**

1. Write a Python program that reads a list of names and ages, then prints the list sorted by age.
2. Write a Python program that will prompt the user for a file name, read all the lines from the file into a list, sort the list, and then print the lines in sorted order.
3. Write a Python program that asks the user for a file name, and then prints the number of characters, words, and lines in the file.
4. Write a Python program that will prompt the user for a string and a file name, and then print all lines in the file that contain the string.
5. Create a class Rectangle. The constructor for this class should take two numeric arguments, which are the length and breadth. Add methods to compute the area and perimeter of the rectangle, as well as methods that simply return the length and breadth. Add a method isSquare that returns a Boolean value if the Rectangle is a Square.
6. Write a class Complex for performing arithmetic with complex numbers. The constructor for this class should take two floating-point values. Add methods for adding, subtracting, and multiplying two complex numbers.
7. Write a Python program that converts a fully parenthesized arithmetic expression from infix to postfix.
8. Write a Python program that reads a postfix expression from standard input, evaluates it, and writes the value to standard output.
9. Write a Python program that takes a command-line argument n and writes the number of primes less than or equal to n.
10. Write a Python program that implements binary search method to search for a key in a sorted list.

**Sample problems covering data structures:**

11. Write Python program to implement the List ADT using a linked list.
12. Write Python programs to implement the deque (double ended queue) ADT using

- a) Array    b) Singly linked list    c) Doubly linked list.
13. Write a Python program to implement priority queue ADT.
14. Write a Python program to perform the following operations:
- a) Construct a binary search tree of elements.
  - b) Search for a key element in the above binary search tree.
  - c) Delete an element from the above binary search tree.
15. Write a Python program to implement all the functions of a dictionary (ADT) using Hashing.
16. Write Python programs that use recursive and non-recursive functions to traverse the given binary tree in
- a) Preorder    b) Inorder    c) Postorder.
17. Write Python programs for implementing the following sorting methods:
- a) Merge sort
  - b) Insertion sort                      e) Heap sort
  - c) Quick sort                          f) Radix sort
18. Write a Python program that counts the occurrences of words in a text file and displays the words in decreasing order of their occurrence counts.
19. Write a Python program that prompts the user to enter a directory or a filename and displays its size.
20. Write a Python program that uses a recursive function to print all the permutations of a string.
21. Write a Python program that prompts the user to enter a directory and displays the number of files in the directory.
22. Suppose the password rules are as follows:
- i) A password must have at least eight characters.
  - ii) A password must consist of only letters and digits.
  - iii) A password must contain at least two digits.
- Write a Python program that prompts the user to enter a password (string)

and displays whether it is valid or invalid password.

23. Write a Python program to compute  $n!$  for large values of  $n$ .

24. In data compression, a set of strings is prefix-free if no string is a prefix of another.

For example, the set of strings 01, 10, 0010, and 1111 is prefix-free, but the set of

strings 01, 10, 0010, 1010 is not prefix-free because 10 is a prefix of 1010. Write a Python program that reads a set of strings from standard input and determines whether the set is prefix-free.

### **Sample problems covering Networking applications:**

25. Write Echo Server and Client programs in Python.

26. Develop Echo Server in Python, one that can handle multiple clients in parallel using forking model on Unix platform.

27. Develop Echo Server in Python, one that can handle multiple clients in parallel using threading model on Unix and Microsoft windows platforms.

28. Develop Echo Server in Python, one that can handle multiple clients in parallel by multiplexing client connections and the main dispatcher with the select system call.

Here a single event loop can process clients and accept new ones in parallel.

29. Implement in Python client and server-side logic to transfer an arbitrary file from server to client over a socket.

### **Sample problems covering GUI applications, Web applications and Database applications:**

#### **GUI applications:**

1. Write a Python program that works as a simple calculator. Use a grid to arrange buttons for the digits and for the +, -, \*, % operations. Add a text field to display the result.

2. Develop a Python GUI application that receives an integer in one text field, and computes its factorial Value and fills it in another text field, when the button named “Compute” is clicked.

3. Write a Python program that creates a user interface to perform integer divisions. The user enters two numbers in the text fields, Num1 and Num2. The division of Num1 and Num2 is displayed in the Result field when the Divide button is clicked. If Num1 or Num2 were not an integer Num2 is Zero, the program should Display an appropriate message in the result field in Red color.

4. Write a Python program that simulates a traffic light. The program lets the user select one of three lights: red, yellow, or green. When a radio button is selected, the light is turned on, and only one light can be on at a time. No light is on when the program starts.

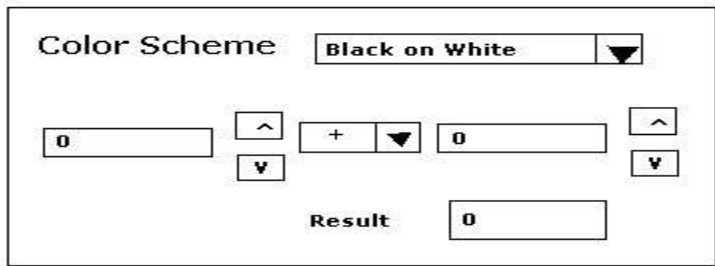
### Web Applications

1. Create a registration form with User name, Password and Aadhar Number. Validate the fields for correctness using JavaScript.
2. Create a database for registration and store the submitted values in the local database using serverside Python programs. If user name already exists, send back an error page.
3. Using cookies, display the user's last login time on the welcome page when the user logs into the website.
  - a) Write an HTML page including any required Javascript that takes a number from one text field in the range of 0 to 999 and shows it in another text field in words. If the number is out of range, it should show "out of range" and if it is not a number, it should show "not a number" message in the result box.
  - b) Implement the same program in standalone GUI Program.
    4. Write a Python GUI application that takes a text file name as input and counts the characters, words and lines in the file. Words are separated with white space characters and lines are separated with new line character.
    5. Write a simple calculator web application that takes two numbers and an operator (+, -, /, \* and %) from an HTML page and sends the result page with the operation performed on the operands. It should check in a database if the same expression is already computed and if so, just return the value from database. Use MySQL or PostgreSQL.
    6. Write an HTML page that contains a list of 5 countries. When the user selects a country, its capital should be printed next to the list. Add CSS to customize the properties of the font of the capital (color, bold and font size).
    7. Write a web Python application that takes name and age from an HTML page. If the age is less than 18, it should send a page with "Hello <name>, you are not authorized to visit this site" message, where <name> should be replaced with the entered name. Otherwise it should send "Welcome <name> to this site" message.
    8. Write a calculator program in HTML that performs basic arithmetic operations (+, -, /, \* and %). Use CSS to change the foreground and background color of the values, buttons and result

display area separately. Validate the input strings using JavaScript regular expressions. Handle any special cases like division with zero reasonably. The screen may look similar to the following:

<b>Value 1</b>	<b>Operator</b>	<b>Value 2</b>	<b>=</b>	<b>Result</b>
<input type="text"/>	<input type="text" value="+"/> ▼	<input type="text"/>	<input type="text" value="="/>	<input type="text"/>

9. Write a Python program that creates a calculator GUI, as shown in figure. Extra components may be added for convenience:



The Color Scheme may be Black on White or Blue on Yellow (selectable) and accordingly all components colors must be changed. The values can be either entered or increased or decreased by a step of 10. The operators are +, -, / and \* (selectable). Once any change takes place, the result

must be automatically computed by the program.

10. Write a Python Application that will read a text file that contains personal information (Name, Mobile Number, age and place (fields are separated by tabs and records are separated with new line). The first line contains the header with field names. After reading the information, it shows two input Text Fields in a window, one for name and the other for value. Once these two values are given, it should list all the records in the file that match the value of the given field in a text area (result box). For example, if the two text boxes are entered with "name" and "ABCD" then it should show all the records for which name is "ABCD". An Illustration is given below that takes a mobile number and lists all the records that have the same mobile number.

<b>Field</b>	<input type="text" value="mobile"/>	
<b>Value</b>	<input type="text" value="9449449449"/>	<input type="button" value="OK"/>
<b>Result</b>	<input type="text" value="abc, 22, Hyd"/> <input type="text" value="def, 23, Delhi"/> <input type="text" value="xxx, 44, Chennai"/>	

11. Consider the following web application for implementation:

- a. The user is first served a login page which takes user's name and password. After submitting the details the server checks these values against the data from a database and takes the following decisions.
- b. If name and password matches, serves a welcome page with user's full name.
- c. If name matches and password doesn't match, then serves "password mismatch" page
- d. If name is not found in the database, serves a registration page, where users full name, present user name (used to login) and password are collected.

### TEXT BOOKS :

1. Exploring Python, Timothy A. Budd, McGraw Hill Publications.
2. Core Python Programming, 2<sup>nd</sup> edition, W.J.Chun, Pearson.
3. Core Python Application Programming, 3rd edition, W.J.Chun, Pearson.
4. Programming Python, 3<sup>rd</sup> edition, Mark Lutz, SPD,O'Reilly.
5. The Python 3 Standard Library by Example, Doug Hellmann, Pearson.
6. Introduction to Programming using Python, Y.Daniel Liang, Pearson.
7. Introduction to Programming in Python, R.Sedgewick, K. Wayne and R.Dondero, Pearson.
8. [www.python.org](http://www.python.org) web site.
9. Python Programming, R.Thareja, Oxford University Press.
- 10.Data structures and Algorithms using Python, Rance D.Necaise, Wiley Student edition.
- 11.Data structures and Algorithms in Python, M.T.Goodrich,R.Tamassia,M.H.Goldwasser, Wiley Student edition.

**M.TECH (DATA SCIENCES)  
I YEAR II SEMESTER**

**DS2L25**

**CRYPTOGRAPHY AND NETWORK SECURITY LAB  
(ELECTIVE -3)  
(LAB-4)**

The experiments are divided into **Part-A & B.**

**PART – A**

Exercises are based on the cryptographic algorithms. They can be implemented using C, C++, Java, etc.

1. Write a Java program to perform encryption and decryption using the following
2. algorithms a. Ceaser cipher b. Substitution cipher c. Hill Cipher
3. Write a C/JAVA program to implement the DES algorithm logic.
4. Write a C/JAVA program to implement the Blowfish algorithm logic.
5. Write a C/JAVA program to implement RSA algorithm.
6. Write a C/JAVA program to implement the Rijndael algorithm logic.
7. Using Java cryptography, encrypt the text “Hello world” using Blowfish. Create your
8. own key using Java keytool.
9. Implement the Diffie-Hellman Key Exchange mechanism using HTML and JavaScript.
10. Consider the end user as one of the parties(Alice) and the JavaScript application as the
11. other party(Bob)
12. Calculate the message digest of a text using the SHA-1 algorithm in JAVA.
13. Calculate the message digest of a text using the MD5 algorithm in JAVA.

**Part – B**

Exercises have to be performed using various software tools/utilities mentioned

**1. Passive Information Gathering**

- a. IP Address and Domain Identification of log entries – DNS, RIR, etc tools
- b. Information Gathering of a web site: WHOIS, ARIN, etc tools
- c. Banner Grabbing: Netcat, etc tools

**2. Detecting Live Systems**

- a. Port Scanning : Nmap, SuperScan
- b. Passive Fingerprinting: Xprobe2
- c. Active Fingerprinting: Xprobe2

**3. Enumerating Systems**

- a. SNMP Enumeration: SolarWinds IP Network Browser,  
[www.solarwinds.com/downloads](http://www.solarwinds.com/downloads)
- b. Enumerating Routing Protocols: Cain & Abel tool, [www.oxid.it](http://www.oxid.it)

**4. Automated Attack and Penetration Tools**

- a. Exploring N-Stalker, a Vulnerability Assessment Tool, [www.nstalker.com](http://www.nstalker.com)

**5. Defeating Malware**



- a. Building Trojans, Rootkit Hunter: [www.rootkit.nl/projects/rootkit\\_hunter.html](http://www.rootkit.nl/projects/rootkit_hunter.html)
- b. Finding malware

#### **6. Securing Wireless Systems**

- a. Scan WAPs: NetStumbler, [www.netstumbler.com/downloads](http://www.netstumbler.com/downloads)
- b. Capture Wireless Traffic: Wireshark, [www.wireshark.org](http://www.wireshark.org)

**M.TECH (DATA SCIENCES)  
I YEAR II SEMESTER**

**DS2A11**

**ENGLISH FOR RESEARCH PAPER WRITING  
(AUDIT – 2)**

**Course objectives:**

Students will be able to:

1. Understand that how to improve your writing skills and level of readability
2. Learn about what to write in each section
3. Understand the skills needed when writing a Title

Ensure the good quality of paper at very first-time submission

**UNIT I**

Planning and Preparation, Word Order, Breaking up long sentences, Structuring Paragraphs and Sentences, Being Concise and Removing Redundancy, Avoiding Ambiguity and Vagueness

**UNIT II**

Clarifying Who Did What, Highlighting Your Findings, Hedging and Criticising, Paraphrasing and Plagiarism, Sections of a Paper, Abstracts. Introduction

**UNIT III**

Review of the Literature, Methods, Results, Discussion, Conclusions, The Final Check.

**UNIT IV**

key skills are needed when writing a Title, key skills are needed when writing an Abstract, key skills are needed when writing an Introduction, skills needed when writing a Review of the Literature,

**UNIT V**

skills are needed when writing the Methods, skills needed when writing the Results, skills are needed when writing the Discussion, skills are needed when writing the Conclusions

**UNIT VI**

useful phrases, how to ensure paper is as good as it could possibly be the first- time submission

**SUGGESTED STUDIES:**

1. Goldbort R (2006) Writing for Science, Yale University Press (available on Google Books)
2. Day R (2006) How to Write and Publish a Scientific Paper, Cambridge University Press
3. Highman N (1998), Handbook of Writing for the Mathematical Sciences, SIAM. Highman'sbook.
4. Adrian Wallwork, English for Writing Research Papers, Springer New York Dordrecht Heidelberg London, 2011

**M.TECH (DATA SCIENCES)  
I YEAR II SEMESTER**

**DS2A12**

**DISASTER MANAGEMENT  
(AUDIT – 2)**

**Course Objectives: -**

Students will be able to:

1. learn to demonstrate a critical understanding of key concepts in disaster risk reduction and humanitarian response.
2. critically evaluate disaster risk reduction and humanitarian response policy and practice from multiple perspectives.
3. develop an understanding of standards of humanitarian response and practical relevance in specific types of disasters and conflict situations.
4. critically understand the strengths and weaknesses of disaster management approaches, planning and programming in different countries, particularly their home country or the countries they work in

**UNIT I**

**Introduction** Disaster: Definition, Factors And Significance; Difference Between Hazard And Disaster; Natural And Manmade Disasters: Difference, Nature, Types And Magnitude.

**UNIT II**

**Repercussions Of Disasters And Hazards:** Economic Damage, Loss Of Human And Animal Life, Destruction Of Ecosystem. Natural Disasters: Earthquakes, Volcanisms, Cyclones, Tsunamis, Floods, Droughts And Famines, Landslides And Avalanches, Man-made disaster: Nuclear Reactor Meltdown, Industrial Accidents, Oil Slicks And Spills, Outbreaks Of Disease And Epidemics, War And Conflicts.

**UNIT III**

**Disaster Prone Areas In India** Study Of Seismic Zones; Areas Prone To Floods And Droughts, Landslides And Avalanches; Areas Prone To Cyclonic And Coastal Hazards With Special Reference To Tsunami; Post-Disaster Diseases And Epidemics

**UNIT IV**

**Disaster Preparedness And Management** Preparedness: Monitoring Of Phenomena Triggering A Disaster Or Hazard; Evaluation Of Risk: Application Of Remote Sensing, Data From Meteorological And Other Agencies, Media Reports: Governmental And Community Preparedness.

**UNIT V**

**Risk Assessment** Disaster Risk: Concept And Elements, Disaster Risk Reduction, Global And National Disaster Risk Situation. Techniques Of Risk Assessment, Global Co-Operation In Risk Assessment And Warning, People's Participation In Risk Assessment. Strategies for Survival.

**UNIT VI**

**Disaster Mitigation** Meaning, Concept And Strategies Of Disaster Mitigation, Emerging Trends In Mitigation. Structural Mitigation And Non-Structural Mitigation, Programs Of Disaster Mitigation In India.

**SUGGESTED READINGS:**

1. R. Nishith, Singh AK, “Disaster Management in India: Perspectives, issues and strategies “New Royal book Company.
2. Sahni, PardeepEt.Al. (Eds.),” Disaster Mitigation Experiences And Reflections”, Prentice Hall Of India, New Delhi.
3. Goel S. L., Disaster Administration And Management Text And Case Studies”,Deep &Deep Publication Pvt. Ltd., New Delhi.

**M.TECH (DATA SCIENCES)  
I YEAR II SEMESTER**

**DS2A13**

**SOFT SKILLS  
(AUDIT – 2)**

**Course Objectives**

- To improve the fluency of students in English
- To facilitate learning through interaction
- To illustrate the role of skills in real-life situations with case studies, role plays etc.
- To train students in group dynamics, body language and various other activities which boost their confidence levels and help in their overall personality development
- To encourage students develop behavioral skills and personal management skills
- To impart training for empowerment, thereby preparing students to become successful professionals

**Learning Outcomes**

- Developed critical acumen and creative ability besides making them industry- ready.
- Appropriate use of English language while clearly articulating ideas.
- Developing insights into Language and enrich the professional competence of the students.
- Enable students to meet challenges in job and career advancement.

**UNIT I : INTRODUCTION**

Definition and Introduction to Soft Skills – Hard Skills vs Soft Skills – Significance of Soft/Life/Self Skills – Self and SWOT Analysis *and*

**1. Exercises on Productivity Development**

- Effective/ Assertive Communication Skills (Activity based)
- Time Management (Case Study)
- Creativity & Critical Thinking (Case Study)
- Decision Making and Problem Solving (Case Study)
- Stress Management (Case Study)

**2. Exercises on Personality Development Skills**

- Self-esteem (Case Study)
- Positive Thinking (Case Study)
- Emotional Intelligence (Case Study)
- Team building and Leadership Skills (Case Study)
- Conflict Management (Case Study)

**3. Exercises on Presentation Skills**

- Netiquette

- Importance of Oral Presentation – Defining Purpose- Analyzing the audience- Planning Outline and Preparing the Presentation- Individual & Group Presentation- Graphical Organizers- Tools and Multi-media Visuals
- One Minute Presentations (Warming up)
- PPT on Project Work- Understanding the Nuances of Delivery- Body Language – Closing and Handling Questions – Rubrics for Individual Evaluation (Practice Sessions)

#### 4. Exercises on Professional Etiquette and Communication

- Role-Play and Simulation- Introducing oneself and others, Greetings, Apologies, Requests, Agreement & Disagreement...etc.
- Telephone Etiquette
- Active Listening
- Group Discussions (Case study)- Group Discussion as a part of Selection Procedure- Checklist of GDs
- Analysis of Selected Interviews (Objectives of Interview)
- Mock-Interviews (Practice Sessions)
- Job Application and Preparing Resume
- Process Writing (Technical Vocabulary) – Writing a Project Report- Assignments

#### 5. Exercises on Ethics and Values

Introduction — Types of Values - Personal, Social and Cultural Values - Importance of Values in Various Contexts

- Significance of Modern and Professional Etiquette – Etiquette (Formal and Informal Situations with Examples)
- Attitude, Good Manners and Work Culture (Live Examples)
- Social Skills - Dealing with the Challenged (Live Examples)
- Professional Responsibility – Adaptability (Live Examples)
- Corporate Expectations

- Note: Hand-outs are to be prepared and given to students.
- Training plan will be integrated in the syllabus.
- Topics mentioned in the syllabus are activity-based.

#### SUGGESTED SOFTWARE:

- The following software from ‘[train2success.com](http://train2success.com)’
  - Preparing for being Interviewed
  - Positive Thinking
  - Interviewing Skills
  - Telephone Skills
  - Time Management
  - Team Building
  - Decision making

**SUGGESTED READING**

1. Alex, K. 2012. *Soft Skills*. S. Chand Publishers
2. Naterop, B. Jean and Revell, Rod. 2004. *Telephoning in English*. Cambridge: CUP
3. Patnaik, P. 2011. *Group Discussion and Interview Skills*. New Delhi: Foundation
4. Rizvi, M. A. 2005. *Effective Technical Communication*. New Delhi: Tata McGraw Hill
5. **Sasikumar, V & Dhamija, P.V. 1993. *Spoken English - A Self-Learning Guide to Conversation Practice*. New Delhi: Tata McGraw-Hill**
6. Sudhir Andrews. 2009. *How to Succeed at Interviews*. New Delhi: Tata McGraw Hill
7. Vivekananda: His Call to the Nation : a Compilation R.K. Math Publication

**M.TECH (DATA SCIENCES)  
I YEAR II SEMESTER**

**DS2A14**

**STRESS MANAGEMENT BY YOGA  
AUDIT – 2**

**Course Objectives**

- To achieve overall health of body and mind
- To overcome stress

**UNIT I**

Definitions of Eight parts of yog. ( Ashtanga )

**UNIT II**

Yam and Niyam.

Do`s and Don`t`s in life.

- a) Ahinsa, satya, astheya, bramhacharya and aparigraha
- ii) Shaucha, santosh, tapa, swadhyay, ishwarpranidhan

**UNIT III**

Asan and Pranayam

- b) Various yog poses and their benefits for mind & body
- ii)Regularization of breathing techniques and its effects-Types of pranayam

**SUGGESTED READING:**

1. ‘Yogic Asanas for Group Training-Part-I’ :Janardan Swami Yogabhyasi Mandal, Nagpur
2. “Rajayoga or conquering the Internal Nature” by Swami Vivekananda, AdvaitaAshrama (Publication Department), Kolkata



**M.TECH (DATA SCIENCES)  
I YEAR II SEMESTER**

**DS2A15**

**SANSKRIT FOR TECHNICAL KNOWLEDGE  
(AUDIT – 2)**

**Course Objectives:**

1. To get a working knowledge in illustrious Sanskrit, the scientific language in the world
2. Learning of Sanskrit to improve brain functioning
3. Learning of Sanskrit to develop the logic in mathematics, science & other subjects
4. enhancing the memory power
5. The engineering scholars equipped with Sanskrit will be able to explore the
6. huge knowledge from ancient literature

**UNIT I**

Alphabets in Sanskrit,  
Past/Present/Future Tense  
Simple Sentences

**UNIT II**

Order  
Introduction of roots  
Technical information about Sanskrit Literature

**UNIT III**

Technical concepts of Engineering-Electrical, Mechanical, Architecture, Mathematics

**SUGGESTED READING:**

1. “Abhyaspustakam” – Dr. Vishwas, Samskrita-Bharti Publication, New Delhi
2. “Teach Yourself Sanskrit” Prathama Deeksha-VempatiKutumbshastri, Rashtriya Sanskrit Sansthanam, New Delhi Publication
3. “India’s Glorious Scientific Tradition” Suresh Soni, Ocean books (P) Ltd., New Delhi

**M.TECH (DATA SCIENCES)  
I YEAR II SEMESTER**

**DS2A16**

**RESEARCH METHODOLOGY  
(AUDIT -2)**

**UNIT I**

Introduction:

Objectives of Research; Definition and Motivation; Types of Research; Research Approaches; Steps in Research Process; Criteria of Good Research, Ethics in Research.

Research Formulation and Literature Review:

Problem Definition and Formulation; Literature Review; Characteristics of a Good Research Question; Literature Review Process.

**UNIT II**

Data Collection:

Primary and Secondary Data; Primary and Secondary Data Sources; Data Collection Methods; Data Processing; Classification of Data.

Basic Statistical Measures:

Types of Scales; Measures of Central Tendency; Skewness; Measure of Variation; Probability Distribution.

**UNIT III**

Data Analysis:

Statistical Analysis; Multivariate Analysis; Correlation Analysis; Regression Analysis; Principle Component Analysis; Sampling.

Research Design:

Need for Research Design; Features of a Good Design; Types of Research Designs; Induction and Deduction.

**UNIT IV**

Hypothesis Formulation and Testing:

Hypothesis; Important Terms; Types of Research Hypothesis; Hypothesis Testing; Z-Test; t-Test; f-Test; Making a Decision: Types of Errors; ROC Graphics.

Test Procedures:

Parametric and Non-parametric Tests; ANOVA; Mann-Whitney Test; Kruskal-Wallis Test; Chi-Square Test; Multi-Variate Analysis.

**UNIT V**

Models for Science and Business:

Algorithmic Research; Methods of Scientific Research; Modelling; Simulations; Industrial Research.

Presentation of the Research Work:

Business Report; Technical Report; Research Report; General Tips for Writing Report; Presentation of Data; Oral Presentation; Bibliography and References; Intellectual Property Rights; Open-Access Initiatives; Plagiarism.

**TEXT BOOK:**

1. Research Methodology - Pearson Publications – S.S. Vinod Chandra; S. Anand Hareendran

**REFERENCE BOOKS:**

1. Research Methodology - David V. Thiel
2. Research Methodology - R. Panner Selvam

**MASTER OF TECHNOLOGY  
(DATA SCIENCES)  
II YEAR I SEMESTER**

**DS3E11**

**RECOMMENDER SYSTEMS  
(ELECTIVE-5)**

**COURSE OBJECTIVE**

- To learn techniques for making recommendations, including non-personalized, content-based, and collaborative filtering
- To automate a variety of choice-making strategies with the goal of providing affordable, personal, and high-quality recommendations

**UNIT I**

**Introduction:** Overview of Information Retrieval, Retrieval Models, Search and Filtering Techniques: Relevance Feedback, User Profiles, Recommender system functions, Matrix operations, covariance matrices, Understanding ratings, Applications of recommendation systems, Issues with recommender system.

**UNIT II**

**Content-based Filtering:** High level architecture of content-based systems, Advantages and drawbacks of content based filtering, Item profiles, Discovering features of documents, pre-processing and feature extraction, Obtaining item features from tags, Methods for learning user profiles, Similarity based retrieval, Classification algorithms.

**UNIT III**

**Collaborative Filtering:** User-based recommendation, Item-based recommendation, Model based approaches, Matrix factorization, Attacks on collaborative recommender systems.

**UNIT IV**

**Hybrid approaches:** Opportunities for hybridization, Monolithic hybridization design: Feature combination, Feature augmentation, Parallelized hybridization design: Weighted, Switching, Mixed, Pipelined hybridization design: Cascade Meta-level, Limitations of hybridization strategies

**UNIT V**

**Evaluating Recommender System:** Introduction, General properties of evaluation research, Evaluation designs: Accuracy, Coverage, confidence, novelty, diversity, scalability, serendipity, Evaluation on historical datasets, Offline evaluations. **Types of Recommender Systems:** Recommender systems in personalized web search, knowledge-based recommender system, Social tagging recommender systems, Trust-centric recommendations, Group recommender systems.

**REFERENCES:**

1. Jannach D., Zanker M. and FelFering A., Recommender Systems: An Introduction, Cambridge University Press (2011), 1st ed.
2. Charu C. Aggarwal, Recommender Systems: The Textbook, Springer (2016), 1st ed.
3. Ricci F., Rokach L., Shapira D., Kantor B.P., Recommender Systems Handbook, Springer(2011), 1st ed.
4. Manouselis N., Drachsler H., Verbert K., Duval E., Recommender Systems For Learning, Springer

**MASTER OF TECHNOLOGY  
(DATA SCIENCES)  
II YEAR I SEMESTER**

**DS3E12**

**DEEP LEARNING  
(ELECTIVE-5)**

**OBJECTIVES:**

- Understand the fundamentals of Deep learning
- Understand the various techniques used for deep learning, and comparing with AI and ML,
- Understand and gain the knowledge of various Deep learning techniques and its applications

**UNIT I**

Introduction to Deep learning, Understanding (Artificial Intelligence) AI, Machine Learning (ML), concept of neuron, McCulloch Pitts Neuron, Thresholding Logic, Perceptrons, Perceptron Learning Algorithm and Convergence, Multilayer Perceptrons (MLPs), Representation Power of MLPs.

**UNIT II**

Sigmoid Neurons, Gradient Descent, Feedforward Neural Networks, Representation Power of Feedforward Neural Networks, Feedforward Neural Networks, Backpropagation, Gradient Descent (GD), Momentum Based GD, Nesterov Accelerated GD, Stochastic GD, AdaGrad, RMSProp, Adam, Eigenvalues and eigenvectors, Eigenvalue Decomposition, Basis, Principal Component Analysis and its interpretations, Singular Value Decomposition

**UNIT -III**

Autoencoders and relation to PCA, Regularization in autoencoders, Denoising autoencoders, Sparse autoencoders, Contractive autoencoders, Bias Variance Tradeoff, L2 regularization, Early stopping, Dataset augmentation, Parameter sharing and tying, Injecting noise at input, Ensemble methods, Dropout, Greedy Layer wise Pre-training, Better activation functions, Better weight initialization methods, Batch Normalization

**UNIT- IV**

Learning Vectorial Representations Of Words, Convolutional Neural Networks (CNN), LeNet, AlexNet, ZF-Net, VGGNet, GoogLeNet, ResNet, Object Detection, RCNN, Fast RCNN, Faster RCNN, YOLO, Visualizing Convolutional Neural Networks, Guided Backpropagation, Deep Dream, Deep Art, Fooling Convolutional Neural Networks, Recurrent Neural Networks, Backpropagation Through Time (BPTT), Vanishing and Exploding Gradients, Truncated BPTT

**UNIT V**

**Introduction to** Gated Recurrent Units (GRUs), Long Short Term Memory (LSTM) Cells, Directed Graphical Models, Markov Networks, and Autoregressive Models, Generative Adversarial Networks (GANs).

Application of: Eigen values and Eigen Vectors, Singular value decomposition (SVD), regularization, normalization, Faster RCNN, BPTT and Truncated BPTT, GRUs, LSTM.

**REFERENCES:**

1. <https://www.cse.iitm.ac.in/~miteshk/CS7015.html>
2. Neural Networks A Systematic Introduction - Ra'ul Rojas  
<http://page.mi.fuberlin.de/rojas/neural/neuron>
3. Deep Learning, Ian Goodfellow, Yoshua Bengio and Aaron Courville, The MIT Press.

**MASTER OF TECHNOLOGY  
(DATA SCIENCES)  
II YEAR I SEMESTER**

**DS3E13**

**CLOUD SECURITY  
(ELECTIVE-5)**

**OBJECTIVES:**

- Understand the fundamentals of cloud computing and security issues
- Understand the requirements, policies risks, standards involved in cloud computing and application to be deployed in a cloud.
- gain knowledgeable about different ways to secure cloud computing.

**UNIT I**

**Cloud Computing Fundamentals:** Features, Service models, deployment models, Service level agreements, **Software Security Fundamentals:** Cloud Information Security Objectives, Confidentiality, Integrity, and Availability Cloud Security Services, Relevant Cloud Security Design Principles, Secure Cloud Software Requirements, Secure Development Practices, Handling Data Approaches to Cloud Software Requirements Engineering, Cloud Security Policy Implementation and Decomposition, NIST 33 Security Principles, Secure Cloud Software Testing, Testing for Security Quality Assurance, Cloud Penetration Testing, Regression Testing, Cloud Computing and Business Continuity Planning/Disaster Recovery Definitions

**UNIT II**

Cloud Computing Security Challenges: Security Policy Implementation, Policy Types, Computer Security Incident Response Team (CSIRT), Virtualization Security Management, Virtual Threats, VM Security Recommendations, VM-Specific Security Techniques, Securing the virtual machine, VM Security research, locking down VMware, Microsoft, XenServer VMs.

**UNIT -III**

Cloud Computing Security Architecture: Architectural Considerations, General Issues, Trusted Cloud Computing, Secure Execution Environments and Communications, Micro architectures, Identity Management and Access Control, Identity Management, Access Control, Autonomic Security. **Cloud Computing Virtualization security:** Architecture, threats to virtualization environment, Security adaption, designing virtual networks for security

**UNIT- IV**

**Cloud Computing standards and Life Cycle Issues :**Standards, The Distributed Management Task Force (DMTF), The International Organization for Standardization (ISO),The European Telecommunications Standards Institute (ETSI),The Organization for the Advancement of Structured Information Standards (OASIS),Storage Networking Industry Association (SNIA), Open Grid Forum (OGF),The Open Web Application Security Project (OWASP), Incident Response,NIST Special Publication, Internet Engineering Task Force Incident-Handling Guidelines, Layered Security and IDS, Computer Security and Incident Response Teams, Encryption and Key Management,VM Life cycle.



**UNIT V**

Cloud Computing Risk Issues: The CIA Triad, Privacy and Compliance Risks, Threats to Infrastructure, Data, and Access Control, Common Threats and Vulnerabilities, Cloud Access Control Issues, Cloud Service Provider Risks,

**TEXT BOOKS:**

1. Ronald L. Krutz, Russell Dean Vines, “Cloud Security”, Wiley [ISBN: 0470589876], , 2010.
2. Dave Shackleford “Virtualization Security” sybex wiley brand [ISBN: 978-1-118-28812-2], 2013.

**REFERENCES:**

1. Gautam Shroff, “Enterprise Cloud Computing Technology Architecture Applications”, Cambridge University Press; 1 edition [ISBN: 978- 0521137355], 2010.
2. Toby Velte, Anthony Velte, Robert Elsenpeter, “Cloud Computing, A Practical Approach”, Tata McGraw-Hill Osborne Media; 1 edition 22, [ISBN: 0071626948], 2009.
3. Tim Mather, Subra Kumaraswamy, Shahed Latif, “Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance”, O'Reilly Media; 1 edition, [ISBN: 0596802765]

**MASTER OF TECHNOLOGY  
(DATA SCIENCES)  
II YEAR I SEMESTER**

**DS3E14**

**GPU COMPUTING  
(ELECTIVE-5)**

**COURSE OBJECTIVE**

- To learn parallel programming with Graphics Processing Units (GPUs).

**UNIT I**

**Introduction:** History, Graphics Processors, Graphics Processing Units, GPGPUs. Clock speeds, CPU / GPU comparisons, Heterogeneity, Accelerators, Parallel programming, CUDA OpenCL / OpenACC, Hello World Computation Kernels, Launch parameters, Thread hierarchy, Warps / Wavefronts, Thread blocks / Workgroups, Streaming multiprocessors, 1D / 2D / 3D thread mapping, Device properties, Simple Programs

**UNIT II**

**Memory:** Memory hierarchy, DRAM / global, local / shared, private / local, textures, Constant Memory, Pointers, Parameter Passing, Arrays and dynamic Memory, Multi-dimensional Arrays, Memory Allocation, Memory copying across devices, Programs with matrices, Performance evaluation with different memories

**UNIT III**

**Synchronization:** Memory Consistency, Barriers (local versus global), Atomics, Memory fence. Prefix sum, Reduction. Programs for concurrent Data Structures such as Worklists, Linked-lists. Synchronization across CPU and GPU

**Functions:** Device functions, Host functions, Kernels functions, Using libraries (such as Thrust), and developing libraries.

**UNIT IV**

**Support:** Debugging GPU Programs. Profiling, Profile tools, Performance aspects

**Streams:** Asynchronous processing, tasks, Task-dependence, Overlapped data transfers, Default Stream, Synchronization with streams. Events, Event-based-Synchronization - Overlapping data transfer and kernel execution, pitfalls.

**UNIT V**

**Case Studies:** Image Processing, Graph algorithms, Simulations, Deep Learning

**Advanced topics:** Dynamic parallelism, Unified Virtual Memory, Multi-GPU processing, Peer access, Heterogeneous processing

**COURSE OUTCOMES****After completion of course, students would be:**

- Students would learn concepts in parallel programming, implementation of programs on GPUs, debugging and profiling parallel programs.

**REFERENCES:**

1. Programming Massively Parallel Processors: A Hands-on Approach; David Kirk, Wen-meiHwu; Morgan Kaufman; 2010 (ISBN: 978-0123814722)
2. CUDA Programming: A Developer's Guide to Parallel Computing with GPUs; Shane Cook; Morgan Kaufman; 2012 (ISBN: 978-0124159334).

**MASTER OF TECHNOLOGY  
(DATA SCIENCES)  
II YEAR I SEMESTER**

**DS3E15**

**DISTRIBUTED DATABASES  
(ELECTIVE-5)**

**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, and 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.
2. Principles of Distributed Database Systems, M. Tamer Ozsu, Patrick Valduriez , Pearson Education, 2nd Edition.

**REFERENCE BOOKS:**

1. Distributed Database Systems, Chanda Ray, Pearson.
2. Distributed Database Management Systems, S.K. Rahimi and Frank.S. Haug, Wiley.

**MASTER OF TECHNOLOGY  
(DATA SCIENCES)  
II YEAR I SEMESTER**

**DS3E16**

**ADHOC NETWORKS  
(ELECTIVE-5)**

**Course objectives:**

This course will enable students to

- Explain fundamental principles of Adhoc Networks
- MAC Protocols for Adhoc Wireless Networks
- Routing Protocols for Adhoc Wireless Networks
- Transport Layer Protocols for Ad-hoc Networks
- Security Protocols for Adhoc Networks
- Wireless Sensor Networks

**UNIT I**

**Ad-hoc Wireless Networks:** Introduction, Issues in Ad-hoc Wireless Networks, Ad-hoc Wireless Internet; **MAC Protocols for Ad-hoc Wireless Networks:** Introduction, Issues in Designing a MAC Protocol, Design Goals of MAC Protocols, Classification of MAC protocols, Contention-Based Protocols, Contention-Based Protocols with Reservation Mechanisms.

**UNIT II**

**Routing Protocols for Ad-hoc Wireless Networks:** Introduction, Issues in Designing a Routing Protocol for Ad-hoc Wireless Networks, Classification of Routing Protocols, Table Driven Routing Protocols; On-Demand Routing Protocols, Hybrid Routing Protocols.

**UNIT III**

**Transport Layer Protocols for Adhoc Networks:** Introduction, Issues in Designing a Transport Layer Protocol, Design Goals of a Transport Layer Protocol, Classification of Transport Layer Solutions, TCP over Adhoc Wireless Networks

**UNIT IV**

**Security Protocols for Adhoc Networks:** Security in Adhoc Wireless Networks, Issues and Challenges in Security Provisioning, Network Security Attacks, Key Management, Secure Routing in Adhoc Wireless Networks.

**UNIT V**

**Wireless Sensor Networks:** Introduction, Sensor Network Architecture, Data Dissemination, Data Gathering, MAC Protocols for Sensor Networks

**TEXT BOOK:**

1. C. Siva Ram Murthy & B. S. Manoj: Adhoc Wireless Networks: Architectures and Protocols, 2nd Edition, Pearson Education, 2011

**REFERENCE BOOKS:**

1. Ozan K. Tonguz and Gianguigi Ferrari: Ad-hoc Wireless Networks, John Wiley, 2007.
2. Xiuzhen Cheng, Xiao Hung, Ding-Zhu Du: Ad-hoc Wireless Networking, Kluwer Academic Publishers, 2004.
3. C.K. Toh: Ad-hoc Mobile Wireless Networks- Protocols and Systems, Pearson Education, 2002

**MASTER OF TECHNOLOGY  
(DATA SCIENCES)  
II YEAR I SEMESTER**

**DS3E17**

**REINFORCEMENT LEARNING  
(ELECTIVE-5)**

**Objectives:**

By the end of this course, students should be able to do the following:

- Learn how to define RL tasks and the core principals behind the RL, including policies, value functions, deriving Bellman equations (as assets by the assignments, an exam and quizzes)
- Implement in code common algorithms following code standards and libraries used in RL
- Understand and work with tabular methods to solve classical control problems
- Understand and work with approximate solutions (deep Q network based algorithms)
- Learn the policy gradient methods from vanilla to more complex
- Explore imitation learning tasks and solutions
- Recognize current advanced techniques and applications in RL

**UNIT I**

**Foundations:** Introduction and Basics of RL, Defining RL Framework and Markov Decision Process, Polices, Value Functions and Bellman Equations, Exploration vs. Exploitation, Code Standards and Libraries used in RL (Python/Keras/Tensorflow)

**UNIT II**

**Tabular methods and Q-networks:** Planning through the use of Dynamic Programming and Monte Carlo, Temporal-Difference learning methods (TD(0), SARSA, Q-Learning), Deep Q-networks (DQN, DDQN, Dueling DQN, Prioritised Experience Replay)

**UNIT III**

**Policy optimization:** Introduction to policy-based methods, Vanilla Policy Gradient, REINFORCE algorithm and stochastic policy search, Actor-critic methods (A2C, A3C), Advanced policy gradient (PPO, TRPO, DDPG)

**UNIT IV**

**Model based RL:** Model-based RL approach

**UNIT V**

**Recent Advances and Applications:** Meta-learning, Multi-Agent Reinforcement Learning, Partially Observable Markov Decision Process, Ethics in RL, Applying RL for real-world problems



**TEXT BOOKS:**

1. Richard S. Sutton and Andrew G. Barto, "Reinforcement learning: An introduction", Second Edition, MIT Press, 2019
2. Li, Yuxi. "Deep reinforcement learning." arXiv preprint arXiv:1810.06339 (2018).
3. Wiering, Marco, and Martijn Van Otterlo. "Reinforcement learning." Adaptation, learning, and optimization 12 (2012):
4. Russell, Stuart J., and Peter Norvig. "Artificial intelligence: a modern approach." Pearson Education Limited, 2016.
5. Goodfellow, Ian, Yoshua Bengio, and Aaron Courville. "Deep learning." MIT press, 2016.
6. David Silver's course on Reinforcement Learning ([link](#))

**SELECTED PAPERS:**

1. Mnih, Volodymyr, Koray Kavukcuoglu, David Silver, Andrei A. Rusu, Joel Veness, Marc G. Bellemare, Alex Graves et al. "Human-level control through deep reinforcement learning." Nature 518, no. 7540 (2015): 529.
2. Van Hasselt, Hado, Arthur Guez, and David Silver. "Deep reinforcement learning with double q-learning." In Thirtieth AAAI conference on artificial intelligence. 2016.
3. Wang, Ziyu, Tom Schaul, Matteo Hessel, Hado Van Hasselt, Marc Lanctot, and Nando De Freitas. "Dueling network architectures for deep reinforcement learning." arXiv preprint arXiv:1511.06581 (2015).
4. Schaul, Tom, John Quan, Ioannis Antonoglou, and David Silver. "Prioritized experience replay." arXiv preprint arXiv:1511.05952 (2015).
5. Bojarski, Mariusz, Davide Del Testa, Daniel Dworakowski, Bernhard Firner, Beat Flepp, Praseon Goyal, Lawrence D. Jackel et al. "End to end learning for self-driving cars." arXiv preprint arXiv:1604.07316 (2016).
6. Schulman, John, Sergey Levine, Pieter Abbeel, Michael Jordan, and Philipp Moritz. "Trust region policy optimization." In International conference on machine learning, pp. 1889-1897. 2015.
7. Schulman, John, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. "Proximal policy optimization algorithms." arXiv preprint arXiv:1707.06347 (2017).
8. Lillicrap, Timothy P., Jonathan J. Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. "Continuous control with deep reinforcement learning." arXiv preprint arXiv:1509.02971 (2015).

**MASTER OF TECHNOLOGY  
(DATA SCIENCES)  
II YEAR I SEMESTER**

**DS3E18**

**MULTIAGENT SYSTEMS  
(ELECTIVE-5)**

Objectives:

- To introduce the student to the concept of an agent and multi-agent systems, and the main applications for which they are appropriate;
- To introduce the main issues surrounding the design of intelligent agents;
- To introduce the main issues surrounding the design of a multi-agent society.
- To introduce a contemporary platform for implementing agents and multi-agent systems

**UNIT I**

**Introduction:** what is an agent? agents and objects, agents and expert systems, agents and distributed systems, typical application areas for agent systems.

**UNIT II**

**Intelligent Agents:** the design of intelligent agents - reasoning agents (eg AgentO), agents as reactive systems (eg subsumption architecture), hybrid agents (eg PRS), layered agents (eg Interrap) a contemporary (Java-based) framework for programming agents (eg the Jack language, the JAM! system).

**UNIT III**

**Multi-Agent Systems:** Classifying multi-agent interactions - cooperative versus non-cooperative, zero-sum and other interactions, what is cooperation? how cooperation occurs - the Prisoner's dilemma and Axelrod's experiments, Interactions between self-interested agents: auctions & voting systems: negotiation

**UNIT IV**

**Interactions between benevolent agents:** cooperative distributed problem solving (CDPS), partial global planning, coherence and coordination;

**UNIT V**

**Interaction languages and protocols:** speech acts, KQML/KIF, the FIPA framework

**TEXT BOOKS:**

1. An Introduction to MultiAgent Systems - Second Edition. Michael Wooldridge (Wiley, 2009)
2. Programming Multi-agent Systems in AgentSpeak Using Jason. Rafael H. Bordini, Jomi Fred Hubner and Michael Wooldridge (Wiley, 2007)

**MASTER OF TECHNOLOGY  
(DATA SCIENCES)  
II YEAR I SEMESTER**

**DS3011**

**BUSINESS ANALYTICS  
(OPEN ELECTIVE-1)**

**Course objective**

- Understand the role of business analytics within an organization.
- Analyze data using statistical and data mining techniques and understand relationships between the underlying business processes of an organization.
- To gain an understanding of how managers use business analytics to formulate and solve business problems and to support managerial decision making.
- To become familiar with processes needed to develop, report, and analyze business data.
- Use decision-making tools/Operations research techniques.
- Mange business process using analytical and management tools.
- Analyze and solve problems from different industries such as manufacturing, service, retail, software, banking and finance, sports, pharmaceutical, aerospace etc.

**UNIT I**

Business analytics: Overview of Business analytics, Scope of Business analytics, Business Analytics Process, Relationship of Business Analytics Process and organisation, competitive advantages of Business Analytics. Statistical Tools: Statistical Notation, Descriptive Statistical methods, Review of probability distribution and data modelling, sampling and estimation methods overview.

**UNIT II**

Trendiness and Regression Analysis: Modelling Relationships and Trends in Data, simple Linear Regression. Important Resources, Business Analytics Personnel, Data and models for Business analytics, problem solving, Visualizing and Exploring Data, Business Analytics Technology.

**UNIT III**

Organization Structures of Business analytics, Team management, Management Issues, Designing Information Policy, Outsourcing, Ensuring Data Quality, Measuring contribution of Business analytics, Managing Changes. Descriptive Analytics, predictive analytics, predicative Modelling, Predictive analytics analysis, Data Mining, Data Mining Methodologies, Prescriptive analytics and its step in the business analytics Process, Prescriptive Modeling, nonlinear ptimization.

**UNIT IV**

Forecasting Techniques: Qualitative and Judgmental Forecasting, Statistical Forecasting Models, Forecasting Models for Stationary Time Series, Forecasting Models for Time Series with a Linear Trend, Forecasting Time Series with Seasonality, Regression Forecasting with Casual Variables, Selecting Appropriate Forecasting Models. Monte Carlo Simulation and Risk Analysis: Monte Carle Simulation Using Analytic Solver Platform, New-Product Development Model, Newsvendor Model, Overbooking Model, Cash Budget Model.

**UNIT V**

Decision Analysis: Formulating Decision Problems, Decision Strategies with the without Outcome Probabilities, Decision Trees, The Value of Information, Utility and Decision Making.,Recent Trends in : Embedded and collaborative business intelligence, Visual data recovery, Data Storytelling and Data journalism.

**COURSE OUTCOMES**

1. Students will demonstrate knowledge of data analytics.
2. Students will demonstrate the ability of think critically in making decisions based on data and deep analytics.
3. Students will demonstrate the ability to use technical skills in predicative and prescriptive modeling to support business decision-making.
4. Students will demonstrate the ability to translate data into clear, actionable insights.

**REFERENCES:**

1. Business analytics Principles, Concepts, and Applications by Marc J. Schniederjans, Dara G. Schniederjans, Christopher M. Starkey, Pearson FT Press.
2. Business Analytics by James Evans, persons Education.

**MASTER OF TECHNOLOGY  
(DATA SCIENCES)  
II YEAR I SEMESTER**

**DS3012**

**INDUSTRIAL SAFETY  
(OPEN ELECTIVE-1)**

**UNIT I**

Industrial safety: Accident, causes, types, results and control, mechanical and electrical hazards, types, causes and preventive steps/procedure, describe salient points of factories act 1948 or health and safety, wash rooms, drinking water layouts, light, cleanliness, fire, guarding, pressure vessels, etc, Safety color codes. Fire prevention and firefighting, equipment and Methods.

**UNIT II**

Fundamentals of maintenance engineering: Definition and aim of maintenance engineering, Primary and secondary functions and responsibility of maintenance department, Types of maintenance, Types and applications of tools used for maintenance, Maintenance cost & its relation with replacement economy, Service life of equipment.

**UNIT III**

Wear and Corrosion and their prevention: Wear- types, causes, effects, wear reduction methods, lubricants-types and applications, Lubrication methods, general sketch, working and applications, i. Screw down grease cup, ii. Pressure grease gun, iii. Splash lubrication, iv. Gravity lubrication, v. Wick feed lubrication vi. Side feed lubrication, vii. Ring lubrication, Definition, principle and factors affecting the corrosion. Types of corrosion, corrosion prevention methods.

**UNIT IV**

Fault tracing: Fault tracing-concept and importance, decision tree concept, need and applications, sequence of fault finding activities, show as decision tree, draw decision tree for problems in machine tools, hydraulic, pneumatic, automotive, thermal and electrical equipment's like, I. Any one machine tool, ii. Pump iii. Air compressor, iv. Internal combustion engine, v. Boiler, vi. Electrical motors, Types of faults in machine tools and their general causes.

**UNIT V**

Periodic and preventive maintenance: Periodic inspection-concept and need, degreasing, cleaning and repairing schemes, overhauling of mechanical components, overhauling of electrical motor, common troubles and remedies of electric motor, repair complexities and its use, definition, need, steps and advantages of preventive maintenance. Steps/procedure for periodic and preventive maintenance of: I. Machine tools, ii. Pumps, iii. Air compressors, iv. Diesel generating (DG) sets, Program and schedule of preventive maintenance of mechanical and electrical equipment, advantages of preventive maintenance. Repair cycle concept and importance

**REFERENCES:**

1. Maintenance Engineering Handbook, Higgins & Morrow, Da Information Services.
2. Maintenance Engineering, H. P. Garg, S. Chand and Company.
3. Pump-hydraulic Compressors, Audels, Mcgrew Hill Publication.
4. Foundation Engineering Handbook, Winterkorn, Hans, Chapman & Hall London.

**MASTER OF TECHNOLOGY  
(DATA SCIENCES)  
II YEAR I SEMESTER**

**DS3013**

**OPERATIONS RESEARCH  
(OPEN ELECTIVE-1)**

**Course Outcomes:** At the end of the course, the student should be able to

1. Students should be able to apply the dynamic programming to solve problems of discrete and continuous variables.
2. Students should be able to apply the concept of non-linear programming
3. Students should be able to carry out sensitivity analysis
4. Student should be able to model the real world problem and simulate it.

**Syllabus Contents:**

**UNIT I**

Optimization Techniques, Model Formulation, models, General L.R Formulation, Simplex Techniques, Sensitivity Analysis, Inventory Control Models

**UNIT II**

Formulation of a LPP - Graphical solution revised simplex method - duality theory - dual simplex method - sensitivity analysis - parametric programming

**UNIT III**

Nonlinear programming problem - Kuhn-Tucker conditions min cost flow problem - max flow problem - CPM/PERT

**UNIT IV**

Model Curriculum of Engineering & Technology PG Courses [Volume -II] [ 31 ] Scheduling and sequencing - single server and multiple server models - deterministic inventory models - Probabilistic inventory control models - Geometric Programming.

**UNIT V**

Competitive Models, Single and Multi-channel Problems, Sequencing Models, Dynamic Programming, Flow in Networks, Elementary Graph Theory, Game Theory Simulation

**REFERENCES:**

1. H.A. Taha, Operations Research, An Introduction, PHI, 2008
2. H.M. Wagner, Principles of Operations Research, PHI, Delhi, 1982.
3. J.C. Pant, Introduction to Optimisation: Operations Research, Jain Brothers, Delhi, 2008
4. Hitler Libermann Operations Research: McGraw Hill Pub. 2009
5. Pannerselvam, Operations Research: Prentice Hall of India 2010
6. Harvey M Wagner, Principles of Operations Research: Prentice Hall of India 2010

**MASTER OF TECHNOLOGY  
(DATA SCIENCES)  
II YEAR I SEMESTER**

**DS3014**

**COST MANAGEMENT AND ENGINEERING PROJECTS  
(OPEN ELECTIVE-1)**

**Introduction and Overview of the Strategic Cost Management Process Cost concepts in decision-making;** Relevant cost, Differential cost, Incremental cost and Opportunity cost.

**Objectives of a Costing System;** Inventory valuation; Creation of a Database for operational control; Provision of data for Decision-Making.

**Project:** meaning, Different types, why to manage, cost overruns centres, various stages of project

**execution:** conception to commissioning. Project execution as conglomeration of technical and non technical activities. Detailed Engineering activities. Pre project execution main clearances and documents

**Project team:** Role of each member. Importance Project site: Data required with

significance. Project contracts. Types and contents. Project execution Project cost control. Bar charts and Network diagram.

**Project commissioning:** mechanical and process Cost Behavior and Profit Planning Marginal Costing; Distinction between Marginal Costing and Absorption Costing; Break-even Analysis, Cost-Volume-Profit Analysis. Various decision-making problems. Standard Costing and Variance Analysis. Pricing strategies: Pareto Analysis. Target costing, Life Cycle Costing. Costing of service sector. Just-in-time approach, Material Requirement Planning, Enterprise Resource Planning, Total Quality Management and Theory of constraints.

**Activity-Based Cost Management, Bench Marking;** Balanced Score Card and Value-Chain Analysis.

**Budgetary Control;** Flexible Budgets; Performance budgets; Zero-based budgets. Measurement of Divisional profitability pricing decisions including transfer pricing. Quantitative techniques for cost management, Linear Programming, PERT/CPM, Transportation problems, Assignment problems, Simulation, Learning Curve Theory.



**REFERENCES:**

1. Cost Accounting A Managerial Emphasis, Prentice Hall of India, New Delhi
2. Charles T. Horngren and George Foster, Advanced Management Accounting
3. Robert S Kaplan Anthony A. Alkinson, Management & Cost Accounting Model Curriculum of Engineering & Technology PG Courses [Volume -II][ 32 ]
4. Ashish K. Bhattacharya, Principles & Practices of Cost Accounting A. H. Wheeler publisher
5. N.D. Vohra, Quantitative Techniques in Management, Tata McGraw Hill Book Co. Ltd.

**MASTER OF TECHNOLOGY  
(DATA SCIENCES)  
II YEAR I SEMESTER**

**DS3015**

**CYBER SECURITY  
(OPEN ELECTIVE-1)**

**Objectives:**

- To learn about cyber crimes and how they are planned
- To learn the vulnerabilities of mobile and wireless devices
- To learn about the crimes in mobile and wireless devices

**UNIT I**

**Introduction to Cybercrime:** Introduction, Cybercrime and Information security, who are cybercriminals, Classifications of Cybercrimes, Cybercrime: The legal Perspectives and Indian Perspective, Cybercrime and the Indian ITA 2000, A Global Perspective on Cybercrimes.

**UNIT II**

**Cyber offenses: How criminals Plan Them** Introduction, How Criminals plan the Attacks, Social Engineering, Cyber stalking, Cyber cafe and Cybercrimes, Botnets: The Fuel for Cybercrime, Attack Vector, Cloud Computing.

**UNIT III**

**Cybercrime: Mobile and Wireless Devices:** Introduction, Proliferation of Mobile and Wireless Devices, Trends in Mobility, Credit card Frauds in Mobile and Wireless Computing Era, Security Challenges Posed by Mobile Devices, Registry Settings for Mobile Devices, Authentication service Security, Attacks on Mobile/Cell Phones, Mobile Devices: Security Implications for Organizations, Organizational Measures for Handling Mobile.

**UNIT IV**

**Tools and Methods Used in Cybercrime: Introduction,** Proxy Servers and Anonymizers, Phishing, Passwaord Cracking, Keyloggers and Spywares, Virus and Worms, Trojan Horses and Backdoors, Steganography, DoS and DDoS Attacks, SQL Injection, Buffer Overflow, Attacks on Wireless Networks

**UNIT V**

**Understanding Computer Forensics** Introduction, Historical background of Cyber forensics, Digital Forensics Science, The Need for Computer Forensics, Cyber Forensics and Digital evidence, Forensics Analysis of Email, Digital Forensics Lifecycle, Chain of Custody concept, Network Forensics, Approaching a computer, Forensics Investigation, Challenges in Computer Forensics, Special Tools and Techniques,Forensics Auditing

**TEXT BOOKS:**

1. **Cyber Security:** *Understanding Cyber Crimes, Computer Forensics and Legal Perspectives*, Nina Godbole and Sunil Belapure, Wiley INDIA.
2. **Introduction to Cyber Security** , Chwan-Hwa(john) Wu,J.David Irwin.CRC Press T&F Group

**REFERENCES:**

1. **Cyber Security Essentials**, James Graham, Richard Howard and Ryan Otson, CRC Press.

**MASTER OF TECHNOLOGY  
(DATA SCIENCES)  
II YEAR I SEMESTER**

**DS3016**

**COMPOSITE MATERIALS  
(OPEN ELECTIVE-1)**

**UNIT I**

**INTRODUCTION:** Definition – Classification and characteristics of Composite materials. Advantages and application of composites. Functional requirements of reinforcement and matrix Effect of reinforcement (size, shape, distribution, volume fraction) on overall composite performance.

**UNIT II**

**REINFORCEMENTS:** Preparation-layup, curing, properties and applications of glass fibers, carbon fibers, Kevlar fibers and Boron fibers. Properties and applications of whiskers, particle reinforcements. Mechanical Behavior of composites: Rule of mixtures, Inverse rule of mixtures. Isostrain and Isostress conditions.

**UNIT III**

**Manufacturing of Metal Matrix Composites:** Casting – Solid State diffusion technique, Cladding – Hot isostatic pressing. Properties and applications. Manufacturing of Ceramic Matrix Composites: Liquid Metal Infiltration – Liquid phase sintering. Manufacturing of Carbon – Carbon composites: Knitting, Braiding, Weaving. Properties and applications.

**UNIT IV**

**Manufacturing of Polymer Matrix Composites:** Preparation of Moulding compounds and prepregs – hand layup method – Autoclave method – Filament winding method – Compression moulding – Reaction injection moulding. Properties and applications.

**UNIT V**

**Strength:** Lamina Failure Criteria-strength ratio, maximum stress criteria, maximum strain criteria, interacting failure criteria, hygrothermal failure. Laminate first ply failure-insight strength; Laminate strength-ply discount truncated maximum strain criterion; strength design using caplet plots; stress concentrations.

**TEXT BOOKS:**

1. Material Science and Technology – Vol 13 – Composites by R.W.Cahn – VCH, West Germany.
2. Materials Science and Engineering, An introduction. WD Callister, Jr., Adapted by R. Balasubramaniam, John Wiley & Sons, NY, Indian edition, 2007.

**REFERENCES:**

1. Hand Book of Composite Materials-ed-Lubin.
2. Composite Materials – K.K.Chawla.
3. Composite Materials Science and Applications – Deborah D.L. Chung.
4. Composite Materials Design and Applications – Danial Gay, Suong V. Hoa, and Stephen W. Tasi.