DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

ACADEMIC REGULATIONS COURSE STRUCTURE AND DETAILED SYLLABUS

FOR

M.TECH TWO YEAR DEGREE COURSE

IN COMPUTER SCIENCE AND ENGINEERING

M.Tech(CSE)

(Applicable for the batches admitted from 2025-26)

G. NARAYANAMMA INSTITUTE OF TECHNOLOGY & SCIENCE

AUTONOMOUS) (For Women)

Shaikpet, Hyderabad –500104.

G.NARAYANAMMA INSTITUTE OF TECHNOLOGY & SCIENCE
AUTONOMOUS (for Women)
Shaikpet, Hyderabad- 500 104

Academic Regulations of M.Tech (Regular/Full Time) Programmes, 2025-26 (R-25)
(Effective for the students admitted from the Academic Year 2025-26 and on wards)

1.0 G.Narayanamma Institute of Technology and Science for women Hyderabad(GNITS) offers **Two** Years (**Four** Semesters) full-time Master of Technology (M.Tech.) Degree programmes, under Choice Based Credit System (CBCS) at its non-autonomous affiliated colleges in different branches of Engineering and Technology with specializations as listed below.

S.No.	***Branch/ Department***	***Specialization***
I.	Computer Science & Engineering	Computer Science & Engineering
II.	Electrical & Electronics Engineering	Power Electronics & Electric Drives
III.	Electronics & Communication Engineering	Digital Electronics & Communication Engineering
IV.	Electronics & Telematics Engineering	Wireless & Mobile Communications
V.	Information Technology	Computer Networks & Information Security

2.0 Eligibility for Admissions

2.1 Admission to the M.Tech. programme shall be made subject to eligibility, qualification and specializations prescribed by the University from timeto time, for each specialization under each M.Tech. programme.

2.2 Admission to the post graduate programme shall be made on the basis of either the merit rank or Percentile obtained by the qualified student in the GATE Examination/ the merit rank obtained by the qualified student in an entrance test conducted by Telangana Government (PGECET) for M.Tech. programmes/ an entrance test conducted by GNITS/ on the basis of any other exams approved by the TS Govt., subject to reservations as laid down by the Govt. from time to time.

2.3 The medium of instructions for all PG Programmes will be **ENGLISH** only.

3.0 M.Tech. Programme Structure

3.1 The M.Tech. Programs of GNITS are of Semester pattern, consisting of **Two** academic years, each academic year having **Two** Semesters (Odd and Even Semesters).

3.2 The two-year M. Tech. program consists of **68** credits and the student has to register for all **68** credits and earn all **68** credits for the award of M. Tech. degree.

3.3 The student shall not take more than four academic years to fulfill all the academic requirements for the award of M. Tech. degree from the date of commencement of first year first semester, failing which the student shall forfeit the seat in M. Tech. programme.

3.4 **UGC/AICTE** specified definitions/descriptions are adopted appropriately for various terms and abbreviations used in these PG academic regulations, as listed below:

3.4.1 Semester Scheme

There shall be a minimum of 15 weeks of instruction, excluding the mid-term and semester-end exams. Around 15 instruction hours, 30 instruction hours and 45 hours of learning need to be followed per one credit of theory course, practical course and project/field-based learning respectively. I0n each semester, there shall be 'Continuous Internal Evaluation (CIE)' and 'Semester End Examination (SEE)' under Choice Based Credit System (CBCS). The curriculum/course structure suggested by AICTE is followed as a reference document.

3.4.2 Credit Courses

All courses are to be registered by the student in a semester to earn credits which shall be assigned to each course in an L: T: P: C (Lecture Periods: Tutorial Periods: Practical Periods: Credits) structure based on the following general pattern:

- One credit for one hour/week/semester for theory/lecture(L)courses or tutorials(T)
- One credit for two hours/week/semester for laboratory/practical(P)courses
- One credit is allocated for three hours per week in a semester for Project/Mini-Project session.

3.4.3 Course Classification

All courses offered for the Post-Graduate M.Tech. Degree program are broadly classified as follows. The University has followed in general the guidelines issued by AICTE/UGC.

S. No.	Broad Course Classification	Course Group/ Category	Course Description
1	Core Courses (CoC)	PC- Professional Core	Includes courses related to the parent discipline/department/ branch of Engineering
		Dissertation	M.Tech. Projector PG Projector Major Project
		Mini Project with Seminar	Seminar based on core contents related to Parent Discipline/Department/Branch of Engineering
2	Elective Courses (EtE)	PE - Professional Electives	Includes elective courses related to the parent discipline/department/branch of Engineering
		OE - Open Electives	Elective courses which include inter-disciplinary courses or courses in an area outside the parent discipline/department/branch of Engineering
3	Audit Courses	--	Non-Credit Audit Courses

4.0 Course Registration

4.1 A Faculty Advisor or Counselor shall be assigned to each specialization, who will advise on the Post Graduate Programme, its Course Structure and Curriculum, Choices/Options for Courses, based on his competence, progress, pre-requisites and interest.

4.2 The on-line Registration Requests for any current semester shall be completed before the commencement of SEEs (Semester End Examinations) of the preceding semester.

4.3 A Student can apply for on-line Registration, only after obtaining the written approval from his Faculty Advisor, which should be submitted to the College Academic Section through the Head of Department (a copy of it being retained with Head of Department, Faculty Advisor and the Student).

4.4 If the Student submits ambiguous choices or multiple options or erroneous entries during on-line Registration for the Course(s) under a given/ specified Course Group/ Category as listed in the Course Structure, only the first mentioned Course in that Category will be taken into consideration.

4.5 Course Options exercised through on-line Registration are final and cannot be changed. further, alternate choices will not be considered. However, if the Course that has already been listed for Registration by the University in a Semester could not be offered due to unforeseen or unexpected reasons, then the Student will be allowed to have alternate choice either for a new Course, if it is offered, or for another existing Course (subject to availability of seats). Such alternate arrangements will be made by the Head of Department, with due notification and time-framed schedule, within the first week from the commencement of Class-work for that Semester.

5.0 Attendance Requirements

Attendance is calculated separately for each course.

5.1 Attendance in all classes (Lectures/Laboratories) is compulsory. The minimum required attendance in each theory course (***also Audit Courses***) including the attendance of mid-term examination / Laboratory etc. is 75%. Two periods of attendance for each theory course shall be considered, if the student appears for the mid-term examination of that course. ***This attendance should also be included in the attendance uploaded every fortnight in the University Website. The attendance of Audit Courses should be uploaded separately to the University.*** A student shall not be permitted to appear for the Semester End Examinations (SEE), if his attendance is less than 75%.

5.2 A student's Seminar report and presentation on Mini Project shall be eligible for evaluation, only if he ensures a minimum of 75%of his attendance in Seminar presentation classes on Mini Project during that Semester.

5.3 **Condoning of shortage of attendance** up to a maximum of 10% (considering the days of attendance in sports, games, NCC, NSS activities and Medical grounds) in each course (Theory/Lab/Mini Project with Seminar) of a semester shall be granted by the College

Academic Committee on genuine reasons.

5.4 A prescribed fee per course shall be payable for condoning shortage of attendance after getting the approval of College Academic Committee for the same. The College Academic Committee shall maintain relevant documents along with the request from the student.

5.5 Shortage of Attendance below 65% in any course shall in **no case be condoned.**

5.6 A Student, whose shortage of attendance is not condoned in any course(s) (Theory/Lab/Mini Project with Seminar) in any Semester, is considered as 'Detained in that course(s), and is not eligible to write Semester End Examination(s) of such course(s), in that Semester; and he/she has to seek re- registration for those course(s) in subsequent Semesters, and attend the same as and when offered.

5.7 A student fulfills the attendance requirement in the present semester, shall not be eligible for readmission into the same class.

5.8 **a)** A student shall put in a minimum required attendance in at least three theory courses (excluding Audit course) in first Year I semester for promotion to first Year II Semester.

b) A student shall put in a minimum required attendance in at least three theory courses (excluding *Audit* course**)** in first Year II semester for promotion to second Year I Semester.

6.0 Academic Requirements

The following academic requirements must be satisfied, in addition to the attendance requirements mentioned in clause no. 5. The performance of the candidate in each semester shall be evaluated course-wise, with a maximum of 100 marks per course (theory/ practical), based on Continuous Internal Evaluation and Semester End Examination.

6.1 A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each course, if he secures not less than:

- 40% of Marks(24 out of 60marks) in the Semester End Examination;
- 40% of Marks in the internal examinations(16 out of 40 marks allotted for CIE);and
- A minimum of 50% of marks in the sum total of CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together; in terms of Letter Grades this implies securing **'B'** Grade or above in a course.

6.2 A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to Mini Project with seminar, if student secures not less than 50% marks (i.e. 50 out of 100 allotted marks). The student would be treated as failed, if student (i) does not submit a seminar report on Mini Project or does not make a presentation of the same before the evaluation committee as per schedule or (ii) secures less than 50% marks in Mini Project with seminar evaluation. The failed student shall reappear for the above evaluation when the notification for supplementary examination is issued.

6.3 A student shall register for all courses for total of **68** credits as specified and listed in the course structure for the chosen specialization, put in the required attendance and fulfill the academic requirements for securing **68** credits obtaining a minimum of **'B'** Grade or above in each course, and shall ***pass all the Audit Courses*** to complete the M.Tech. Programme successfully.

Note: (1) The SGPA will be computed and printed on the marks memo only if the candidate passes in all the courses offered and gets minimum B grade in all the courses.

(2) CGPA is calculated only when the candidate passes in all the courses offered in all the semesters

6.4 Letter Grades obtained in all those courses covering the above specified **68** credits alone shall be considered for the calculation of final CGPA, which will be indicated in the consolidated grade memo.

6.5 When a student is detained due to shortage of attendance in any course(s) in any semester, shall not be permitted to write the Semester End Examinations. However, he is eligible for re-registration of such course(s) in the subsequent semester(s), as and when next offered, with the academic regulations of the batch in to which he is re-registered, by paying the prescribed fees per course. In all these re-registration cases, the student shall have to secure a fresh set of internal marks and Semester End Examination marks for performance evaluation in such course(s), and SGPA/CGPA calculations.

6.6 A student eligible to appear for the Semester End Examination in any course, but absent from it or failed (failing to secure 'B' Grade or above), may reappear for that course at the supplementary examination as and when conducted. In such cases, his Internal Marks assessed earlier for that course will be carried over, and added to the marks secured in the supplementary semester end examination, for the purpose of evaluating his performance in that course.

6.7 A Student who fails to earn **68** credits as per the specified course structure, and as indicated above, within **four** academic years from the date of commencement of his first year first semester, shall forfeit his seat in M. Tech. programme and his admission **shall stand cancelled.**

7.0 Evaluation-Distribution and Weightage of Marks

The performance of a student in each semester shall be evaluated course- wise (irrespective of credits assigned) for a maximum of 100 marks.

7.1 The performance of a student in every course (including practicals and Project) will be evaluated for100 marks each, with 40 marks allotted for CIE (Continuous Internal Evaluation) and 60 marks for SEE (Semester End-Examination). The Continuous Internal Evaluation for theory courses shall be made based on the average of the marks secured in the two Mid-Term Examinations conducted, first Mid- Term examinations in the middle of the Semester and second Mid-Term examinations during the last week of instruction.

7.2 In CIE, for theory courses, during a semester, there shall be two mid-term examinations. Each Mid-Term examination consists of two parts i) **Part – A**for 10 marks, ii) **Part – B** for 20 marks with a total duration of two hours as follows:

1. Mid-Term Examination for 30 marks:
   a. Part-A: Objective/quiz paper for 10 marks.
   b. Part–B: Descriptive paper for 20 marks.

The objective/quiz paper is set with multiple choice, fill-in the blanks and match the following type of questions for a total of 10 marks. The descriptive paper shall contain 6 full questions out of which, the student has to answer 4 questions, each carrying 5 marks. The average of the two Mid Term Examinations shall be taken as the final marks for Mid Term Examination for 30 marks.

The remaining 10 marks of Continuous Internal Assessment (outof40) are distributed as:

2. 5 marks for Assignment. (Average of 2 Assignments each for 5 marks)
3. Course Viva-Voce/PPT/Poster Presentation/ Case Study on a topic in the concerned course for 5 marks.

While the first mid-term examination shall be conducted on 50% of the syllabus, the second mid-term examination shall be conducted on the remaining 50% of the syllabus.

Five (5) marks are allocated for assignments (as specified by the course teacher concerned). The first assignment should be submitted before the conduct of the first mid-term examination, and the second assignment should be submitted before the conduct of the second mid-term examination. The average of the two assignments shall be taken as the final marks for assignment.

Course Viva-Voce/PPT/Poster Presentation/ Case Study on a topic in the concerned course for five marks before II Mid-Term Examination.

- The Student, in each course, shall have to earn 40% of marks (i.e. 16 marks out of 40 marks) in CIE, 40% of marks (i.e. 24 marks out of 60) in SEE and Overall 50% of marks (i.e. 50 marks out of 100 marks) both CIE and SEE marks taking together.

The student is eligible to write Semester End Examination of the concerned course, if the student scores

≥40% (16 marks) of 40 Continuous Internal Examination (CIE) marks.

In case, the student appears for Semester End Examination (SEE) of the concerned course but not scored minimum 40% of CIE marks (16 marks out of 40 internal marks), his performance in that course in SEE shall stand cancelled in spite of appearing the SEE.

The details of the end semester question paper pattern are explained in the next clause:

7.3 The Semester End Examinations(SEE),for theory courses, will be conducted for 60 marks consisting of two parts viz. i) **Part- A** for 10 marks, ii) **Part - B** for 50 marks.

- Part-A is a compulsory question which consists of ten sub-questions with uniform coverage from all units carrying equal marks.
- Part-B consists of five questions (numbered from 2 to 6) carrying 10 marks each. Each of these questions is from each unit and may contain sub-questions. For each question there will be an "either" "or" choice, which means that there will be two questions from each unit and the student should answer either of the two questions.
- The duration of Semester End Examination is three hours.

7.4 For practical courses there shall be a Continuous Internal Evaluation (CIE) during the semester for 40 marks and 60 marks for semester end examination. Out of the 40 marks for internal evaluation:

1. A write-up on day-to-day experiment in the laboratory (in terms of aim, components/procedure, expected outcome) which shall be evaluated for 10 marks
2. 10 marks for viva-voce(or)tutorial(or)case study(or) application(or)poster presentation of the course concerned.
3. Internal practical examination conducted by the laboratory teacher concerned shall be evaluated for 10 marks.
4. The remaining 10 marks are for Laboratory Project, which consists of the Design(or)Software/ Hardware Model Presentation (or) App Development (or) Prototype Presentation submission which Shall be evaluated after completion of laboratory course and before semester end practical examination.

In the Semester End Examination, held for three hours, total 60 marks are divided and allocated as shown below:

1. 10 marks for write-up
2. 15 for experiment/program
3. 15 for evaluation of results
4. 10 marks for presentation on another experiment/program in the same laboratory course and
5. 10 marks for viva-voce on concerned laboratory course.

- The Student, in each course, shall have to earn 40% of marks (i.e. 16 marks out of 40 marks) in CIE, 40% of marks (i.e. 24 marks out of 60) in SEE and Overall 50% of marks (i.e. 50 marks out of 100 marks) both CIE and SEE marks taking together.

The student is eligible to write Semester End Examination of the concerned course, if the student scores

≥40% (16 marks) of 40 Continuous Internal Examination(CIE) marks.

In case, the student appears for Semester End Examination (SEE) of the concerned course but not scored minimum 40% of CIE marks (16 marks out of 40 internal marks), his performance in that course in SEE shall stand cancelled in spite of appearing the SEE.

7.5 For conducting laboratory end examinations of all PG Programmes, one internal examiner and one external examiner are to be appointed by the Principal. The external examiner should be selected from outside the College.

7.6 There shall be Mini Project with Seminar during I year II semester for internal evaluation of 100 marks. The Departmental Academic Committee (DAC) will review the progress of the mini project during the seminar presentations and evaluate the same for 50 marks. Mini Project Viva Voce will be evaluated by the DAC for another 50 marks before the semester end examinations. Student shall carry out the mini project in consultation with the mini project supervisor which may include critically reviewing the literature, project implementation and submit it to the department in the form of a report and shall make an oral presentation before the DAC consisting of Head of the Department, Mini Project supervisor and two other senior faculty members of the department. The student has to secure a minimum of 50% of marks in i) seminar presentation and ii) mini project viva voce, to be declared successful. If he fails to obtain the minimum marks, he has to reappear for the same as and when scheduled.

7.7 Every candidate shall be required to submit a dissertation on a topic approved by the Dissertation Review Committee.

7.8 The M.Tech. Dissertation shall be prepared in the structure prescribed by the College, adhering to the style files and formatting guidelines. To facilitate this process, each Department will organize a brief orientation session for the entire class/section at the beginning of the final semester, guiding the students on the required structure and formatting of the dissertation.

7.9 A Dissertation Review Committee (DRC) shall be constituted with the Head of the Department as Chairperson, Dissertation Supervisor and one senior faculty member of the Department offering the M.Tech. programme.

7.10 Registration of Dissertation Work: A candidate is permitted to register for the Dissertation Work after satisfying the attendance requirement in all the courses, both theory and laboratory.

7.11 After satisfying the previous clause, a candidate must present in ***Dissertation Work Review-I***, in consultation with his Dissertation Supervisor, the title, objective and plan of action of his Dissertation work to the Dissertation Review Committee (DRC) for approval ***within four weeks*** from the commencement of **Second year First Semester**. Only after obtaining the approval of the DRC can the student initiate the Dissertation work.

7.12 If a candidate wishes to change his supervisor or topic of the Dissertation, he can do so with the approval of the DRC. However, the DRC shall examine whether or not the change of topic/supervisor leads to a major change of her initial plans of Dissertation proposal. If yes, her date of registration for the project work starts from the date of change of Supervisor or topic as the case may be.

7.13 A candidate shall submit his Dissertation progress report in two stages at least with a gap of **three** Months between them.

7.14 The work on the Dissertation shall be initiated at the beginning of the II year and the duration of the Dissertation is two semesters. A candidate is permitted to submit Dissertation Thesis only after successful completion of all theory and practical courses with the approval of DRC ***not earlier than 40 weeks*** from the date of approval of the Dissertation work. For the approval of DRC, the candidate shall submit the draft copy of thesis to the Head of the Department and make an oral presentation before the DRC.

7.15 ***The Dissertation Work Review - II*** in II Year I Semester carries 100 internal marks. Evaluation should be done by the DRC for 50 marks and the Supervisor will evaluate the work for the other 50marks. The Supervisor and DRC will examine the Problem Definition, Objectives, Scope of Work, Literature Survey in the same domain and progress of the Dissertation Work. A candidate has to secure a minimum of 50% of marks to be declared successful in Dissertation Work Review - II. If he fails to obtain the minimum required marks, he has to reappear for Dissertation Work Review - II as and when conducted.

7.16 ***The Dissertation Work Review - III*** in II Year II Sem. carries 100 internal marks. Evaluation should be done by the DRC for 50 marks and the Supervisor will evaluate it for the other 50 marks. The DRC will examine the overall progress of the Dissertation Work and decide whether or not the Dissertation is eligible for final submission. A candidate has to secure a minimum of 50% of marks to be declared successful in Dissertation Work Review-III. If she fails to obtain the required minimum marks, she has to re-appear for Dissertation Work Review- III as and when conducted. For Dissertation Evaluation (Viva Voce) in II Year II Semester there are external marks of 100 and it is evaluated by the external examiner. The candidate has to secure a minimum of 50% marks in Dissertation Evaluation (Viva-Voce) examination.

7.17 Dissertation Work Reviews - II and III shall be conducted in phase I (Regular) and Phase II (Supplementary). Phase II will be conducted only for unsuccessful students in Phase I. The unsuccessful students in Dissertation Work Review - II (Phase II) shall reappear for it at the time of Dissertation Work Review-III(PhaseI). These students shall re-appear for Dissertation Work Review - III in the next academic year at the time of Dissertation Work Review - II only after completion of Dissertation Work Review - II, and then Dissertation Work Review - III follows. The unsuccessful students in Dissertation Work Review-III(PhaseII) shall re-appear for Dissertation Work Review-III in the next academic year only at the time of Dissertation Work Review - II (Phase I).

7.18 After approval from the DRC,a soft copy of the thesis should be submitted for Anti-Plagiarism check and the plagiarism report should be submitted to the University and be included in the final thesis. The Thesis will be accepted for submission, if the similarity

index is less than **30%**.If the similarity index is more than the required percentage, the student is advised to revise the thesis and re-submit the soft copy of the thesis after one month. The maximum number of re-submissions of thesis after plagiarism check is limited to ***TWO***. The candidate has to register for the Dissertation work and work for two semesters. After three attempts, the admission is liable to be cancelled.

7.19 Three copies of the Dissertation Thesis certified by the supervisor shall be submitted to the exam section.

7.20 The thesis shall be adjudicated by an external examiner selected by the Principal. For this, the HOD of the department shall submit a panel of **three** examiners from among the list of experts in the relevant specialization as submitted by the supervisor concerned.

7.21 If the report of the external examiner is unsatisfactory, the candidate shall revise and resubmit the Thesis. If the report of the examiner is unsatisfactory again, the thesis shall be summarily rejected. Subsequent actions for such dissertations may be considered, only on the specific recommendations of the external examiner and /or Dissertation Review Committee. No further correspondence in this matter will be entertained, if there is no specific recommendation for resubmission.

7.22 If the report of the examiner is satisfactory, the Head of the Department shall coordinate and decide for the conduct of Dissertation Viva-Voce examination. The Dissertation Viva-Voce examination shall be conducted by a board consisting of the Supervisor, Head of the Department and the external examiner who adjudicated the Thesis. The candidate has to secure a minimum of 50% of marks in Dissertation Evaluation (Viva-Voce) examination.

7.23 If he fails to fulfill the requirements as specified in the above clause, he will reappear for the Dissertation Viva-Voce examination ***only after three months***. In the reappeared examination also, if he fails to fulfill the requirements, he will not be eligible for award of the degree, unless he is asked to revise and resubmit his Dissertation Work by the board within a specified time period.

7.24 The Dissertation Viva-Voce External examination marks must be submitted to the examinations section on the same day of the examination.

7.25 For Audit courses, a student has to secure 40 marks out of 100 marks (i.e. 40% of the marks allotted) in the continuous internal evaluation for passing the course. These marks should also be uploaded along with the internal marks of other courses.

7.26 No marks or letter grades shall be allotted for Audit Courses. Only Pass/Fail shall be indicated in Grade Card.

8 Re-Admission/Re-Registration

8.1 Re-Admission for Discontinued Student

A student, who has discontinued the M. Tech. degree programme due to any reason

whatsoever, may be considered for '**readmission**' into the same degree programme (with the same specialization) with the academic regulations of the batch into which he gets readmitted, with prior permission from the authorities concerned.

8.2 If a student is detained in a course (s) due to shortage of attendance in any semester, he may be permitted to **re-register** for the same course(s) or the student may register in an equivalent course, If the same course is not available, as suggested by the Board of Studies of that department, in the subsequent semester(s), with the academic regulations of the batch into which he seeks re-registration, with prior permission from the authorities concerned.

8.3 A candidate shall be given chance to re-register any number of courses, if the candidate failed in these courses due to securing less than 40% marks in CIE. A candidate must re-register for failed courses within four weeks of commencement of the class work, in the next academic year and secure the required minimum attendance. In the event of the student taking this chance, his Continuous Internal Evaluation (internal) marks and Semester End Examination marks obtained in the current semester only will be accepted, if he secures pass grade.

9 Examinations and Assessment- The Grading System

9.1 Grades will be awarded to indicate the performance of each student in each Theory Course, or Lab/Practicals, or Mini Project with Seminar, Dissertation, etc., based on the percentage of marks obtained in CIE + SEE (Continuous Internal Evaluation + Semester End Examination, both taken together), and a corresponding Letter Grade shall be given.

9.2 As a measure of the student's performance, a 10-point Absolute Grading System using the following Letter Grades (UGC Guidelines) and corresponding percentage of marks shall be followed:

% of Marks Secured in a Course(Class Intervals)	**Letter Grade(UGC Guidelines)**	**Grade Points**
90% and above (>= 90%, ≤100%)	O(Outstanding)	10
Below 90% but not less than 80% (>= 80%, <90%)	A^+(Excellent)	9
Below 80% but not less than 70% (>= 70%, < 80%)	A(Very Good)	8
Below70% but not less than 60% (>= 60%, < 70%)	B^+(Good)	7
Below 60% but not less than 50% (>= 50%, < 60%)	B(above Average)	6
Below 50% (< 50%)	F(FAIL)	0
Absent	**Ab**	**0**

9.3 A student obtaining **'F'** Grade in any Course is deemed to have 'failed' and is required to reappear as 'Supplementary Candidate' for the Semester End Examination (SEE), as and when conducted. In such cases, his Internal Marks (CIE Marks) in those courses will remain as obtained earlier.

9.4 If a student has not appeared for the examinations, '**Ab**' Grade will be allocated to her for

any course and shall be considered 'failed' and will be required to reappear as 'Supplementary Candidate' for the Semester End Examination (SEE), as and when conducted.

9.5 A Letter Grade does not imply any specific marks percentage; it is only the range of percentage of marks.

9.6 In general, a student shall not be permitted to repeat any Course (s) only for the sake of 'Grade Improvement' or 'SGPA/ CGPA Improvement'.

9.7 A student earns Grade Point (GP) in each Course, on the basis of the Letter Grade obtained by him in that Course. The corresponding 'Credit Points' (CP) are computed by multiplying the Grade Point with Credits for that particular Course.

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

9.8 The student passes the Course only when he **gets GP >= 6 (B Grade or above).**

9.9 The Semester Grade Point Average (SGPA) is calculated by dividing the Sum of Credit Points secured from all Courses registered in a Semester, by the total number of credits offered in that Semester. SGPA is rounded off to two decimal places. SGPA is thus computed as

$$\text{SGPA}=\{\sum_{i=1}^{N} C_i G_i\}/\{\sum_{i=1}^{N} C_i\}\ldots\text{.ForeachSemester}$$

where 'i' is the Course indicator index (taking into account all Courses in a Semester), 'N' is the no. of Courses offered in the Semester (as specifically required and listed under the Course Structure of the parent Department), C_i is the no. of Credits allotted to the i^{th} Course, and G_i represents the Grade Points corresponding to the Letter Grade awarded for that i^{th} Course.

9.10 The Cumulative Grade Point Average (CGPA) is a measure of the overall cumulative performance of a student over all Semesters considered for registration. The CGPA is the ratio of the Total Credit Points secured by a student in all registered Courses in all Semesters, and the Total Number of Credits registered in all the Semesters. CGPA is rounded off to two decimal places. CGPA is thus computed from the I Year Second Semester onwards, at the end of each Semester, as per the formula

$$\text{CGPA}=\{\sum_{j=1}^{M} C_j G_j\}/\{\sum_{j=1}^{M} C_j\}$$

(ie.,up to and inclusive of S Semesters,S□2),

Where 'M' is the total no. of Courses(as specifically required and listed under the Course Structure of the parent Department) the Student has 'registered'. C_j is the no. of Credits allotted to the j^{th} Course, and G_j represents the Grade Points (GP) corresponding to the Letter Grade awarded for that j^{th} Course. After registration and completion of I Year I Semester however ,the SGPA of that Semester itself may be taken as the CGPA, as there are no cumulative effects.

Illustration of calculation of SGPA

Course	Credits	Letter Grade	Grade points	Credit Points
Course1	4	A	8	4*8=32
Course2	4	O	10	4*10=40
Course3	4	B	6	4*6=24
Course4	3	B	6	3*6=18
Course5	3	A+	9	3*9=27
Course6	3	B	6	3*6=18
	21			159

SGPA=159/21=7.57

Illustration of calculation of CGPA from SGPA

Semester	Credits	SGPA	Credits*SGPA
Semester I	24	7	24*7=168
Semester II	24	6	24*6=144
Semester III	24	6.5	24*6.5=156
Semester IV	24	6	24*6=144
	96		612

CGPA=612/96=6.37

10.0 Award of Degree and Class

10.1 If a student who registers for all the specified Courses as listed in the Course Structure, satisfies all the Course Requirements, and passes the examinations prescribed in the entire PG Programme, and secures the required number of **68** Credits (with CGPA 6.0), shall be declared to have 'qualified' for the award of the M.Tech. Degree in the chosen Branch of Engineering/Technology with the specialization that he was admitted into.

10.2 Award of Class

After a student has earned the requirements prescribed for the completion of the programme 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	≥7.50
First Class	6.50 ≤ CGPA < 7.50
Second Class	6.00 ≤ CGPA < 6.50

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

11.0 Withholding of Results

If the student has not paid the dues, if any, to the University or if any case of indiscipline is pending against him, the result and degree of the student will be withheld and he will not be allowed into the next semester.

12.0 Conversion of CGPA into equivalent Percentage of Marks

The following formula shall be used for the conversion of CGPA into equivalent marks, whenever it is necessary

Percentage (%) of Marks= (FinalCGPA–0.5)x10

13.0 Mapping with the Sustainable Development Goals

All the courses specified in the course structure of every programme are mapped with the one or more sustainable development goals.

14.0 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 The academic regulations should be read as a whole for the purpose of any interpretation.

14.4 In case of any doubt or ambiguity in the interpretation of the above rules, the decision of the Princiapl is final.

14.5 The College may change or amend the academic regulations or syllabi at any time and the changes or amendments made shall be applicable to all the students with effect from the dates notified by the Principal.

MALPRACTICES RULES

DISCIPLINARY ACTION FOR / IMPROPER CONDUCT IN EXAMINATIONS

Nature of Malpractices/Improper conduct		Punishment
If the student:		
1. (a)	Possesses or keeps accessible in examination hall, any paper, note book, programmable calculators, cell phones, pager, palm computers or any other form of material concerned with or related to the subject of the examination (theory or practical) in which student is appearing but has not made use of (material shall include any marks on the body of the student 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 student orally or by any other body language methods or communicates through cell phones with any student or persons in or outside the exam hall in respect of any matter.	Expulsion from the examination hall and cancellation of the performance in that subject only of all the students 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 student is appearing.	Expulsion from the examination hall and cancellation of the performance in that subject and all other subjects the student 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.
3.	Impersonates any other student in connection with the examination	The student who has impersonated shall be expelled from examination hall. The student is also debarred and forfeits the seat. The performance of the original student 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 student is also debarred for two consecutive semesters from class work and all university examinations. The continuation of the course by the student 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, 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 student 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 student is also debarred for two consecutive semesters from class work and all university examinations. The continuation of the course by the student is subject to the academic regulations in connection with forfeiture of seat.
5.	Uses objectionable, abusive or offensive language in the answer paper or in letters to the examiners or writes to the examiner requesting him to award pass marks.	Cancellation of the performance in that subject.
6.	Refuses to obey the orders of the chief superintendent/assistant –superintendent / any officer on duty or misbehaves or creates disturbance of any kind in and around the examination hall or organizes a walk out or instigates others to walk out, or threatens the officer-in charge or any person on duty in or outside the examination hall of any injury to his person or to any of his relations whether by words, either spoken or written or by signs or by visible representation, assaults the officer-in-charge, or any person on duty in or outside the examination hall or any of his relations, or indulges in any other act of misconduct or mischief which result in damage to or destruction of property in the examination hall or any part of the college campus or engages in any other act which in the opinion of the officer on duty amounts to use of unfair means or misconduct or has the tendency to disrupt the orderly conduct of the examination.	In case of students of the college, they shall be expelled from examination halls and cancellation of their performance in that subject and all other subjects the student(s) has (have) already appeared and shall not be permitted to appear for the remaining examinations of the subjects of that semester/year. The students also are debarred and forfeit their seats. In case of outsiders, they will be handed over to the police and a police case is registered against them.
7.	Leaves the exam hall taking away answer script or intentionally tears of the script or any part thereof inside or outside the examination hall.	Expulsion from the examination hall and cancellation of performance in that subject and all the other subjects the student 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 student is also debarred for two consecutive semesters from class work and all university examinations. The continuation of the course by the student 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 student 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 student is also debarred and forfeits the seat.

9.	If student of the college, who is not a student for the particular examination or any person not connected with the college indulges in any malpractice or improper conduct mentioned in clause 6 to 8.	Student of the colleges expulsion from the examination hall and cancellation of the performance in that subject and all other subjects the student 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 student 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 student 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 student 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 Principal for further action to award suitable punishment.	

G.NARAYANAMMA INSTITUTE OF TECHNOLOGY & SCIENCE
AUTONOMOUS (for Women)
Shaikpet, Hyderabad- 500 104

M.Tech. in COMPUTER SCIENCE AND ENGINEERING
Course Structure & Syllabus (GNR-25 Regulations) Applicable from AY 2025-2026 Batch

I YEAR **I SEMESTER**

Course Code	Course Title	L	T	P	Credits
Professional Core - I	Mathematical Foundations of Computer Science	3	0	0	3
Professional Core - II	Advanced Data Structures	3	0	0	3
Professional Elective - I	1. Database Programming with PL/SQL 2. Deep Learning 3. Natural Language Processing 4. Advanced UNIX Programming	3	0	0	3
Professional Elective - II	1. Applied Cryptography 2. Software Quality Engineering 3. Mining Massive Datasets 4. Agile Methodologies	3	0	0	3
Lab – I	Advanced Data Structures Lab	0	0	4	2
Lab – II	Professional Elective - I Lab	0	0	4	2
	Research Methodology & IPR	2	0	0	2
Audit – I	Audit Course- I	2	0	0	0
	Total	**16**	**0**	**8**	**18**

Note: Professional Elective- I and Professional Elective- I Lab must be of same course.

I YEAR **II SEMESTER**

Course Code	Course Title	L	T	P	Credits
Professional Core - III	Advanced Algorithms	3	0	0	3
Professional Core - IV	Computational Intelligence	3	0	0	3
Professional Elective - III	1. Enterprise Cloud Concepts 2. Cyber Security 3. Parallel computing 4. Large Language Models	3	0	0	3
Professional Elective - IV	1. Bioinformatics 2. Adhoc and Sensor Networks 3. Robotic Process Automation 4. Generative AI	3	0	0	3
Lab – III	Advanced Algorithms Lab	0	0	4	2
Lab – IV	Professional Elective - III Lab	0	0	4	2
	Mini Project with Seminar	0	0	4	2
Audit – II	Audit Course- II	2	0	0	0
	Total	**14**	**0**	**12**	**18**

Note: Professional Elective- III and Professional Elective- III Lab must be of same course.

II YEAR **I SEMESTER**

Course Code	Course Title	L	T	P	Credits
Professional Elective - V	1. Digital Forensics 2. Advanced Operating Systems 3. Quantum Computing 4. Prompt Engineering	3	0	0	3
Open Elective	Open Elective	3	0	0	3
Dissertation	Dissertation Work Review - II	0	0	12	6
	Total	**6**	**0**	**12**	**12**

II YEAR **II - SEMESTER**

Course Code	Course Title	L	T	P	Credits
Dissertation	Dissertation Work Review - III	0	0	12	6
Dissertation	Dissertation Viva-Voce	0	0	28	14
	Total	**0**	**0**	**40**	**20**

Note: *For Dissertation Work Review - I, Please refer GNR-25 Academic Regulations*

Audit Course I & II:

1. English for Research Paper Writing
2. Disaster Management
3. Sanskrit for Technical Knowledge
4. Value Education
5. Constitution of India
6. Pedagogy Studies
7. Stress Management by yoga
8. Personality Development Through Life Enlightenment Skills

Open Electives for other Departments:

1. Intellectual Property Rights
2. Generative AI
3. Intrusion Detection Systems
4. Digital Forensics

I Year M.Tech. CSE I Semester **L T P C**
Course Code: 531BA **3 0 0 3**

MATHEMATICAL FOUNDATIONS OF COMPUTER SCIENCE

(Professional Core – I)

Prerequisites: Discrete Mathematics, Probability and Statistics

Course Objectives:

This course will

1. Introduce the foundational concepts of discrete mathematics relevant to computer science and engineering.
2. Explain formal logic notation and diverse methods of mathematical proof, including induction.
3. Explore fundamental set theory, relations, applications in computing, graph theory concepts and their practical uses in computer science.
4. Study combinatorial principles such as permutations, combinations, counting techniques, recurrence relations, and generating functions.

UNIT 1: (~ 10 Lecture Hours)

The Foundations Logic and Proofs: Propositional Logic, Applications of Propositional Logic, Propositional Equivalence, Predicates and Quantifiers, Nested Quantifiers, Rules of Inference, Introduction to Proofs, Proof Methods and Strategy.

Fundamentals: Random Variables, Discrete Probability Distributions, Probability Theory, Bayes Theorem, Continuous Probability Distributions, expected value, Variance, Conditional Expectation, Probability Distributions- Binomial, Poisson and Normal Distribution

UNIT 2: (~ 10Lecture Hours)

Basic Structures, Sets, Functions, Sequences, Sums, Matrices and Relations: Sets, Functions, Sequences & Summations, Cardinality of Sets and Matrices Relations, Relations and Their Properties, n-ary Relations and Their Applications, Representing Relations, Closures of Relations, Equivalence Relations, Partial Orderings

UNIT 3: (~ 8 Lecture Hours)

Algorithms, Induction and Recursion: Algorithms, The Growth of Functions, Complexity of Algorithms. Induction and Recursion: Mathematical Induction, Strong Induction and Well-Ordering, Recursive Definitions and Structural Induction, Recursive Algorithms, Program Correctness

UNIT 4: (~ 10 Lecture Hours)

Advanced Counting Techniques: Recurrence Relations, Solving Linear Recurrence Relations, Divide-and-Conquer Algorithms and Recurrence Relations, Generating Functions, Inclusion-Exclusion, Applications of Inclusion-Exclusion

UNIT 5: (~ 8 Lecture Hours)

Graphs: Graphs and Graph Models, Graph Terminology and Special Types of Graphs, Representing Graphs and Graph Isomorphism, Connectivity, Euler and Hamilton Paths, Shortest-Path Problems, Planar Graphs, Graph Coloring

Trees: Introduction to Trees, Applications of Trees, Tree Traversal, Spanning Trees, Minimum Spanning Trees

Text Books:

1. Kenneth H. Rosen, Discrete mathematics and its applications, 8th edition, McGraw-hill, 2019.
2. J.P. Tremblay, R.Manohar, Discrete Mathematical Structures with Applications to Computer Science, McGraw-hill, 1st edition.
3. Joe Mott, Abraham. Kandel, Teodore P. Baker, Discrete Mathematics for Computer Scientists & Mathematicians, Prentis Hall of India, 2nd edition.

Reference Books:

1. Ralph.P. Grimald, Discrete and Combinatorial Mathematics - an applied introduction, Pearson education, 5th edition.
2. Thomas Kosy, Discrete Mathematical Structures, Tata McGraw Hill publishing co.

Online Resources:

1. https://edutainmentzone.blogspot.com › Home › DS › Education
2. https://www.geeksforgeeks.org/engineering-mathematics/discrete-mathematics-tutorial/

Course Outcomes:

After completion of the course, students will be able to

1. Analyze and construct rigorous mathematical proofs involving fundamental concepts of probability and various probability distributions.
2. Apply principles of logic and set theory to formulate precise mathematical statements.
3. Analyze finite and discrete structures to solve complex counting problems using combinatorial techniques.
4. Understand and describe the properties of sequences and related mathematical constructs.
5. Apply graph theory concepts to model and solve computing problems.

I Year M.Tech. CSE I Semester **L T P C**
Course Code: 531BB **3 0 0 3**

ADVANCED DATA STRUCTURES

(Professional Core – II)

Prerequisites: Data Structures

Course Objectives:
This course will enable the students to

1. Understand the theoretical foundations of advanced data structures and their role in efficient problem solving.
2. Analyze and compare the efficiency of various heaps, search trees, and hashing methods for real-time applications.
3. Apply pattern-matching algorithms and string processing techniques to domains such as text retrieval and data mining.
4. Evaluate the trade-offs between different data structures and select appropriate ones for complex, large-scale applications.

UNIT 1: (~ 8 Lecture Hours)
Heap Structures- Introduction, Min-Max Heaps, Leftist Trees, Binomial Heaps, Fibonacci Heaps

UNIT 2: (~ 10 Lecture Hours)
Hashing and Collisions- Introduction, Hash Tables, Hash Functions, Division Method, Multiplication Method, Mid-Square Method, Folding Method, Collisions

UNIT 3: (~ 10 Lecture Hours)
Search Structures- Optimal Binary Search Trees (OBST), AVL Trees, Red-Black Trees, Splay Trees, Multiway Search Trees, B-trees, 2-3 Trees

UNIT 4: (~ 8 Lecture Hours)
Digital Search Structures- Digital Search Trees, Binary Tries and Patricia, Multiway tries, Suffix Trees, Standard Tries, Compressed Tries

UNIT 5: (~ 9 Lecture Hours)
Pattern Matching- Introduction, Brute Force, Boyer–Moore Algorithm, Knuth-Morris-Pratt (KMP) Algorithm, Naïve String Matching, Horspool Algorithm, Rabin-Karp Algorithm

Text Books:

1. Sahni, Horowitz, Mehatha, Fundamentals of data structures in C++, Universities Press.
2. TH Cormen, Introduction to Algorithms, PHI.

Reference Books:

1. SK Basu, Design methods and analysis of Algorithms, PHI.
2. Mark Allen Weiss, Data Structures & Algorithm Analysis in C++, Pearson Education.
3. Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekaran, Fundamentals of Computer Algorithms, 2nd edition, Universities Press.

Online Resources:

1. https://www.programiz.com/dsa
2. https://www.hackerearth.com/practice/data-structures/advanced-data-structures
3. https://www.nptelprep.in/courses/106102064/materials

Course Outcomes:

After completion of the course, students will be able to

1. Implement and analyze advanced heap structures for priority-based applications.
2. Apply modern hashing techniques for efficient data storage and retrieval.
3. Construct and compare balanced search trees (AVL, Red-Black, B+ Trees) to optimize searching and indexing operations.
4. Develop efficient text indexing using tries, suffix trees, and suffix arrays for large-scale data applications.
5. Evaluate advanced pattern-matching algorithms (KMP, Boyer-Moore) in domains like intrusion detection, search engines, and computational biology.

I Year M.Tech. CSE I Semester **L T P C**
Course Code: 531BC **3 0 0 3**

DATABASE PROGRAMMING WITH PL/SQL

(Professional Elective – I)

Prerequisites: Database Management Systems

Course Objectives:

This course will enable the students to

1. Get knowledge on significance of SQL fundamentals.
2. Evaluate functions and triggers of PL/SQL.
3. Acquire knowledge on control structures, packages in PL/SQL and its applications.
4. Develop skills to implement advanced database features.

UNIT 1: (~ 8 Lecture Hours)

PL/SQL Basics

Block Structure, Behavior of Variables in Blocks, Basic Scalar and Composite Data Types, Control Structures, Exceptions, Bulk Operations, Functions, Procedures, and Packages, Transaction Scope

UNIT 2: (~ 10 Lecture Hours)

Language Fundamentals & Control Structures

Lexical Units, Variables and Data Types, Conditional Statements, Iterative Statements, Cursor Structures, Bulk Statements, Introduction to Collections, Object Types: Varray and Table Collections, Associative Arrays, Oracle Collection API

UNIT 3: (~ 9 Lecture Hours)

Functions and Procedures

Function and Procedure Architecture, Transaction Scope, Calling Subroutines, Positional Notation, Named Notation, Mixed Notation, Exclusionary Notation, SQL Call Notation, Functions, Function Model Choices, Creation Options, Pass-by-Value Functions, Pass-by-Reference Functions, Procedures, Pass-by-Value Procedures, Pass-by-Reference Procedures, Supporting Scripts

UNIT 4: (~ 9 Lecture Hours)

Packages

Package Architecture, Package Specification, Prototype Features, Serially Reusable Precompiler Directive, Variables, Types, Components: Functions and Procedures, Package Body, Prototype Features, Variables, Types, Components: Functions and Procedures, Definer vs. Invoker Rights Mechanics, Managing Packages in the Database Catalog, Finding, Validating, and Describing Packages, Checking Dependencies, Comparing Validation Methods: Timestamp vs. Signature

UNIT 5: (~ 9 Lecture Hours)

Triggers

Introduction to Triggers, Database Trigger Architecture, Data Definition Language Triggers, Event Attribute Functions, Building DDL Triggers, Data Manipulation Language Triggers, Statement-Level Triggers, Row-Level Triggers, Compound Triggers, INSTEAD OF Triggers, System and Database Event Triggers, Trigger Restrictions, Maximum Trigger Size, SQL Statements, LONG and LONG RAW Data Types

Text Books:

Michael McLaughlin, Oracle Database 12c PL/SQL Programming, McGraw Hill Education.

Reference Books:

1. Benjamin Rosenzweig, Elena Silvestrova Rakhimov, Oracle PL/SQL by example 5th eition.
2. Dr. P. S. Deshpande, SQL & PL / SQL for Oracle 11g Black Book.

Online Resources:

1. https://www.geeksforgeeks.org/plsql/pl-sql-tutorial/
2. https://www.tutorialspoint.com/plsql/index.htm/

Course Outcomes:

After completion of the course, students will be able to

1. Apply the fundamentals of PL/SQL programming constructs and database interaction.
2. Analyze the use of collections, exceptions, and triggers to handle complex database operations.
3. Interpret PL/SQL features to build efficient, reliable, and scalable database applications.
4. Develop modular database programs using functions, procedures, and packages.
5. Construct different types of triggers to automate database operations and enforce business rules.

I Year M.Tech. CSE I Semester **L T P C**
Course Code: 531BD **3 0 0 3**

DEEP LEARNING

(Professional Elective – I)
(Common to CSE & CNIS)

Course Objectives:
This course will

1. Provide knowledge of deep learning architectures and optimization strategies.
2. Develop skills for analyzing challenges in neural network training.
3. Explore applications of deep learning in computer vision and natural language processing.
4. Encourage innovation in applying deep learning to real-world problems.

UNIT 1: (~ 9 Lecture Hours)
Introduction: Feed forward Neural networks, Gradient descent and the back propagation algorithm, Unit saturation, the vanishing gradient problem, and ways to mitigate it. ReLU Heuristics for avoiding bad local minima, Heuristics for faster training, Nestors accelerated gradient descent, Regularization, Dropout

UNIT 2: (~ 9 Lecture Hours)
Convolutional Neural Networks: Architectures, convolution/pooling layers, Recurrent Neural Networks: LSTM, GRU, Encoder Decoder architectures. Deep Unsupervised Learning: Auto encoders, Variational Auto-encoders, Adversarial Generative Networks, Auto-encoder and DBM Attention and memory models, Dynamic Memory Models

UNIT 3: (~ 9 Lecture Hours)
Applications of Deep Learning to Computer Vision: Image segmentation, object detection, automatic image captioning, Image generation with Generative adversarial networks, video to text with LSTM models, Attention Models for computer vision tasks

UNIT 4: (~ 9 Lecture Hours)
Applications of Deep Learning to NLP: Introduction to NLP and Vector Space Model of Semantics, Word Vector Representations: Continuous Skip-Gram Model, Continuous Bag-of-Words model (CBOW), Glove, Evaluations and Applications in word similarity

UNIT 5: (~ 9 Lecture Hours)
Analogy reasoning: Named Entity Recognition, Opinion Mining using Recurrent Neural Networks, Parsing and Sentiment Analysis using Recursive Neural Networks, Sentence Classification using Convolutional Neural Networks, Dialogue Generation with LSTMs

Text Books:

1. Ian Goodfellow, Yoshua Bengio and Aaron Courville, Deep Learning, MIT Press.
2. Mohamed Elgendy. Oreilly, Deep Learning for Vision System, 2020.

Reference Books:

1. D. Jurafsky and J. H. Martin, Speech and Language Processing, 3rd edition (Draft). Stanford, CA, USA: Stanford Univ., 2023.
2. Y. Goldberg, Neural Network Methods in Natural Language Processing. San Rafael, CA, USA: Morgan & Claypool, 2017.
3. R. Shanmugamani, Deep Learning for Computer Vision, Birmingham, U.K., Packt Publishing, 2018.
4. P. Goyal, S. Pandey, and K. Jain, Deep Learning for Natural Language Processing. Berkeley, CA, USA, Apress, 2018.

Online Resources:

1. http://neuralnetworksanddeeplearning.com/
2. https://www.cs.utexas.edu/~ml/papers/venugopalan.iccv15.pdf
3. https://arxiv.org/abs/2111.07624
4. https://ieeexplore.ieee.org/document/7550880
5. https://aclanthology.org/D14-1080.pdf

Course Outcomes:

After completion of the course, students will be able to

1. Explain the fundamental concepts and techniques used in deep learning models and their training process.
2. Analyze the challenges encountered during the training of deep learning models.
3. Apply advanced deep learning models to diverse tasks in vision and language domains.
4. Evaluate different architectures for their suitability in specific applications.
5. Design and develop deep learning-based solutions to address real-world challenges across domains by integrating attention mechanisms, generative models, and memory networks.

I Year M.Tech. CSE I Semester **L T P C**
Course Code: 531BE **3 0 0 3**

NATURAL LANGUAGE PROCESSING
(Professional Elective-1)

Course Objectives:
This course will enable the students to
1. Identify core concepts and challenges in language processing and analyze linguistic structure at various levels.
2. Interpret the meaning and connections in language to help machines better comprehend it.
3. Analyze how natural language generation systems work to perform various tasks.
4. Assess the effectiveness of systems for natural language understanding and generation.

UNIT 1: (~10 Lecture Hours)
Finding the Structure of Words: Words and their Components, Issues and Challenges, Morphological Models
Finding the Structure of Documents: Introduction, Methods, Complexity of Approaches, Performances of the Approaches

UNIT 2: (~9 Lecture Hours)
Syntax: Parsing Natural Language, Treebanks: A Data-Driven Approach to Syntax, Representation of Syntactic Structure, Parsing Algorithms, Models for Ambiguity Resolution in Parsing, Multilingual Issues: What is a Token?

UNIT 3: (9 Lecture Hours)
Semantic Parsing: Introduction, Semantic Interpretation, System Paradigms, Word sense, Predicate-Argument Structure, Meaning Representation

UNIT 4: (~ 9 Lecture Hours)
Discourse Processing: Introduction, Cohesion, Reference Resolution, Discourse Coherence and Structure
Natural Language Generation: Introduction, Architectures of NLG Systems, Generation Tasks and Representations, Applications of NLG

UNIT-5: (~ 8 Lecture Hours)
Language Modeling: Introduction, n-Gram Models, Language Model Evaluation, Parameter Estimation, Language Model Adaptation, Types of Language Models- Class-Based Language Models, Syntax-Based Language Models, Neural Network Language Models, Language-Specific Modeling Problems, Multilingual and Cross Lingual Language Modeling

Text Books:
1. Daniel M. Bikel & Imed Zitouni, Multilingual Natural Language Processing Applications: From Theory to Practice, Pearson Publication, 2013.
2. Tanveer Siddiqui, U.S. Tiwary, Natural Language Processing and Information Retrieval, Oxford Higher Education, 2008.

Reference Books:

1. Daniel Jurafsky and James H. Martin, Speech and Language Processing, An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition, 3rd edition(Draft), 2020.
2. Steven Bird, Ewan Klein and Edward Loper, Natural Language Processing with Python, O Reilly, 2009.

Online Resources:

1. https://onlinecourses.nptel.ac.in/noc23_cs45/preview
2. Natural Language Processing Specialization (DeepLearning.AI) | Coursera
3. Stanford CS 224N | Natural Language Processing with Deep Learning

Course Outcomes:

After completion of the course, students will be able to

1. Understand the key ideas and challenges involved in processing natural language.
2. Apply different methods and parsing algorithms to analyze and interpret the structure of language at various levels.
3. Interpret meaning and relationships within language to improve comprehension by machines.
4. Analyze the architectures of natural language generation systems to understand how they perform different generation tasks and applications.
5. Evaluate the systems that can effectively understand and generate natural language.

I Year M.Tech. CSE I Semester **L T P C**
Course Code: 531BF **3 0 0 3**

ADVANCED UNIX PROGRAMMING

(Professional Elective-1)

Course Objectives:

This course will enable the students to

1. Understand the core concepts, structure and essential tools of the Unix operating system.
2. Apply various Unix commands and techniques to solve programming tasks and automate processes.
3. Analyze different approaches and tools in Unix to solve complex problems efficiently.
4. Design and build advanced applications by applying various Unix concepts and system programming techniques.

UNIT 1: (~10 Lecture Hours)

Linux Utilities - File handling utilities, Security by file permissions, Process utilities, Disk utilities, Networking commands, Filters, Text processing utilities and Backup utilities.

Shell programming with Bourne again shell (bash) - Introduction, shell responsibilities, 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, interrupt processing, functions, debugging shell scripts

UNIT 2: (~9 Lecture Hours)

Files and Directories - File Concept, File types, File System Structure, file metadata-inodes, kernel support for files, system calls for file I/O operations- open, creat, read, write, close, lseek, dup2, file status information-stat family, file and record locking- fcntl function, file permissions - chmod, fchmod, file ownership-chown, lchown, fchown, links-soft links and hard links – symlink, link, unlink. Directories - Creating, removing and changing directories- mkdir, rmdir, chdir, obtaining current working directory-getcwd, Directory contents, Scanning Directories-opendir, readdir, closedir, rewinddir functions

UNIT 3: (9 Lecture Hours)

Process – Process concept, Layout of a C program image in main memory, Process environment-Environment list, Environment variables, getenv, setenv, Kernel support for process, process identification, process control - process creation, replacing a process image, waiting for a process, process termination, zombie process, orphan process, system call interface for process management-fork, vfork, exit, wait, waitpid, exec family, Process Groups, Sessions and Controlling Terminal, Differences between threads and processes

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

UNIT 4: (~ 9 Lecture Hours)

Interprocess Communication - Introduction to IPC, IPC between processes on a single computer system, IPC between processes on different systems, 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

Message Queues - Kernel support for messages, APIs for message queues, client/server example

Semaphores - Kernel support for semaphores, APIs for semaphores, file locking with semaphores

UNIT 5: (~ 8 Lecture Hours)
Shared Memory - Kernel support for shared memory, APIs for shared memory, shared memory example
Sockets - Introduction to Berkeley Sockets, IPC over a network, Client- Server model, Socket address structures (Unix domain and Internet domain), Socket system calls for connection oriented protocol and connectionless protocol, example-client/server programs-Single Server-Client connection, Multiple simultaneous clients, Socket options- setsockopt and fcntl system calls, Comparison of IPC mechanisms

Text Books:

1. Terrence Chan, Unix System Programming using C++, PHI.
2. Sumitabha Das, Unix: Concepts and Applications, 4th edition, TMH.

Reference Books:

1. W. R. Stevens and S. A. Rago, Advanced Programming in the Unix Environment, 2nd edition, Pearson Education.
2. B. A. Forouzan and R. F. Gilberg, Unix and Shell programming, Cengage Learning.
3. Robert Love, Linux System Programming, O'Reilly.
4. W. Richard Stevens, Unix Network Programming, PHI.

Online Resources:

1. https://www.coursera.org/specializations/advanced-unix-system-programming-performance
2. https://www.udemy.com/topic/unix/?srsltid=AfmBOoqd8z1bu6FlkO7n7MXMr7v7wwtzErGOoP0aEqC-mdnYSGW25E7X
3. https://www.coursera.org/courses?query=unix

Course Outcomes:
After completion of the course, students will be able to

1. Understand the structure of the Unix operating system, essential commands and shell features.
2. Apply Unix commands, shell scripts and system calls to solve basic programming tasks.
3. Analyze Unix internal commands to identify optimal approaches for concurrency, scheduling and resource sharing.
4. Evaluate different Unix programming constructs in problem-solving.
5. Design and develop advanced Unix-based applications that integrate system calls to solve real-world problems.

I Year M.Tech. CSE I Semester **L T P C**
Course Code: 531BG **3 0 0 3**

APPLIED CRYPTOGRAPHY

(Professional Elective – II)

Prerequisites: Cryptography and Network Security

Course Objectives:

This course will enable the students to

1. Understand the various cryptographic protocols.
2. Analyze key length and algorithm types and modes.
3. Illustrate different public key algorithms in crypto systems.
4. Understand special algorithms for protocols and usage in the real world.

UNIT 1: (~ 8 Lecture Hours)

Foundations: Terminology, Steganography, Substitution Ciphers and Transposition Ciphers, Simple XOR, One-Time Pads, Computer Algorithms, Large Numbers

Cryptographic Protocols: Protocol Building Blocks: Introduction to Protocols, Communications Using Symmetric Cryptography, One-Way Functions, One-Way Hash Functions, Communications Using Public-Key Cryptography, Digital Signatures, Digital Signatures with Encryption, Random and Pseudo-Random-Sequence Generation

UNIT 2: (~ 10 Lecture Hours)

Cryptographic Techniques: Key length: Symmetric Key length, Public key length, comparing symmetric and public key length

Algorithm types and modes: Electronic Codebook Mode, Block Replay, Cipher Block Chaining Mode, Stream Cipher, Self-Synchronizing Stream Ciphers, Cipher-Feedback Mode, Synchronous Stream Ciphers, Output-Feedback Mod, Counter Mode, Other Block-Cipher Modes

UNIT 3: (~ 8 Lecture Hours)

Public-Key Algorithms: Background, Knapsack Algorithms, RSA, Pohlig-Hellman, Rabin, ElGamal, McEliece, Elliptic Curve Cryptosystems, LUC, Finite Automaton Public-Key Cryptosystems

Public-Key Digital Signature Algorithms: Digital Signature Algorithm(DSA), DSA Variants, Gost Digital Signature Algorithm, Discrete Logarithm Signature Schemes, Ong-Schnorr-Shamir, ESIGN

UNIT 4: (~ 10 Lecture Hours)

Special Algorithms for Protocols: Multiple-Key Public-Key Cryptography, Secret-Sharing Algorithms, Subliminal Channel, Undeniable Digital Signatures, Designated Confirmer Signatures, Computing with Encrypted Data, Fair CoinFlips, One-Way Accumulators, All-or-Nothing Disclosure of Secrets, Fair and Failsafe Cryptosystems, Zero-Knowledge Proofs of Knowledge, Blind Signatures, Oblivious Transfer

Secure Multiparty Computation, Probabilistic Encryption, Quantum Cryptography

UNIT 5: (~ 9 Lecture Hours)

Real World Approaches: IBM Secret key management protocol, ISDN, Kerberos, KryptoKnight, Privacy enhanced mail (PEM), Message security protocol (MSP), PGP, Public-Key Cryptography Standards (PKCS), Universal Electronic Payment System (UEPS)

Textbooks:

Bruce Schneier, Applied Cryptography, Protocols, Algorithms, and Source Code in C (cloth), 2nd edition.

Reference Books:

1. Christof Paar and Jan Pelzl, Bart Preneel -Foreword, Understanding Cryptography,2010, Kindle Edition,Springer.
2. Arto Salomaa, Public-Key Cryptography, Texts in Theoretical Computer Science. An EATCS Series, 2nd Edition, Kindle Edition.Springer

Online Resources:

1. https://www.crypto101.io/
2. https://www.tutorialspoint.com/cryptography/
3. https://www.geeksforgeeks.org/cryptography-and-its-types/
4. https://api.pageplace.de/preview/DT0400.9781439895849_A37871705/preview-9781439895849_A37871705.pdf
5. https://csrc.nist.gov/projects/post-quantum-cryptography/post-quantum-cryptography-standardization
6. https://link.springer.com/book/10.1007/978-3-540-88702-7

Course Outcomes:

After completion of the course, students will be able to

1. Explain fundamental cryptographic terminologies, classical ciphers, and protocols used in secure communication.
2. Analyze key lengths, cryptographic algorithm types, and operating modes for their security and efficiency.
3. Apply public-key algorithms and digital signature techniques in cryptosystems.
4. Evaluate special algorithms for protocols such as secret sharing, zero-knowledge proofs, and secure multiparty computation.
5. Demonstrate real-world cryptographic applications in protocols like Kerberos, PGP, PKCS, and electronic payment systems.

I Year M.Tech. CSE I Semester **L T P C**
Course Code: 531BH **3 0 0 3**

SOFTWARE QUALITY ENGINEERING

(Professional Elective – II)

Course Objectives:
This course will enable the students to

1. Understanding the Importance of Quality.
2. Gain Insights into Quality Assurance Practices.
3. Explore the Fundamentals of Quality Engineering.
4. Master Key Testing Techniques and Activities.

UNIT 1: (~ 9 Lecture Hours)
Software Quality
Quality: perspectives and expectations, Quality frameworks and ISO-9126. Correctness and defects: Definitions, properties and Measurements, A historical perspective of quality, software quality

UNIT 2: (~ 10 Lecture Hours)
Quality Assurance
Classification: QA as dealing with defects, Defect prevention- Education and training, Formal method, Other defect prevention techniques. Defect Reduction - Inspection: Direct fault detection and removal. Testing: Failure observation and fault removal, other techniques and risk identification, Defect Containment- software fault tolerance, safety assurance and failure containment

UNIT 3: (~ 9 Lecture Hours)
Quality Engineering
Quality Engineering: Activities and process. Quality planning: Goal setting and Strategy formation, Quality assessment and Improvement, Quality engineering in software process

UNIT 4: (~ 8 Lecture Hours)
Test Activities, Management and Automation
Test planning and preparation, Test execution, Result checking and measurement, Analysis and follow- up, Activities People and Management, Test Automation

UNIT 5: (~ 9 Lecture Hours)
Coverage and usage testing based on checklist and partitions
Checklist based testing and its limitations, Testing for partition Coverage, Usage based Statistical testing with Musa's operational profiles, Constructing operational profiles. Case Study: OP for the cartridge Support Software

Text Books:

1. Jeff Tian, Software Quality Engineering, Testing, Quality Assurance, and Quantifiable improvement, 1st edition, Wiley–IEEE Computer Society Press**,** 2005**.**
2. Richard N. Taylor, Software Architecture: Foundations, Theory, and Practice, 1st edition, Wiley, 2009.

Reference Books:

1. Ian Sommerville, Software Engineering, 10th edition, Pearson, 2015.
2. RogerS.Pressman & Bruce R.Maxim, Software Engineering: A Practitioner's Approach, 8th edition, McGraw-Hill, 2015.

Online Resources:

https://edutainmentzone.blogspot.com › Home › DS › Education

Course Outcomes:

After completion of the course, students will be able to

1. Explain perspectives, frameworks, and standards (e.g., ISO-9126) that define software quality.
2. Apply quality assurance techniques (defect prevention, detection, reduction, and containment.
3. Analyze software processes using quality engineering principles for planning, assessment and improvement.
4. Evaluate testing activities, management strategies, and automation tools.
5. Design operational profiles and testing strategies for software quality improvement.

I Year M.Tech. CSE I Semester **L T P C**
Course Code: 531BJ **3 0 0 3**

MINING MASSIVE DATASETS
(Professional Elective - II)

Prerequisites: Machine Learning

Course Objectives:
This course will enable the students to
1. Apply algorithmic techniques to solve fundamental problems in mining large and complex datasets.
2. Analyze and implement parallel algorithms designed for efficient processing of massive data.
3. Evaluate stream processing algorithms for handling continuously arriving data in real-time environments
4. Explain and assess the principles behind page ranking algorithms and online advertisement systems.

UNIT 1: (~9 Lecture Hours)
Data Mining-Introduction-Definition of Data Mining-Statistical Limits on Data Mining, Map Reduce and the New Software Stack-Distributed File Systems, Map Reduce, Algorithms Using Map Reduce

UNIT 2: (~9 Lecture Hours)
Similarity Search: Finding Similar Items-Applications of Near-Neighbor Search, Shingling of Documents, Similarity-Preserving Summaries of Sets, Distance Measures. Streaming Data: Mining Data Streams-The Stream Data Model, Sampling Data in a Stream, Filtering Streams

UNIT 3: (~9 Lecture Hours)
Link Analysis-PageRank, Efficient Computation of PageRank, Link Spam Frequent Itemsets-Handling Larger Datasets in Main Memory, Limited-Pass Algorithms, Counting Frequent Items in a Stream. Clustering-The CURE Algorithm, Clustering in Non-Euclidean Spaces, Clustering for Streams and Parallelism

UNIT 4: (~9 Lecture Hours)
Advertising on the Web-Issues in On-Line Advertising, On-Line Algorithms, The Matching Problem, The Adwords Problem, Adwords Implementation. Recommendation Systems-A Model for Recommendation Systems, Content-Based Recommendations, Collaborative Filtering, Dimensionality Reduction, The Net Flix Challenge

UNIT 5: (~9 Lecture Hours)
Mining Social-Network Graphs-Social Networks as Graphs, Clustering of Social-Network Graphs, Partitioning of Graphs, Simrank, Counting Triangles

Text Books:

Jure Leskovec, Anand Rajaraman, Jeff Ullman, Mining of Massive Datasets, 3rd edition.

Reference Books:

1. Jiawei Han and Micheline Kamber, Data Mining – Concepts and Techniques 3rd edition Elsevier.
2. Margaret H Dunham, Data Mining Introductory and Advanced topics, PEA.
3. Ian H. Witten and Eibe Frank, Data Mining: Practical Machine Learning Tools and Techniques, Morgan Kaufmann.

Online Resources:

1. http://www.mmds.org/
2. https://www.coursera.org/learn/scala-spark-big-data

Course Outcomes:

After completion of the course, students will be able to

1. Implement Map Reduce programming techniques to efficiently handle massive datasets.
2. Develop and implement algorithms for massive data sets and methodologies in the context of data mining.
3. Analyze and evaluate algorithms used to extract models and meaningful information from massive datasets.
4. Design and assess recommendation systems using content-based and collaborative filtering approaches.
5. Compare and select appropriate algorithms for solving specific classes of data mining problems.

I Year M.Tech. CSE I Semester **L T P C**
Course Code: 531BK **3 0 0 3**

AGILE METHODOLOGIES
(Professional Elective – II)

Course Objectives:
This course will enable the students to
1. Understand the benefits and pitfalls of agile model.
2. Acquire knowledge of agile software development practices and how small teams can apply them to create high- quality software.
3. Provide a good understanding of software design and a set of software technologies.
4. Perform detailed examination and demonstration of Agile development and testing techniques.

UNIT 1: (~10 Lecture Hours)
Introduction Extreme Programming (XP) - Agile Development Why Agile - Understanding Success, Beyond Deadlines, Importance of Organizational Success, Introduction to Agility, How to Be Agile - Agile methods, Don't make your own method, Road to mastery, Understanding XP (Extreme Programming) - XP life cycle, XP team, XP Concepts, Adopting XP - Knowing whether XP is suitable, Implementing XP, assessing Agility, Practicing XP - Thinking - Pair Programming, Energized work, Informative Workspace, Root cause Analysis, Retrospectives

UNIT 2: (~8 Lecture Hours)
Collaborating: Trust, Sit together, Real customer involvement, Ubiquitous language, meetings, coding standards, Iteration demo, Reporting

UNIT 3: (~8 Lecture Hours)
Releasing: Bug free Release, Version Control, fast build, continuous integration, Collective ownership, Documentation, Introduction to bug tracking tool (Monday.com)

UNIT 4: (~10 Lecture Hours)
Planning: Version, Release Plan, Risk Management, Iteration Planning, Slack, Stories, Estimating.
Developing: Incremental requirements, Customer tests, Test-driven development, Refactoring, Incremental design and architecture, spike solutions, Performance optimization, Exploratory testing

UNIT 5: (~ 9 Lecture Hours)
Mastering Agility: Values and Principles, Improve the process, Rely on People, Eliminate Waste, Deliver Value, Seek Technical Excellence

Text Books:
1. James Shore and Shane Warden, The art of Agile Development, 11th Indian Reprint, O'Reilly, 2018.
2. Ken Schwaber and Mike Beedle, Agile Development with Scrum, Prentice Hall, 2001.

Reference Books

1. Andrew Stellman and Jennifer Greene, Learning Agile, O'Reilly, 4th Indian Reprint, 2018.
2. Venkat Subramaniam and Andy Hunt, SPD, Practices of an Agile Developer, 5th Indian Reprint, 2015.
3. Jim Highsmith, Agile Project Management - Pearson Low price, Edition 2004.
4. Alistair Cockburn, Agile Software Development – The Cooperative Game, 2nd edition, 2007.

Online Resources:

1. https://www.coursera.org/learn/agile-project-management
2. https://WWW.Coursera.org/specializations/agile-development

Course Outcomes:

After completion of the course, students will be able to

1. Analyze the importance of interacting with business stakeholders in determining the re requirements for a software system.
2. Apply iterative software development processes like how to plan them and how to execute them.
3. Develop techniques and tools for improving team collaboration and software quality.
4. Analyze software process improvement as an ongoing task for development teams.
5. Demonstrate mastery of agile values and principles, and show how agile approaches can be effectively scaled up to the enterprise level.

I Year M.Tech. CSE I Semester **L T P C**
Course Code: 53111 **0 0 4 2**

ADVANCED DATA STRUCTURES LAB

(Professional Core – II)

Prerequisites: Data Structures

Course Objectives:

This course will enable the students to

1. Implement advanced data structures such as trees, heaps, sorting techniques and hashing for efficient data organization and retrieval.
2. Develop skills to construct and manipulate balanced search trees such as AVL and Red-Black trees for optimal performance.
3. Apply efficient pattern matching algorithms for string searching in real-world applications.
4. Enhance problem-solving and programming skills through practical implementation of theoretical concepts.

List of Programs:

Week 1:

Write a program to perform the following operations:

a) Insert an element into a binary search tree.
b) Delete an element from a binary search tree.
c) Search for a key element in a binary search tree.

Week 2:

Write a program for implementing the following sorting methods:

a) Merge sort b) Heap sort c) Quick sort

Week 3:

Write a program to perform the following operations:

a) Insert an element into a B- tree
b) Delete an element from a B- tree
c) Search for a key element in a B- tree

Week 4:

Write a program to perform the following operations:

a) Insert an element into a Min-Max heap
b) Delete an element from a Min-Max heap
c) Search for a key element in a Min-Max heap

Week 5:

Write a program to perform the following operations:

a) Insert an element into a Leftist tree
b) Delete an element from a Leftist tree
c) Search for a key element in a Leftist tree

Week 6:

Write a program to perform the following operations:

a) Insert an element into a binomial heap
b) Delete an element from a binomial heap
c) Search for a key element in a binomial heap

Week 7:

Write a program to perform the following operations:

a) Insert an element into a AVL tree
b) Delete an element from a AVL search tree
c) Search for a key element in a AVL search tree

Week 8:

Write a program to perform the following operations:

a) Insert an element into a Red-Black tree
b) Delete an element from a Red-Black tree
c) Search for a key element in a Red-Black tree

Week 9:

Write a program to implement all the functions of a dictionary using hashing.

Week 10:

Write a program for implementing Knuth-Morris-Pratt pattern matching algorithm.

Week 11:

Write a program for implementing Brute Force pattern matching algorithm.

Week 12:

Write a program for implementing Boyer pattern matching algorithm.

Text Books:

1. E. Horowitz, S. Sahni and Susan Anderson Freed, Fundamentals of Data structures in C, 2nd edition, Universities Press.
2. CA.S. Tanenbaum, Y. Langsam, and M.J. Augenstein, Data Structures Using PHI/Pearson education.
3. Ashok Kamthane, Introduction to Data Structures in C, 1st edition, Pearson.

Reference Books:

1. B.W. Kernighan and Dennis M.Ritchie, The C Programming Language, PHI/Pearson Education
2. J.A. Jones and K. Harrow, C Programming with problem solving, Dreamtech Press
3. R.F. Gilberg and B.A. Forouzan,Data structures: A Pseudocode Approach with C, 2nd edition, Cengage Learning

Online Resources:

1. https://www.cs.emory.edu/~cheung/Courses/253/Syllabus/Text
2. https://www.geeksforgeeks.org/dsa

Course Outcomes:

After completion of the course, students will be able to

1. Implement and basic operations (insert, delete, search) on various data structures like BST, B-tree, Leftist tree, Min-Max heap, Binomial heap, AVL tree, and Red-Black tree.
2. Analyze the performance of advanced sorting techniques such as Merge Sort, Heap Sort, and Quick Sort.
3. Apply hashing techniques for efficient dictionary implementation and collision resolution.
4. Demonstrate the working of string-matching algorithms (Brute Force, KMP, Boyer-Moore) for searching substrings in large texts.
5. Evaluate and select appropriate data structures and algorithms based on problem requirements to optimize performance.

I Year M.Tech. CSE I Semester **L T P C**
Course Code: 53112 **0 0 4 2**

DATABASE PROGRAMMING WITH PL/SQL LAB

(Professional Elective – I)

Prerequisites: Database Management System

Course Objectives:

This course will enable the students to

1. Acquire knowledge on significance of SQL fundamentals.
2. Evaluate functions and triggers of PL/SQL.
3. Get knowledge on control structures, packages in PL/SQL and its applications.
4. Develop skills to implement advanced database features**.**

List of Programs:

Week 1:

Write a Pl/SQL program using FOR loop to insert ten rows into a database table.

Week 2:

a) Write a PL/SQL block that uses **composite data types (record)** to store and display employee details.
b) Given the table EMPLOYEE (EmpNo, Name, Salary, Designation, DeptID), write a cursor to select the five highest paid employees from the table.

Week 3:

Illustrate how you can embed PL/SQL in a high-level host language such as C/Java and demonstrates how a banking debit transaction might be done.

Week 4:

a) Given an integer i, write a PL/SQL procedure to insert the tuple (i, 'xxx') into a given relation.
b) Demonstrate the use of **VARRAY** by storing 10 student marks and displaying them.

Week 5:

Write a PL/SQL program to demonstrate Exceptions.

Week 6:

Write a PL/SQL program to demonstrate Cursors.

Week7:

Write a PL/SQL program to demonstrate Functions.

Week 8:

Write a PL/SQL **function** to return the total salary paid in the department given its department number.

Week 9:

a) Write a PL/SQL program to demonstrate Packages.
b) Write a PL/SQL package that has overloaded functions to calculate area of a circle and a rectangle**.**

Week 10:

Develop a package specification and body that includes procedures for inserting and deleting a student record.

Week 11:

Write PL/SQL queries to create Procedures.

Week 12:

Write PL/SQL queries to create Triggers.

Week 13:

Develop an **INSTEAD OF trigger** on a view that updates underlying base tables.

Week 14:

Create a **statement-level trigger** that prevents deletion of rows from the department table.

Text Books:

Michael McLaughlin, Oracle Database 12c PL/SQL Programming, McGraw Hill Education.

Reference Books:

1. Benjamin Rosenzweig and Elena Silvestrova Rakhimov, Oracle PL/SQL by example 5th edition.
2. Dr. P. S. Deshpande, SQL & PL / SQL for Oracle 11g Black Book.

Online Resources:

1. https://www.geeksforgeeks.org/plsql/pl-sql-tutorial/
2. https://www.tutorialspoint.com/plsql/index.htm/

Course Outcomes:

After completion of the course, students will be able to

1. Apply the fundamentals of PL/SQL programming constructs and database interaction.
2. Analyze the use of collections, exceptions, and triggers to handle complex database operations.
3. Interpret PL/SQL features to build efficient, reliable, and scalable database applications.
4. Develop modular database programs using functions, procedures, and packages.
5. Construct different types of triggers to automate database operations and enforce business rules.

I Year M.Tech. CSE I Semester **L T P C**
Course Code: 53113 **0 0 4 2**

DEEP LEARNING LAB

(Professional Elective – I)

Course Objectives:

This course will

1. Provide learners with practical exposure to setting up deep learning environments and implementing models using suitable programming frameworks.
2. Introduce fundamental and advanced deep learning architectures for solving a variety of real-world problems.
3. Develop the ability to apply deep learning methods to tasks in computer vision, natural language processing, and related domains.
4. Equip learners with skills to train, evaluate, and optimize deep learning models using appropriate datasets.

List of Programs:

Week 1:

Setting up the Spyder IDE Environment and Executing a Python Program.

Week 2:

Installing Keras, Tensorflow and Pytorch libraries and making use of them.

Week 3:

Using the COCO dataset, apply a Convolutional Neural Network (CNN) to perform object detection in images and evaluate the accuracy of the model.

Week4:

Design and implement a Convolutional Neural Network (CNN) with fully connected layers to classify handwritten digits from the MNIST dataset.

Week 5:

Demonstrate the effect of Dropout and L2 regularization on overfitting using a small image Dataset.

Week 6:

Implement semantic image segmentation using U-Net on a medical dataset (e.g., lung CT scan dataset).

Week 7:

Apply transfer learning with a pre-trained model (ResNet/VGG) on a custom dataset (small subset of COCO).

Week 8:

Implement image captioning using CNN + LSTM on Flickr8k dataset.

Week 9:

Train a Skip-Gram model to learn word embeddings on a text corpus and evaluate word similarity.

Week 10:

Train a sentiment analysis model on IMDB dataset, use RNN layers with LSTM/GRU.

Week 11:

Apply an Autoencoder to encode and reconstruct chest X-ray images from the NIH Chest X-ray14 open-source dataset.

Week 12:

Applying Generative Adversial Networks for image generation and unsupervised tasks.

Text Books:

1. Goodfellow, Yoshua Bengio and Aaron Courville, Deep Learning, Ian MIT Press.
2. R. Shanmugamani, Deep Learning for Computer Vision, Birmingham, U.K.: Packt Publishing, 2018.

Reference Books:

1. Géron, Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, 2nd edition. Sebastopol, CA, USA: O'Reilly Media, 2022.
2. R. Acharya and A. K. Ray, Deep Learning for Medical Image Analysis. Cambridge, MA, USA: Academic Press, 2017.
3. F. Chollet, Deep Learning with Python, 2nd edition. Shelter Island, NY, USA: Manning Publications, 2021.
4. P. Goyal, S. Pandey, and K. Jain, Deep Learning for Natural Language Processing. Berkeley, CA, USA: Apress, 2018.

Online Resources:

1. www.cs.toronto.edu/~fritz/absps/imagenet.pdf
2. http://neuralnetworksanddeeplearning.com/
3. https://github.com/fchollet/deep-learning-with-python-notebooks

Course Outcomes:

After completion of the course, students will be able to

1. Demonstrate the working of programming environments and commonly used deep learning frameworks.
2. Implement major deep learning architectures such as CNNs, RNNs, LSTMs, Auto encoders, and GANs for diverse tasks.
3. Analyze common training challenges in deep learning models and their impact on performance.
4. Evaluate the performance of deep learning models using appropriate datasets and metrics.
5. Design and develop deep learning-based solutions for practical applications across multiple domains.

I Year M.Tech. CSE I Semester **L T P C**
Course Code: 53114 **0 0 4 2**

NATURAL LANGUAGE PROCESSING LAB

(Professional Elective – I)

Course Objectives:

This course will enable the students to

1. Identify and apply techniques to preprocess and manipulate natural language text.
2. Analyze linguistic features and sentence structures using computational methods.
3. Implement algorithms to process and manipulate language data.
4. Assess smoothing techniques and probabilistic methods to improve language processing.

Week 1:

Tokenization

a) Write a Python program to perform sentence tokenization on a given paragraph. Display each sentence separately.
b) Write a Python program to perform word tokenization on a given paragraph. Display the list of words.

Week 2:

Stemming

Write a Python program that takes a paragraph as input, tokenizes it into words, and applies stemming to each word. Display the stemmed words.

Week 3:

Stop word removal

Write a Python program to remove common English stop words (such as “a”, “an”, “the”, etc.) from a given paragraph. Display the filtered text without the stop words.

Week 4:

Word Analysis

a) Write a Python program to count the frequency of each word in a given text. Display the top 5 most frequent words along with their counts.
b) Write a Python program to find and display the longest and shortest words in a given paragraph. Also, show their lengths.

Week 5:

Word Generation

a) Write a Python program to generate new words by adding common prefixes and suffixes to a given root word. Display the list of generated words.
b) Write a Python program to create random words by combining syllables or letter sequences. Generate and display 10 such random words.

Week 6:

PoS Tagging

a) Write a Python program to perform POS tagging on a given sentence. Display each word along with its corresponding POS tag.
b) Write a Python program to read a paragraph, tokenize it into sentences and words, then perform POS tagging on each sentence. Display the tagged output sentence-wise.

Week 7:
Morphology

a) Write a Python program to identify and extract prefixes and suffixes from a list of words. Display each word along with its detected prefix and suffix (if any).
b) Write a Python program to analyze the morphological structure of words by breaking them down into their root forms and affixes using basic string operations or available NLP libraries.

Week 8:
Chunking

Write a Python program to perform noun phrase chunking on a given sentence. Display the chunked phrases in a tree format.

Week 9:
N-grams

Write a Python program to generate unigrams, bigrams, and trigrams from a given sentence. Display each set of N-grams separately.

Week 10:
N-gram Smoothing

Write a Python program to compare bigram probabilities with and without add-one smoothing. Use a small sample sentence and display the results side by side.

Text Books:

1. Daniel M. Bikel and Imed Zitouni, Multilingual Natural Language Processing Applications: From Theory to Practice, Pearson Publication, 2013.
2. Tanveer Siddiqui and U.S. Tiwary, Natural Language Processing and Information Retrieval, Oxford Higher Education, 2008.

Reference Books:

1. Daniel Jurafsky and James H. Martin, Speech and Language Processing, An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition, 3rd edition(Draft), 2020.
2. Steven Bird, Ewan Klein and Edward Loper, Natural Language Processing with Python, O Reilly, 2009.

Online Resources:

1. https://onlinecourses.nptel.ac.in/noc23_cs45/preview
2. Natural Language Processing Specialization (DeepLearning.AI) | Coursera
3. Stanford CS 224N | Natural Language Processing with Deep Learning

Course Outcomes:

After completion of the course, students will be able to

1. Demonstrate the ability to preprocess and manipulate natural language text.
2. Apply computational methods to analyze linguistic features of text.
3. Implement language processing algorithms to generate and transform words and phrases.
4. Apply syntactic processing methods to analyze structural patterns in text.
5. Evaluate language models and smoothing techniques to understand and improve probabilistic language processing.

I Year M.Tech. CSE I Semester **L T P C**
Course Code: 53115 **0 0 4 2**

ADVANCED UNIX PROGRAMMING LAB

(Professional Elective – I)

Course Objectives:

This course will enable the students to

1. Develop proficiency in using Unix commands, shell features and scripting for system-level programming.
2. Apply Unix programming constructs and system calls to design and implement effective solutions to computing problems.
3. Analyze system behavior related to concurrency, scheduling and resource management within Unix environments.
4. Evaluate and create Unix-based applications that integrate advanced programming techniques to address real-world challenges.

Week 1:

a) Write a shell script that accepts a file name, starting and ending line numbers as arguments and displays all lines between the given line numbers.
b) Write a shell script that deletes all lines containing the specified word in one or more files supplied as arguments to it.

Week 2:

a) Write a shell script that displays a list of all files in the current directory to which the user has read, write and execute permissions.
b) Write a shell script that receives any number of file names as arguments, checks if every argument supplied is a file or directory and reports accordingly. Whenever the argument is a file it reports number of lines present in it.

Week 3:

a) 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.
b) Write a shell script to list all of the directory files in a directory.

Week 4:

a) Write a C program that makes a copy of a file using standard I/O and system calls.
b) Implement in C language, the following Unix commands using system calls
i) cat ii) ls iii) mv

Week 5:

Write a C program that takes one or more file/directory names as command line input and reports following information a) File Type b) Number of Links c) Time of last Access d) Read, write and execute permissions.

Week 6:

a) Write a C program to emulate the Unix ls –l command.
b) Write a C program to list every file in directory, its inode number and file name.
c) Write a C program that demonstrates redirection of standard output to a file Ex: ls > f1.

Week 7:

a) Write a C program to create child process and allow parent process to display "parent" and the child to display "child" on the screen.
b) Write a C program to create zombie process.
c) Write a C program to illustrate how an orphan process is created.

Week 8:

a) Write a C program that illustrates how to execute two commands concurrently with a command pipe. Ex: ls –l
b) Write C programs that illustrate communication between two unrelated processes using named pipes.

Week 9 & 10:

Case Study: Priority-Based Communication in a Railway Reservation System

In a railway reservation system, efficient coordination between the ticket booking module and the seat allocation module is crucial for timely processing of passenger requests. Different travel classes have varying priority levels, with First Class bookings typically processed before Second Class, and so on.

To simulate this, two independent modules communicate using System V message queues. The booking module sends requests with passenger and journey details, assigning priorities based on travel class. The seat allocation module receives these requests, processes them according to priority, and confirms seat availability.

To ensure synchronized and mutually exclusive access to shared resources like the message queue, semaphores are used. Semaphores help coordinate concurrent booking requests by controlling access, preventing race conditions, and managing resource availability, thus maintaining the integrity and efficiency of the reservation system.

Programming Tasks:

1. **Booking Sender:**
   Write a C program that creates a System V message queue and sends booking requests with varying priorities:
   - Priority 3 for First Class
   - Priority 2 for Second Class
   - Priority 1 for General Class

   Each booking request should include passenger name, journey details (source and destination), and travel class.

   Once it receives the booking details print the seat number allotted.
2. **Allocation Receiver:**
   Write a C program that connects the sender, reads the received messages from message queue, displays as per the priority order, and confirms the passenger, the details for seat allocation.

   * Allocation of seat number may be manual or automated
3. **Semaphore-Controlled Booking Sender**
   Write a C program that:
   - Uses a binary System V semaphore to ensure mutual exclusion on the booking sender side.
   - Demonstrate that at any given point, only one booking request is sent to the message queue.
   - Implements the semaphore operations (semget, semop, and semctl) to lock and unlock the critical section around sending messages.

4. **Producer-Consumer with Semaphores and Message Queue**
   Write C programs to:
   - Simulate multiple booking senders (producers) and one seat allocator (consumer).
   - Use a System V message queue for communication.
   - Use three semaphores:
     - empty — counts available slots (buffer size).
     - full — counts occupied slots.
     - mutex — ensures mutual exclusion.
   - Producers wait on empty and mutex, send booking requests, then signal mutex and full.
   - Consumer waits on full and mutex, receives messages, then signals mutex and empty.

Week 11 & 12:

Case Study: Real-Time Data Logging in an Environmental Monitoring System

An environmental monitoring station collects data from multiple sensors such as temperature, humidity, and air quality at regular intervals. These sensors act as producers, generating data that must be reliably transferred to a consumer process responsible for logging the data to a file or database.

To prevent data loss or overwriting when sensors produce data rapidly, a shared buffer is used with synchronization mechanisms. Using System V semaphores, the producer process places sensor data into the shared buffer only if space is available, while the consumer process reads from the buffer only when data is present. This coordination ensures data integrity and efficient resource use in a real-time environment.

Programming Tasks:

1. **Producer Process:**
   Write a C program that simulates sensor data generation and writes this data into a shared buffer. Use System V semaphores to ensure the producer waits if the buffer is full.
2. **Consumer Process:**
   Write a C program that reads sensor data from the shared buffer and logs it to a file. Use System V semaphores to ensure the consumer waits if the buffer is empty.

Text Books:

1. Terrence Chan, Unix System Programming using C++, PHI.
2. Sumitabha Das, Unix: Concepts and Applications, 4th edition, TMH.

Reference Books:

1. W. R. Stevens and S. A. Rago, Advanced Programming in the Unix Environment, 2nd edition, Pearson Education.
2. B. A. Forouzan and R. F. Gilberg, Unix and Shell programming, Cengage Learning.
3. Robert Love, Linux System Programming, O'Reilly.
4. W. Richard Stevens, Unix Network Programming, PHI.

Online Resources:

1. https://www.coursera.org/specializations/advanced-unix-system-programming-performance
2. https://www.udemy.com/topic/unix/?srsltid=AfmBOoqd8z1bu6FlkO7n7MXMr7v7wwtzErGOoP0aEqC-mdnYSGW25E7X
3. https://www.coursera.org/courses?query=unix

Course Outcomes:

After completion of the course, students will be able to

1. Demonstrate practical skills in using essential Unix commands and shell features.
2. Implement basic programming tasks by using Unix commands, shell scripts and system calls.
3. Analyze Unix internal commands to understand and optimize concurrency, scheduling and resource sharing.
4. Compare and evaluate Unix programming constructs by applying them to solve problems.
5. Design advanced Unix applications by integrating system calls to solve practical problems.

I Year M.Tech. CSE I Semester **L T P C**
Course Code: 531FA **2 0 0 2**

RESEARCH METHODOLOGY AND IPR

(Common to All Branches)

Course Objectives:

This course will

1. Familiarize students with research Methodology, Types and Approaches.
2. Enable students to design research frameworks and apply data collection methods.
3. Develop skills in research report writing, dissertation, and proposal drafting.
4. Impart knowledge on Intellectual Property Rights and Patent process.
5. Create awareness about copyrights and the International IPR framework.

UNIT 1: (~6 Lecture Hours)

Research Methodology: Objectives and Motivation of Research - Types of Research, Research Approaches - Significance of Research - Research Methods versus Methodology – Scientific Research Methods- Importance of Research Methodology - Research Process -
Criteria of Good Research

UNIT 2: (~6 Lecture Hours)

Research Design: Meaning of Research Design - Need of Research Design - Features of a Good Design - Concepts - Different Research Designs - Basic Principles of Experimental Design - Developing a Research Plan - Design of Experimental Set-up - Use of Standards and Codes - Data collection methods - Collection of Primary Data, Secondary data – Data Organization - Methods of Data Grouping - Diagrammatic and Graphic representation of Data

UNIT 3: (~6 Lecture Hours)

Research Report Writing: Format of the Research report – Synopsis – Dissertation- References/Bibliography/ Webliography. Research / Proposal Preparation: Writing a Research Proposal and Research Report - Writing Research Grant Proposal - Introduction to the use of software tools: Grammarly, Overleaf and References function in Microsoft word

UNIT 4: (~6 Lecture Hours)

Introduction to Intellectual Property rights and Patents Intellectual Property rights -Introduction – Features - Types of Intellectual property - Importance of Intellectual property rights. Patents - Concept of Patent – Duration – Patent Process – Patent searching – Procedure for filing of Patents - Ownership, Transfer, Assignment and Licensing of Patent

UNIT 5: (~7 Lecture Hours)

Copyrights and International Scenario Copyrights – Fundamentals of Copyright law - Originality of material- Rights of Reproduction - Rights to perform the work publicly - Copyright Ownership issues - Copyright registration - Notice of Copyright - Remedies for infringement in Copyrights. International Scenario: International Agencies - Treaties and Conventions related to Patents and Copy rights

Text Books:

1. C.R Kothari, Research Methodology, Methods & Technique, New Age International Publishers, 2004.
2. R. Ganesan, Research Methodology for Engineers, MJP Publishers, 2011.
3. Robert P. Merges, Peter S. Menell and Mark A. Lemley, Intellectual Property in New Technological Age, 2016.
4. T. Ramappa, Intellectual Property Rights Under WTO, S. Chand, 2008.
5. Satarkar, S.V., Intellectual property rights and copy right, ESS Publications, 2000.

Reference Books:

1. Ranjit Kumar, Research Methodology: A Step by Step Guide for beginners, 2nd edition, 2012.
2. Halbert, Resisting Intellectual Property, Taylor & Francis Ltd, 2007.

Online Resources:

1. https://onlinecourses.nptel.ac.in/noc23_ge36/preview(Course Title: Research Methodology, by Prof. Edamana Prasad, Prof. Prathap Haridoss (IIT Madras))
2. https://onlinecourses.swayam2.ac.in/ini25_hs10/preview (Course Title: Introduction to Research Methodology, by Prof. Mohammad Azher (Aligarh Muslim University))
3. https://onlinecourses.nptel.ac.in/noc20_hs55 (Course Title: Patent Law for Engineers and Scientists, by Dr. Feroz Ali (IIT Madras))
4. https://onlinecourses.nptel.ac.in/noc20_hs54 (Course Title: Patent Drafting for Beginners, by Dr. Feroz Ali (IIT Madras))

Course Outcomes:

After completion of the course, students will be able to

1. Describe research problem and outline the Research Design process.
2. Use appropriate software tools in developing structured research reports, proposals, and dissertations.
3. Analyse the role of Intellectual Property Rights and patent processes.
4. Evaluate copyright laws and international treaties governing intellectual property protection.
5. Generate research reports, dissertations and proposals.

I Year M.Tech. CSE II Semester **L T P C**
Course Code: 532BL **3 0 0 3**

ADVANCED ALGORITHMS
(Professional Core – III)

Prerequisites: Algorithm Design and Analysis

Course Objectives:
This course will
1. Introduce advanced methods of choosing, designing and analyzing algorithms.
2. Able to choose appropriate algorithms and use it for a specific problem.
3. Familiarize with basic paradigms and data structures used to solve advanced algorithmic problems.
4. Understand different classes of problems concerning their computation difficulties.

UNIT 1: (~8 Lecture Hours)
Sorting: Review of various sorting algorithms, topological sorting.
Graph: Definitions and Elementary Algorithms: Shortest path by BFS, Shortest path in edge-weighted case (Dijkasra's), depth-first search and computation of strongly connected component, emphasis on correctness proof of the algorithm and time/space analysis, example of amortized analysis

UNIT 2: (~8 Lecture Hours)
Matroids: Introduction to greedy paradigm, algorithm to compute a maximum weight maximal independent set. Application to MST
Graph Matching: Algorithm to compute maximum matching. Characterization of maximum matching by augmenting paths, Edmond's Blossom algorithm to compute augmenting path

UNIT 3: (~10 Lecture Hours)
Flow-Networks: Maxflow-Mincut theorem, Ford-Fulkerson Method to compute maximum flow, Edmond-Karp maximum-flow algorithm
Matrix Computations: Strassen's algorithm and introduction to divide and conquer paradigm, inverse of a triangular matrix, relation between the time complexities of basic matrix operations, LUP-decomposition

UNIT 4: (~10 Lecture Hours)
Shortest Path in Graphs: Floyd-Warshall algorithm and introduction to dynamic programming paradigm- Optimal Binary Search Tree, 0/1 Knapsack Problem, Longest Common Subsequence, Matrix Chain Multiplication
Modulo Representation of integers/polynomials: Chinese Remainder Theorem, Conversion between base- representation and modulo-representation. Extension to polynomials. Application: Interpolation problem

UNIT 5: (~9 Lecture Hours)
Linear Programming: Geometry of the feasibility region and Simplex algorithm.
NP-completeness: proof of NP-hardness and NP-completeness-Clique Problem, Vertex-Cover Problem, Subset-Sum Problem
Approximation algorithms: Introduction, Vertex-Cover Problem, Sum of Subsets

Text Books:

Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, Introduction to Algorithms, 3rd edition, The MIT Press, 2009.

Reference Books:

1. Alfred V. Aho, John E. Hopcroft and Jeffrey D. Ullman, The Design and Analysis of Computer Algorithms, Pearson Addison-Wesley Publication, originally published on 1974.
2. Jon Kleinberg and Eva Tardos, Algorithm Design, Pearson Addison-Wesley Publication, 2009.

Online Resources:

1. https://nptel.ac.in/courses/106104019/
2. http://www.nptelvideos.in/2012/11/data-structures-and-algorithms. html

Course Outcomes:

After completion of the course, students will be able to

1. Analyze and relate space and time complexity for various algorithmic paradigms.
2. Apply the suitable data structure for solving a problem using various strategies.
3. Differentiate the complexities of a problem solved in various approaches.
4. Evaluate various algorithmic design techniques.
5. Design appropriate mathematical notation to solve a problem using algorithmic paradigms solutions for real world problem.

I Year M.Tech. CSE II Semester **L T P C**
Course Code: 532BM **3 0 0 3**

COMPUTATIONAL INTELLIGENCE

(Professional Core – IV)

Course Objectives:

This course will enable the students to

1. Introduce the principles and scope of Computational Intelligence(CI).
2. Understand key CI techniques and their components.
3. Develop skills to design and apply CI models for problem-solving.
4. Explore hybrid approaches and practical applications of CI.

UNIT 1: (~ 8 Lecture Hours)

Introduction: Computational Intelligence-Emergence, Definition, Soft Computing, Fundamentals of Soft Computing, Computational Learning theory, Synergism in Soft computing, Comparison of CI, AI, and Soft Computing, Applications of Computational Intelligence

UNIT 2: (~ 10 Lecture Hours)

Artificial Neural Networks: Perceptron, Multi-Layer Perceptron (MLP), Backpropagation algorithm, Activation functions, Regularization, Self-Organizing Maps, Hopfield Networks, Radial Basis Function Networks, Introduction to Deep Learning-Overview of CNN, RNN

UNIT 3: (~ 9 Lecture Hours)

Fuzzy sets: Fuzzy sets, Membership functions, Discontinuous and Continuous membership functions, Typical membership functions, operations on Fuzzy sets, Fuzzy relations, Fuzzy logic, The Composition rule of inference, Fuzzy pattern Recognition-clustering, Image segmentation using Fuzzy C-means

UNIT 4: (~ 10 Lecture Hours)

Introduction to Evolutionary Computing: The Chromosome, Fitness function, Initial Population, Selection operators, Genetic Algorithms, Applications of GA in Optimization, Machine Learning, Intelligent search, Genetic Programming

UNIT 5: (~ 8 Lecture Hours)

Swarm Intelligence: Particle Swarm Optimization, Ant Colony Optimization, Hybrid CI systems: Neuro-Fuzzy, GA+NN

Case Studies: Fuzzy models for face matching and mood detection, Computational Intelligence in mobile robotics

Text Books:

1. A. Konar, Computational Intelligence: Principles, Techniques and Applications. Berlin, Germany, Springer, 2005.
2. A. P. Engelbrecht, Computational Intelligence: An Introduction, 2nd Edition. Hoboken, NJ, USA: Wiley, 2007.

Reference Books:

1. J. S. R. Jang, C. T. Sun, and E. Mizutani, Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence. Upper Saddle River, NJ, USA: Prentice-Hall, 1997.
2. S. N. Sivanandam and S. N. Deepa, Principles of Soft Computing, 2nd edition, New Delhi, India: Wiley India, 2011.

3. LiMinFu, Neural Networks in Computer Intelligence, Tata McGraw-Hill Edition, 2003.

Online Resources:

1. https://nptel.ac.in/courses/106105173
2. https://www.coursera.org/learn/neural-networks-deep-learning
3. https://nptel.ac.in/courses/112/107/112107209/
4. https://github.com/krzjoa/awesome-computational-intelligence

Course Outcomes:

After completion of the course, students will be able to

1. Describe the basic concepts, scope, and components of Computational Intelligence.
2. Explain the working principles of Neural Networks, Fuzzy Systems, Evolutionary Computing and Swarm Intelligence.
3. Apply suitable Computational Intelligence techniques to solve engineering, optimization, and decision-making problems.
4. Analyze and compare different Computational Intelligence techniques for solving optimization and decision-making problems.
5. Design and evaluate hybrid CI models and apply them to real-world applications such as pattern recognition.

I Year M.Tech. CSE II Semester **L T P C**
Course Code: 532BN **3 0 0 3**

ENTERPRISE CLOUD CONCEPTS

(Professional Elective - III)

Course Objectives:
This course will enable the students to

1. Acquire knowledge in understanding the significance of cloud computing and its impact on enterprises.
2. Understand the fundamental concepts, service models, and deployment models of cloud computing.
3. To analyze cloud-enabled smart enterprises for improved agility and efficiency.
4. Knowledge in transforming traditional enterprises into cloud-centric organizations.

UNIT 1: (~ 9 Lecture Hours)
Understanding Cloud Computing: Origins and influences, Basic Concepts and Terminology, Goals and Benefits, Risks and Challenges
Fundamental Concepts and Models: Roles and Boundaries, Cloud Characteristics, Cloud Delivery Models, Cloud Deployment Models

UNIT 2: (~ 8 Lecture Hours)
Cloud-Enabling Technology:
Broadband Networks and Internet Architecture, Data Center Technology, Virtualization Technology
Cloud Computing Mechanisms:
Cloud Infrastructure Mechanisms: Logical Network Perimeter, Virtual Server, Cloud Storage Device, Cloud Usage Monitor, Resource Replication

UNIT 3: (~ 9 Lecture Hours)
Cloud Management Mechanisms: Remote Administration System, Resource Management System, SLA Management System, Billing Management System, Case Study Example
Cloud Computing Architecture:
Fundamental Cloud Architectures: Workload Distribution Architecture, Resource Pooling Architecture, Dynamic Scalability Architecture, Elastic Resource Capacity Architecture, Service Load Balancing Architecture, Cloud Bursting Architecture, Elastic Disk Provisioning Architecture, Redundant Storage Architecture, Case Study Example

UNIT 4: (~ 8 Lecture Hours)
Cloud-Enabled Smart Enterprises
Introduction, Revisiting the Enterprise Journey, Service-Oriented Enterprises, Cloud Enterprises, Smart Enterprises, The Enabling Mechanisms of Smart Enterprises
Cloud-Inspired Enterprise Transformations
Introduction, The Cloud Scheme for Enterprise Success, Elucidating the Evolving Cloud Idea, Implications of the Cloud on Enterprise Strategy, Establishing a Cloud-Incorporated Business Strategy

UNIT 5: (~ 8 Lecture Hours)

Transitioning to Cloud-Centric Enterprises

The Tuning Methodology, Contract Management in the Cloud.

Cloud-Instigated IT Transformations

Introduction, Explaining Cloud Infrastructures, A Briefing on Next-Generation Services, Service Infrastructures, Cloud Infrastructures, Cloud Infrastructure Solutions, Clouds for Business Continuity, The Relevance of Private Clouds, The Emergence of Enterprise Clouds.

Text Books:

1. Erl Thomas, Puttini Ricard and Mahmood Zaigham, Cloud Computing, Concepts, Technology & Architecture 1st edition.
2. Pethuru Raj, Cloud Enterprise Architecture, CRC Press.

Reference Books:

James Bond, The Enterprise Cloud, O'Reilly Media, Inc.

Online Resources:

1. https://www.purrweb.com/blog/enterprise-cloud-computing-soar-to-the-heights
2. https://brainhub.eu/library/enterprise-cloud-computing

Course Outcomes:

After completion of the course, students will be able to

1. Understand the origins, fundamental, and challenges of cloud computing.
2. Apply cloud-enabling technologies for effective resource management.
3. Analyze cloud management mechanisms and evaluate architectures in real-world case studies.
4. Assess how cloud computing enables smart enterprises and drives the transformation of traditional enterprises into cloud-centric organizations.
5. Design strategies to transition enterprises into cloud-centric models.

I Year M.Tech. CSE II Semester **L T P C**
Course Code: 532BP **3 0 0 3**

CYBER SECURITY

(Professional Elective – III)

Prerequisites: Cryptography and Network Security

Course objectives:
This course will enable the students to

1. Understand various types of cyber-attacks and cyber-crimes.
2. Learn threats and risks within context of the cyber security.
3. Having an overview of the cyber laws & concepts of cyber forensics.
4. Study the defensive techniques against these attacks.

UNIT 1: (~9 Lecture Hours)
Introduction to Cyber Security: Basic Cyber Security Concepts, layers of security, Vulnerability threat, Harmful acts, Internet Governance – Challenges and Constraints, Computer Criminals, CIA Triad, Assets and Threat, motive of attackers, active attacks, passive attacks, Software attacks, hardware attacks, Cyber Threats-Cyber Warfare, Cyber Crime, Cyber terrorism, Cyber Espionage, etc., Comprehensive Cyber Security Policy

UNIT 2: (~9 Lecture Hours)
Cyberspace and the Law & Cyber Forensics: Introduction, Cyber Security Regulations, Roles of International Law. The INDIAN Cyberspace, National Cyber Security Policy. 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, Forensics Investigation, Challenges in Computer Forensics

UNIT 3: (~9 Lecture Hours)
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, Organizational security Policies and Measures in Mobile Computing Era, Laptops

UNIT 4: (~9 Lecture Hours)
Cyber Security: Organizational Implications: Introduction, cost of cybercrimes and IPR issues, web threats for organizations, security and privacy implications, social media marketing: security risks and perils for organizations, social computing and the associated challenges for organizations

UNIT 5: (~9 Lecture Hours)
Privacy Issues: Basic Data Privacy Concepts: Fundamental Concepts, Data Privacy Attacks, Data linking and profiling, privacy policies and their specifications, privacy policy languages, privacy in different domains- medical, financial, etc.
Cybercrime: Examples and Mini-Cases
Examples: Official Website of Maharashtra Government Hacked, Indian Banks Lose Millions of Rupees, Parliament Attack, Pune City Police Bust Nigerian Racket, e-mail spoofing instances.
Mini-Cases: The Indian Case of online Gambling, An Indian Case of INTELLECTUAL Property Crime, Financial Frauds in Cyber Domain

Text Books:

1. Nina Godbole and Sunit Belpure, Cyber Security Understanding Cyber Crimes, Computer Forensics and Legal Perspectives, Wiley.
2. B.B. Gupta, D.P. Agrawal and Haoxiang Wang, Computer and Cyber Security: Principles, Algorithm, Applications, and Perspectives, CRC Press, ISBN 9780815371335, 2018.

Reference Books:

1. James Graham, Richard Howard and Ryan Otson, Cyber Security Essentials, CRC Press.
2. Chwan-Hwa(john) Wu and J. David Irwin, Introduction to Cyber Security, CRC Press T&F Group.

Online Resources:

1. Introduction to Cyber Security: More free resources from The Open University | OpenLearn - Open University
2. https://onlinecourses.swayam2.ac.in/cec20_cs15/preview
3. https://www.cybrary.it/course/introduction-to-it-and-cybersecurity/
4. https://www.cert-in.org.in/

Course Outcomes:

After completion of the course, students will be able to

1. Analyze and evaluate the cyber security needs of an organization.
2. Interpret national and international cyber security regulations, legal frameworks, and policies to assess their impact on organizational practices.
3. Design and develop security architectures and policies to safeguard organizational assets from cybercrimes and attacks.
4. Investigate security challenges in mobile and wireless devices, and recommend organizational measures to mitigate associated risks.
5. Understand and evaluate fundamental concepts of data privacy, privacy attacks, policies, and apply them across various domains through real-world case studies.

I Year M.Tech. CSE II Semester **L T P C**
Course Code: 532BQ **3 0 0 3**

PARALLEL COMPUTING
(Professional Elective – III)

Course Objectives:
This course will enable the students to
1. Understand the foundations of parallel Computing.
2. Apply various parallel computing architectures and programming models.
3. Develop the principles and design of parallel Algorithms.
4. Gain knowledge of writing efficient parallel programs

UNIT 1: (~ 8 Lecture Hours)
Parallel Computing: Introduction, Motivation and scope - Parallel Programming Platforms – Basic Communication Operations.

UNIT 2: (~ 10 Lecture Hours)
Principles of Parallel Algorithm Design - Analytical Modelling of Parallel Programs.

UNIT 3: (~ 8 Lecture Hours)
Programming using Message Passing Paradigm (MPI) – Programming Shared Address Space Platforms (PThreads) .

UNIT 4: (~ 10 Lecture Hours)
Dense Matric Algorithms (Matrix-Vector Multiplication, Matrix-Matrix Multiplication) – Sorting Algorithms (Issues, Bubble Sort, Quick Sort, Bucket Sort, Enumeration Sort, Radix Sort).

UNIT 5: (~ 9 Lecture Hours)
Graph Algorithms (Minimum Spanning Tree: Prim's Algorithm - Single-Source Shortest Paths: Dijkstra's Algorithm) Search Algorithms (DFS, BFS).

Text Books:

Ananth Grama, George Karypis, Vipin Kumar, Anshul Gupta,Introduction to Parallel Computing, 2nd edition , Addison-Wesley, 2003.

Reference Books:
1. Michaek J. Quinn, Parallel Computing – Theory and Practice, 2nd edition, Tata McGrawHill Edition.
2. V. Rajaraman, C. Siva Ram Murthy, Parallel Computers – Architectures and Programming, PHI.

Online Resources:

https://nptel.ac.in/courses/106102163

Course Outcomes:

After completion of the course, students will be able to

1. Understand the concepts of parallel architectures
2. Apply the principles and design parallel algorithms
3. Apply data structures that efficiently model the information in a problem.
4. Develop an efficient parallel algorithm to solve it.
5. Implement an efficient and correct code to solve it, analyze its performance

I Year M.Tech. CSE II Semester **L T P C**
Course Code: 532BR **3 0 0 3**

LARGE LANGUAGE MODELS
(Professional Elective – III)

Course Objectives:
This course will enable the students to
1. Introduce the foundations of transformer architectures and their evolution into LLMs.
2. Equip students with skills to train, fine-tune, and deploy LLMs for various tasks.
3. Explore ethical, legal, and societal implications of LLMs in real-world applications.
4. Expose students to state-of-the-art LLM frameworks, evaluation techniques, and research trends.

UNIT 1: (~8 Lecture Hours)
Foundations of Large Language Models: Introduction to LLMs: Definition, scope, and historical evolution from statistical NLP to transformers. The Transformer architecture: Attention mechanisms, self-attention, multi-head attention. Pretraining objectives: Masked language modeling (MLM), Causal language modeling (CLM). Evolution of LLMs: BERT, GPT series, T5, LLaMA, Mistral

UNIT 2: (~10 Lecture Hours)
Training and Fine-Tuning LLMs: Pretraining datasets and tokenization: BPE, Sentence Piece, Word Piece. Fine-tuning approaches: Full fine-tuning, LoRA, adapters, instruction tuning. Domain adaptation and few -shot/zero-shot learning. Data augmentation for LLMs and prompt-based tuning

UNIT 3: (~8 Lecture Hours)
Prompt Engineering and Applications: Principles of prompt design: Zero-shot, few-shot, and chain-of-thought prompting. System prompts, role prompting, and context length optimization.
Use cases: Text generation, summarization, code generation, question answering, chatbots. Tools & frameworks: Lang Chain, Llama Index, Hugging Face Transformers

UNIT 4: (~10 Lecture Hours)
Evaluation and Deployment of LLMs: Evaluation metrics: Perplexity, BLEU, ROUGE, METEOR, human evaluation. Benchmark datasets: GLUE, Super GLUE, HELM, BIG-bench.
Deployment strategies: API-based deployment, on-prem deployment, inference optimization. Scaling and latency considerations; quantization and pruning for LLMs

UNIT 5: (~9 Lecture Hours)
Ethics, Safety, and Future Directions: Bias, fairness, and toxicity in LLMs. Hallucination problem and mitigation techniques. Legal and regulatory issues: Copyright, data privacy, AI Act. Trends in LLM research: Multimodal LLMs, retrieval-augmented generation (RAG), open-source LLM ecosystems

Text Books:
1. Jay Alammar, Maarten Grootendorst, Hands on Large Language Models by O'Reilly, 2023
2. Thushan Ganegedara, Natural Language Processing with Tensor Flow, by Packt Publishing, 2nd edition 2022.

Reference Books:

1. David Foster, Generative Deep Learning, O'Reilly, 2020.
2. Lewis Tunstall, Leandro von Werra and Thomas Wolf, Natural Language Processing with Trans- formers, 2022.
3. Sebastian Raschka, Build a Large Language Model (From Scratch), ISBN 9781633437166.

Online Resources:

1. Mixture of Experts: Mixture of Experts Explained (huggingface.co)
2. Various benchmarks to evaluate LLMs: LLM Benchmarks: Understanding Language Model Performance (humanloop.com)
3. Types of attention mechanism: Understanding and Coding the Self-Attention Mechanism of Large Language Models From Scratch (sebastianraschka.com)

Course Outcomes:

After completion of the course, students will be able to

1. Understand the architecture and inner workings of transformer-based LLMs.
2. Identify challenges in LLM training, deployment, and scaling.
3. Apply prompt engineering and fine-tuning techniques for domain-specific tasks.
4. Analyze ethical, legal, and societal implications of LLM usage.
5. Evaluate LLM performance using standard metrics and benchmarks.

I Year M.Tech. CSE II Semester **L T P C**
Course Code: 532BS **3 0 0 3**

BIOINFORMATICS
(Professional Elective - IV)

Prerequisites: Machine Learning

Course Objectives:
This course will enable the students to
1. Explain the central dogma of molecular biology and demonstrate the use of XML and related technologies (DTD, Schema) in representing biological data.
2. Apply Perl programming techniques to manipulate biological sequences.
3. Describe the architectures, interfaces, and roles in bioinformatics data management.
4. Implement sequence alignment and phylogenetic analysis methods.

UNIT 1: (~10 Lecture Hours)
The Central Dogma & XML (Bio XML) for Bioinformatics: Watson's definition, information flow, from data to knowledge, Convergence, the organization of DNA, the organization of Proteins, Introduction, Differences between HTML and XML, fundamentals of XML, fundamentals of XML namespaces. Introduction to DTDs, Document Type Declarations, declaring elements, declaring attributes, working with entities XML Schemas, Essential Concepts, working with simple types, working with complex types, Basic namespaces issues

UNIT 2: (~8 Lecture Hours)
Perl (Bioperl) for Bioinformatics: Representing sequence data, program to store a DNA sequence, concatenating DNA fragments, Transcription, Calculating the reverse complement in Perl, Proteins, files, reading proteins in files, Arrays, Flow control, finding motifs, counting Nucleotides, exploding strings into arrays, operating on strings, writing to files, subroutines and bugs

UNIT 3: (~8 Lecture Hours)
Databases: Flat file, Relational, object oriented databases, Object Relational and Hypertext, Data life cycle, Database Technology, Database Architecture, Database Management Systems and Interfaces

UNIT 4: (~8 Lecture Hours)
Sequence Alignment Algorithms: Biological motivations of sequence analysis, the models for sequence analysis and their biological motivation, global alignment, local alignment, End free-space alignment and gap penalty, Sequence Analysis tools and techniques

UNIT 5: (~8 Lecture Hours)
Phylogenetic Analysis: Introduction, methods of Phylogenetic analysis, distance methods, the neighbor- Joining (NJ) method, The Fitch / Margoliash method, character-based methods, Other methods, Tree evaluation and problems in phylogenetic analysis, Clustering, Protein structure visualization and Protein structure prediction

Text Books:
1. S.C. Rastogi amd N. Mendiratta, Bioinformatics Methods and Applications, CBS publications, 2004.
2. James D. Tisdall, Beginning Perl for Bioinformatics, O'Reilly media, 1st edition, 2001.

Reference Books:

1. D.R. Westhead and J.H. Parish, Bioinformatics, Viva books private limited, New Delhi (2003).
2. Att Wood, Bioinformatics, Pearson Education, 2004.
3. Bryan Bergeron and M.D, Bioinformatics Computing, Pearson Education, 2003.

Online Resources:

1. https://www.ncbi.nlm.nih.gov/education/
2. https://www.bioinformatics.org/

Course Outcomes:

After completion of the course, students will be able to

1. Understand the Central Dogma & XML (Bio XML) for Bioinformatics.
2. Analyze Perl (Bioperl) for Bioinformatics.
3. Illustrate Databases technology, architecture and its interfaces.
4. Understand Sequence Alignment Algorithms, Phylogenetic Analysis.
5. Apply protein structure visualization and prediction tools to interpret biological data in the context of phylogenetic analysis.

I Year M.Tech. CSE II Semester **L T P C**
Course Code: 532BT **3 0 0 3**

ADHOC AND SENSOR NETWORKS

(Professional Elective – IV)

Course Objectives:

This course will enable the students to

1. Understand the design issues in ad hoc and sensor networks.
2. Be familiar with different types of adhoc routing protocols.
3. Exposed to the TCP issues in adhoc networks.
4. Learn the architecture and protocols of wireless sensor networks.

UNIT 1: (~10 Lecture Hours)

Introduction to Ad Hoc Wireless Networks: Characteristics of MANETS, Applications of MANETS, Challenges

Routing in Ad hoc Networks: Topology based versus position based approaches, Topology based routing protocols, and position based routing, other routing protocols

UNIT 2: (~10 Lecture Hours)

Data Transmission in MANETS: The broadcast storm, Multicasting, Geocasting

TCP Over Ad Hoc Networks: TCP protocol overview, TCP and MANETS, Solutions for TCP over Ad Hoc

UNIT 3: (~9 Lecture Hours)

Basics of Wireless Sensors and Applications: Introduction, Empirical Energy Consumption, Sensing and Communication Range, Design Issues, Clustering of Sensors

Data Retrieval In Sensor Networks: Classification of WSNs, Routing Layer: Flat Based Routing-Directed diffusion(DD),Sequential Assignment Routing(SAR),Minimum Cost Forwarding Algorithm(MCFA),Hierarchical Routing: Cluster Based Routing Protocol(CBRP), Adaptive Routing: Low energy Adaptive Clustering Hierarchy(LEACH), Threshold sensitive Energy Efficient(TEEN), Adaptive Periodic TEEN(APTEEN), SPIN based Adaptive Routing, Power Efficient Gathering in Sensor Information Systems (PEGASIS), High-Level Application Layer Support

UNIT 4: (~8 Lecture Hours)

MAC Protocols for wireless sensor networks – Low duty cycle protocols and wakeup concepts – Contention- Based protocols – Schedule-Based protocols – IEEE 802.15.4 Zigbee – Topology control Routing protocols

Security: Security in Ad Hoc Wireless Networks, Key Management, Secure Routing, Cooperation in MANETs, Intrusion Detection Systems

UNIT 5: (~8 Lecture Hours)

Sensor Network Platforms and Tools: Sensor Network Hardware. Sensor Network Programming Challenges and Node-Level Software Platforms

Operating System-Tiny OS: Imperative Language: nesC, Data flow style language: Tiny GALS, Node- Level Simulators, NS-2 and its sensors network extension, TOSSIM

Text Books:

1. Carlos de Morais Cordeiro and Dharma Prakash Agrawal, Ad Hoc and Sensor Networks: Theory and Applications, World Scientific Publications/Cambridge University Press, 2006.
2. Feng Zhao and Leonidas Guibas, Wireless Sensor Networks: Information Processing Approach, Elsevier Science Imprint, Morgan Kauffman Publishers, 2005.

Reference Books:

1. C. Siva Ram Murthy and B. S. Manoj, Ad Hoc Wireless Networks: Architectures and Protocols, Pearson Education, 2004.
2. Sudip Misra, Isaac Woungang, and Subhas Chandra Misra, Guide to Wireless Ad Hoc Networks, Springer International Edition, 2011.
3. Sudip Misra, Isaac Woungang, and Subhas Chandra Misra, Guide to Wireless Sensor Networks, Springer International Edition, 2012.

Online Resources:

1. Wireless Adhoc and Sensor Networks - Course (nptel.ac.in)
2. Wireless Ad Hoc and Sensor Networks | (iitk.ac.in)
3. Advanced Network Technologies Virtual Lab — IIT Kharagpur (iitkgp.ac.in)

Course Outcomes:

After completion of the course, students will be able to

1. Understand the concepts, network architectures and applications of ad hoc and wireless sensor networks.
2. Analyze the protocol design issues of ad hoc and sensor networks.
3. Design routing protocols for ad hoc and wireless sensor networks with respect to some protocol design issues.
4. Evaluate the QoS related performance measurements of ad hoc and sensor networks.
5. Differentiate the different operating Systems for various mobile and sensor networks.

I Year M.Tech. CSE II Semester **L T P C**
Course Code: 532BU **3 0 0 3**

ROBOTIC PROCESS AUTOMATION

(Professional Elective – IV)

Prerequisites: Database Management Systems**,** Compu**t**er Ne**t**works

Course Objectives:

This course will enable the students to

1. Introduce the fundamentals of Robotic Process Automation.
2. Familiarize with Automation Anywhere Enterprise Platform.
3. Develop skills in using Automation Anywhere tools.
4. Apply automation techniques to business scenarios.

UNIT 1: (~ 7 Lecture Hours)

Introduction to Robotic Process Automation & Bot Creation Introduction to RPA and Use cases – Automation Anywhere Enterprise Platform – Advanced features and capabilities – Ways to create Bots

UNIT 2: (~ 8 Lecture Hours)

Web Control Room and Client Introduction - Features Panel - Dashboard (Home, Bots, Devices, Audit, Workload, Insights) - Features Panel – Activity (View Tasks in Progress and Scheduled Tasks)- Bots (View Bots Uploaded and Credentials)

UNIT 3: (~ 9 Lecture Hours)

Devices (View Development and Runtime Clients and Device Pools) - Workload (Queues and SLA Calculator) - Audit Log (View Activities Logged which are associated with Web CR) – Administration (Configure Settings, Users, Roles, License and Migration) - Demo of Exposed API's – Conclusion – Client introduction and Conclusion

UNIT 4: (~ 9 Lecture Hours)

Bot Creator Introduction – Recorders – Smart Recorders – Web Recorders – Screen Recorders - Task Editor – Variables - Command Library – Loop Command – Excel Command – Database Command - String Operation Command - XML Command

UNIT 5: (~ 9 Lecture Hours)

Terminal Emulator Command - PDF Integration Command - FTP Command - PGP Command - Object Cloning Command - Error Handling Command - Manage Windows Control Command - Workflow Designer - Report Designer

Text Books:

Alok Mani Tripathi, Learning Robotic Process Automation, Create Software robots and automate business processes with the leading RPA tool – UiPath, Kindle Edition.

Reference Books:

Tom Taulli, Robotic Process Automation Handbook, 2020 Edition, Kindle Edition.

Online Resources:

1. https://www.uipath.com/rpa/academy
2. https://www.coursera.org/specializations/roboticprocessautomation
3. https://www.simplilearn.com/basics-of-automation-anywhere-free-course-skillup

Course Outcomes:

After completion of the course, students will be able to

1. Describe the fundamental concepts of Robotic Process Automation (RPA) and illustrate its real-world applications and industry use cases**.**
2. Demonstrate the use of Automation Anywhere Enterprise Platform by creating, configuring, and executing simple bots for automation tasks.
3. Compare and analyze different recorders, commands, and bot creation methods to determine suitable approaches for specific automation scenarios.
4. Evaluate and manage devices, workloads, audit logs, and administrative controls in the Web Control Room to ensure effective bot governance and security.
5. Design, develop, and optimize end-to-end automation workflows using advanced commands.

I Year M.Tech. CSE II Semester **L T P C**
Course Code: 532BV **3 0 0 3**

GENERATIVE AI
(Professional Elective – IV)

Prerequisites: Artificial Intelligence

Course Objectives:
This course will enable the students to
1. Introduce the foundations, evolution, and core concepts of AI, ML, DL, NLP, and Generative AI.
2. Understand the advanced neural architectures and generative models such as GANs, VAEs, and Transformers.
3. Explore Large Language Models, prompt engineering, and their real-world applications.
4. Familiarize learners with frameworks, multimodal applications, and ethical considerations in Generative AI.

UNIT 1: (~ 10 Lecture Hours)
Foundations of AI and Generative Models: Introduction and historical evolution to Artificial Intelligence (AI), Machine Learning (ML), Natural Language Processing (NLP)and Deep Learning (DL), Structure of Artificial Neural Networks (ANNs), Mathematical and computational foundations of generative modeling, Overview of generative models and their applications across various domains; Importance of Generative AI in modern applications, Transfer learning and in advancing Generative AI

UNIT 2: (~ 9 Lecture Hours)
Advanced Neural Architectures for Generative AI: Variational Auto encoders (VAEs): principles and applications, Generative Adversarial Networks (GANs): architecture and working principles; Transformer architecture and attention mechanisms (in detail); Long Short-Term Memory Networks (LSTMs) and the limitations of traditional RNNs/LSTMs, Advanced Transformer architectures and techniques, Pre-training and transfer learning strategies for generative models

UNIT 3: (~ 9 Lecture Hours)
Large Language Models and Prompt Engineering: Overview of Large Language Models (LLMs), GPT architecture, variants, and working principles, Pré-training and fine-tuning GPT models for applications (e.g., chatbots, text generation), Case study: GPT-based customer support chatbot, BERT architecture, pre-training objectives, and fine-tuning, Prompt Engineering: Designing effective prompts, controlling model behavior, and improving output quality, Fine-tuning language models for creative writing and chatbot development

UNIT 4: (~ 9Lecture Hours)
Multi-Agent Systems and Generative AI Applications: Introduction to Multi-Agent Systems (MAS),Types of agents: reactive, deliberative, hybrid, and learning agents, Multi-agent collaboration and orchestration for generative tasks, Use cases: autonomous research assistants, cooperative creative generation, distributed problem-solving, Frameworks and tools: AutoGen, CrewAI, Hugging GPT for LLM-powered multi-agent systems, Generative AI applications: Art, Creativity, Image/Video generation, Music composition, Healthcare, Finance, Real-world case studies and deployment challenges

UNIT 5: (~ 9 Lecture Hours)
Frameworks, Multimodal Applications, and Ethics: Lang Chain framework: components and LLM application development, Retrieval-Augmented Generation (RAG), Embeddings, Indexing networks, and Vector databases, Generative AI across modalities: Text, Code, Image, and Video generation, Image and Video generation using GANs and VAEs, Multimodal Generative AI: integration and training strategies, Ethical considerations: bias, fairness, trust, and responsible AI deployment, Social and legal implications of Generative AI, Risk mitigation strategies and real-world ethical case studies

Text Books:

1. Altaf Rehmani, Generative AI for Everyone, Understanding the Essentials and Applications of this Breakthrough Technology.
2. Charu C. Aggarwal, Neural Networks and Deep Learning, A Textbook. Joseph Babcock, Raghav Bali, Generative AI with Python and TensorFlow 2, 2024.

Reference Books:

1. Josh Kalin, Generative Adversarial Networks Cookbook, Packt Publishing Limited.
2. Jesse Sprinter, Generative AI in Software Development: Beyond the Limitations of Traditional Coding, 2024.

Online Resources:

1. https://onlinecourses.nptel.ac.in/noc25_cs161/preview
2. https://onlinecourses.nptel.ac.in/noc25_cs159/preview.

Course Outcomes:
After completion of the course, students will be able to

1. Demonstrate knowledge of AI foundations, generative models, and advanced neural architectures.
2. Apply generative AI techniques to create solutions for text, image, video, and multimodal tasks.
3. Design, fine-tune, and optimize Large Language Models for specific applications.
4. Evaluate ethical, social, and legal implications of Generative AI deployments and propose
5. Develop and deploy generative AI-based applications by leveraging state-of-the-art frameworks, libraries, and APIs to solve real-world problems effectively.

I Year M.Tech. CSE II Semester **L T P C**
Course Code: 53216 **0 0 4 2**

ADVANCED ALGORITHMS LAB

(Professional Core – III)

Prerequisites: Algorithm Design and Analysis

Course Objectives:

This course will enable the students to

1. Write and execute programs in Java to implement advanced algorithms.
2. Choose an appropriate design paradigm to solve problems.
3. Built up strong skills to implement algorithms for various problems.
4. Familiarize with efficient utilization of programming language constructs while implementing algorithms along with their complexities.
5. Able to understand strategies to solve real time problems.

List of Experiments:

Week 1:

Implementation of Sorting- heap sort, quick sort, topological sort using queue.

Week 2:

Implement Graph traversal algorithms – BFS, DFS

Week 3:

Implement Single Source Shortest Paths (Dijkstra's and Bellman Ford)

Week 4:

Implement Minimum Cost Spanning Trees (Prims and Kruskals).

Week 5:

Implement Edmond Blossom algorithm.

Week 6:

Implement Network Ford Fulkerson and Edmond Karp Method.

Week 7:

Implement LUP decomposition method.

Week 8:

Implement Floyd Warshall algorithm and Longest common subsequence.

Week 9:

Implement OBST, Matrix chain Multiplication .

Week 10:

Implement Simplex Algorithm.

Textbooks:

1. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein, Introduction to Algorithms, 3rd edition, MIT Press. 2009.
2. Mark Allen Weiss, Data Structures and Algorithm Analysis in C++, 2nd edition, Pearson, 2004.

Reference Books:

1. S.N.Sivanandam and S.N.Deepa, Principles of Soft Computing 2nd edition, Wiley Publication 2016.
2. Satish Kumar, Neural Networks -A classroom approach, 2nd edition, TMH, 2017.

Online Resources:

1. https://www.geeksforgeeks.org/top-algorithms-and-data-structures-for-competitive-programming/
2. http://www. nptelvideos. com/java/java _ video _ Lecture Hours_tutorials.php
3. https://nptel.ac.in/courses/106104019/

Course Outcomes:

After completion of the course, students will be able to

1. Solve problems related to divide and conquer strategy.
2. Implement various applications related to graphs and flow networks
3. Analyze and implement advanced sorting techniques.
4. Develop the dynamic programming algorithms and analyze.
5. Evaluate different designing methods for development of algorithms to realistic problems.

I Year M.Tech. CSE II Semester **L T P C**
Course Code: 53217 **0 0 4 2**

ENTERPRISE CLOUD CONCEPTS LAB

(Professional Elective - III)

Course Objectives:

This course will enable the students to

1. Acquire knowledge in setting up virtualization platforms and installing different operating systems.
2. Develop and deploying simple applications on cloud service platforms.
3. Get knowledge in managing cloud resources through file transfer, VM provisioning, and service utilization.
4. Configure and execute distributed applications using Hadoop.

List of Programs:

Week 1:

Install VirtualBox/VMware Workstation and set up a Linux operating system on top of Windows 7/8 host.

Week 2:

Install VirtualBox/VMware Workstation and set up a Windows operating system on top of Windows 7/8 host.

Week 3:

Install a C compiler in a Linux virtual machine created using VirtualBox/VMware and execute simple programs.

Week 4:

Install a C compiler in a Windows virtual machine created using VirtualBox/VMware and execute simple programs.

Week 5:

Install Google App Engine and create a "Hello World" application using **Python**, along with other simple web applications.

Week 6:

Install Google App Engine and create a "Hello World" application using **Java**, along with other simple web applications.

Week 7:

Transfer files between two virtual machines using **shared folders / drag-and-drop features** of VirtualBox/VMware.

Week 8:

Transfer files between two virtual machines using **network-based methods** (e.g., SCP / FTP in Linux or file sharing in Windows).

Week 9:

Register on Trystack (Online OpenStack Demo), configure the environment, and launch a virtual machine instance.

Week 10:

Manage the launched instance on Trystack by attaching storage, configuring networking, and accessing the VM remotely.

Week 11:

Install and configure a Hadoop single-node cluster.

Week 12:

Run simple Hadoop applications such as **WordCount** (and optionally Grep or Sort) on the single-node cluster.

Textbooks:

1. Rajkumar Buyya, Christian Vecchiola and S. ThamaraiSelvi, Mastering Cloud Computing: Foundations and Applications Programming, 1st edition, McGraw Hill Education, 2013.
2. ArshdeepBahga and Vijay Madisetti, Cloud Computing, A Hands-On Approach, 1st edition, Universities Press, 2014.

Reference Books:

Thomas Erl, Zaigham Mahmood, and Ricardo Puttini, Cloud Computing: Concepts, Technology & Architecture, 1st edition, Pearson Education, 2013.

Online Resources:

1. https://www.iitk.ac.in/nt/faq/vbox.htm
2. https://www.google.com/urlsa=t&rct=j&q=&esrc=s&source=web&cd=&ved=2ahUKEwjqrNG0za73AhXZt1YBHZ21DWEQFnoECAMQAQ&url=http%3A%2F%2Fwww.cs.columbia.edu%2F~sedwards%2F classes%2F2015%2F1102-fall%2Flinuxvm.pdf&usg=AOvVaw3xZPuF5xVgk-AQnBRsTtHz
3. https://www.cloudsimtutorials.online/cloudsim/
4. https://edwardsamuel.wordpress.com/2014/10/25/tutorial-creating-openstack-instance-in-trystack/
5. https://www.edureka.co/blog/install-hadoop-single-node-hadoop-cluster

Course Outcomes:

After completion of the course, students will be able to

1. Demonstrate the installation and configuration of virtualization platforms and operating systems for cloud environments.
2. Apply programming tools and compilers within virtualized systems to develop and execute basic applications.
3. Develop and deploy simple applications using cloud service platforms such as Google App Engine.
4. Illustrate and analyze techniques for resource sharing and communication between virtual machines in a cloud environment.
5. Configure and operate cloud infrastructure components such as OpenStack and Hadoop to execute and evaluate distributed applications.

I Year M.Tech. CSE II Semester **L T P C**
Course Code: 53218 **0 0 4 2**

CYBER SECURITY LAB
(Professional Elective - III)

Prerequisites: Cryptography and Network Security

Course Objectives:
This course will enable the students to
1. Get hands-on exposure to various cyber security threats, attacks, and defense mechanisms.
2. Gain practical knowledge of cyber forensic tools for evidence collection and analysis.
3. Familiarize with cryptographic techniques, password security, and secure communication mechanisms.
4. Develop analytical and problem-solving skills for investigating and mitigating real-world cyber security incidents.

List of Experiments

Week 1:
Perform an Experiment for port scanning with NMAP.

Week 2:
Setup a honeypot and monitor the honeypot on the network.

Week 3:
Install Jcrpt /Cryptool tool (or any other equivalent) and demonstrate Asymmetric, Symmetric crypto algorithm, Hash and Digital/PKI signatures.

Week 4:
Generate minimum 10 passwords of length 12 characters using open SSL command.

Week 5:
Perform practical approach to implement Foot printing-Gathering target information using Dmitry-Dmagic, UAtester.

Week 6:
Working with sniffers for monitoring network communication (Wireshark).

Week 7:
Use Snort to perform real time traffic analysis and packet logging.

Week 8:
Perform email analysis using Autopsy tool.

Week 9:
Perform Registry analysis and get boot time logging using process monitor tool.

Week 10:
Perform File type detection using Autopsy tool.

Week 11:

Perform Memory capture and analysis using FTK imager tool.

Week 12:

Perform Network analysis using the Network Miner tool.

Text Books:

1. E. P. Dorothy, Real Digital Forensics for Handheld Devices, Auerback Publications, 2013.
2. E. Casey, Handbook of Digital Forensics and Investigation, Academic Press, 2010.

Reference Books:

1. J. Sammons, The Basics of Digital Forensics: The Primer for Getting Started in Digital Forensics, Syngress Publishing, 2012.
2. C. H. Malin, E. Casey and J. M. Aquilina, Malware Forensics Field Guide for Windows Systems: Digital Forensics Field Guides, Syngress, 2012.
3. J. Wiles and A.Reyes, The Best Damn Cybercrime and Digital Forensics Book Period, Syngress, 2007.

Course Outcomes:

After completion of the course, students will be able to

1. Identify and demonstrate cyber threats, attacks, and vulnerabilities using security tools.
2. Apply cryptographic algorithms, password generation methods, and security mechanisms to secure digital assets.
3. Conduct practical investigations such as port scanning, footprinting, sniffing, and traffic analysis to detect malicious activities.
4. Perform forensic analysis on digital evidence including emails, registries, memory, and files using appropriate forensic tools.
5. Develop problem-solving skills to address real-world cyber security issues and evaluate the effectiveness of security solutions.

I Year M.Tech. CSE II Semester **L T P C**
Course Code: 53219 **0 0 4 2**

PARALLEL COMPUTING LAB
(Professional Elective - III)

Course Objectives:
This course will enable the students to

1. Understand the fundamentals of parallel programming paradigms including MPI, OpenMP, and Pthreads.
2. Develop parallel programs for matrix operations, sorting algorithms, and graph algorithms using MPI, OpenMP, and Pthreads.
3. Analyze and optimize the performance of parallel algorithms.
4. Apply parallel programming techniques to solve real-world computational problems efficiently.

List of Programs:

Week 1:
Design a parallel program to implement Matrix-Vector using MPI library.

Week 2:
Design a parallel program to implement Matrix-Matrix Multiplication using MPI library.

Week 3:
Design a parallel program to implement Bubble Sort using OpenMP and Pthread Programming Constructs.

Week 4& 5:
Design a parallel program to implement Quick Sort using OpenMP and Pthread Programming Constructs.

Week 6 & 7:
Design A Parallel Program to implement Bucket Sort using OpenMP and Pthread Programming Constructs.

Week 8 & 9:
Design a parallel program to implement Prim's Algorithm using OpenMP and Pthread Programming Constructs.

Week 10:
Design a parallel program to implement DFS Algorithm using OpenMP and Pthread Programming Constructs.

Week 11 &12:
Design a parallel program to implement BFS Algorithm using OpenMP and Pthread Programming Constructs.

Week 13 & 14:
Design a parallel program to implement Dijkstra's Algorithm using MPI library.

Text Books:

Ananth Grama, George Karypis, Vipin Kumar and Anshul Gupta, Introduction to Parallel Computing, 2nd edition, Addison-Wesley, 2003.

Reference Books:

1. Michaek J. Quinn, Parallel Computing – Theory and Practice, 2nd edition, Tata McGrawHill Edition.
2. V. Rajaraman and C. Siva Ram Murthy, Parallel Computers – Architectures and Programming, PHI.

Online Resources:

https://nptel.ac.in/courses/106102163

Course Outcomes:

After completion of the course, students will be able to

1. Implement parallel matrix operations using MPI.
2. Develop parallel sorting algorithms with OpenMP and Pthreads.
3. Design parallel graph algorithms (Prim's, DFS, BFS, Dijkstra's) using MPI, OpenMP, and Pthreads.
4. Analyze performance of parallel programs across different parallel paradigms.
5. Implement the data structures that efficiently model the information in a problem.

I Year M.Tech. CSE II Semester **L T P C**
Course Code: 53220 **0 0 4 2**

LARGE LANGUAGE MODELS LAB

(Professional Elective - III)

Course Objectives:

This course will enable the students to

1. Introduce the foundations of transformer architectures into LLMs.
2. Implement various LLM Applications using language Chain and Fine Tune LLAMA 2.
3. Understand and implement Text Classification Using BERT and Tensorflow.
4. Understand and design Multimodal Generative Model,

List of Experiments:

Week 1:

Build your own simple LLM Application using Lang Chain.

Week 2:

Fine tune LLAMA 2 With Custom Dataset Using LoRA and QLoRA Techniques.

Week 3:

Implement text classification using BERT and Tensorflow.

Week 4:

Implement unigram/bigram/trigram models.

Week 5:

Compute perplexity, and compare with a simple neural LM.

Week 6:

Train a small LSTM LM and visualize hidden states/gradients.

Week 7:

Implement self-attention from scratch (matrix form) and verify shapes/outputs.

Week 8:

Implement multi-head attention and show how different heads specialize.

Week 9:

Fine-tune BERT for sentiment classification (SST-2 or custom dataset).

Week 10:

Explore prompt engineering, temperature/top-k sampling, and chain-of-thought style prompting on a small GPT2.

Week 11:

Create prompts that cause models to hallucinate or produce unsafe outputs — discuss mitigations.

Week 12:

Build a simple multimodal generative model that combines text and image inputs to generate captions.

Text Books:

1. Jay Alammar and Maarten Grootendorst, Hands on Large Language Models by O'Reilly, 2023
2. Thushan Ganegedara, Natural Languge Processing with Tensor Flow, by Packt Publishing, 2nd edition 2022.

Reference Books:

1. David Foster, Generative Deep Learning, O'Reilly, 2020.
2. Lewis Tunstall, Leandro von Werra and Thomas Wolf, Natural Language Processing with Trans- formers, 2022.
3. Sebastian Raschka, Build a Large Language Model (From Scratch), ISBN 9781633437166.

Online Resources:

1. Mixture of Experts: Mixture of Experts Explained (huggingface.co).
2. Various benchmarks to evaluate LLMs: LLM Benchmarks: Understanding Language Model Performance (humanloop.com).
3. Types of attention mechanism: Understanding and Coding the Self-Attention Mechanism of Large Language Models from Scratch (sebastianraschka.com).

Course Outcomes:

After completion of the course, students will be able to

1. Build simple LLM Application using Lang Chain. Lo Query PDF using Lang Chain and Pine cone.
2. Apply LORA, QLORA techniques to fine tune LLAMA 2 with Custom Dataset.
3. Implement text classification using BERT and Tensorflow.
4. Analyze and apply advanced multimodal models such as Gemini Vision.
5. Integrate open-source large language models (LLMs) for implementing function calling capabilities within AI-driven applications.

II Year M.Tech. CSE I Semester **L T P C**
Course Code: 533BW **3 0 0 3**

DIGITAL FORENSICS

(Professional Elective – V)
(Common to CSE & CNIS)

Prerequisites: Computer Networks

Course Objectives:
This course will enable the students to

1. Provides an in-depth study of the rapidly changing and fascinating field of computer forensics.
2. Combines both the technical expertise and the knowledge required to investigate, detect and prevent digital crimes.
3. Knowledge on digital forensics legislations, digital crime, forensics processes and procedures, data acquisition and validation, e-discovery tools.
4. E-evidence collection and preservation, investigating operating systems and file systems, network.

UNIT 1: (~ 8 Lecture Hours)
Digital Forensics Science: Forensics science, computer forensics, and digital forensics.
Computer Crime: Criminalistics as it relates to the investigative process, analysis of cyber criminalistics area, holistic approach to cyber-forensics

UNIT 2: (~ 8 Lecture Hours)
Cyber Crime Scene Analysis: Discuss the various court orders etc., methods to search and seizure electronic evidence, retrieved and un-retrieved communications, Discuss the importance of understanding what court documents would be required for a criminal investigation

UNIT 3: (~ 10 Lecture Hours)
Computer Forensics Evidence and capture: Data Recovery Defined-Data Back-up and Recovery-The Role of Back -up in Data Recovery-The Data -Recovery Solution.
Evidence Collection and Data Seizure: Why Collect Evidence? Collection Options Obstacles-Types of Evidence-The Rules of Evidence-Volatile Evidence-General Procedure-Collection and Archiving-Methods of Collections-Art facts-Collection Steps -Controlling Contamination: The chain of custody

UNIT 4: (~ 9 Lecture Hours)
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 Consideration-Practical Implementation

UNIT 5: (~ 8 Lecture Hours)
Mobile Forensics: mobile forensics techniques, mobile forensics tools. Legal Aspects of Digital Forensics: IT Act 2000, amendment of IT Act 2008. Recent trends in mobile forensic technique and methods to search and seizure electronic evidence

Text Books:

1. John Vacca, Computer Forensics: Computer Crime Scene Investigation, Laxmi Publications, 2nd edition.
2. John Sammons, The Basics of Digital Forensics, Elsevier, 2nd edition.

Reference Books:

1. B. Nelson, A. Phillips, C. Steuart, and K. Steuart, Guide to Computer Forensics and Investigations, 6th edition. Boston, MA, USA: Cengage Learning, 2018.
2. Thomas J. Holt, Adam M. Bossler and Kathryn C. and Seigfried-Spellar, Cybercrime and Digital Forensics: An Introduction, Routledge.
3. William Oettinger, Learn Computer Forensics: A beginner's guide to searching, analyzing, and securing digital evidence, Packt Publishing; 1st edition (30 April 2020), ISBN : 1838648178.

Online Resources:

1. https://onlinecourses.nptel.ac.in/noc22_cs66
2. https://www.sans.org/digital-forensics
3. https://www.dfir.training
4. https://www.cybrary.it/course/computer-hacking-and-forensics
5. https://tryhackme.com

Course Outcomes:

After completion of the course, students will be able to

1. Understand the fundamentals of forensic science and computer crime, and explain the role of digital forensics in investigating cyber-criminalistics.
2. Analyze the procedures of cyber-crime scene investigation, including legal requirements, search and seizure of evidence, and interpretation of court orders.
3. Apply techniques for evidence collection, data recovery, and ensure chain of custody while maintaining forensic soundness.
4. Demonstrate the ability to preserve, duplicate, and authenticate digital evidence while complying with legal and forensic standards.
5. Evaluate mobile forensic tools and techniques, and assess the impact of legal frameworks and emerging trends in digital forensics.

II Year M.Tech. CSE I Semester **L T P C**
Course Code: 533BX **3 0 0 3**

ADVANCED OPERATING SYSTEMS

(Professional Elective – V)

Prerequisites: Operating Systems

Course Objectives:
This course will enable the students to
1. Understand the main concepts of advanced operating systems.
2. Identify Hardware and software features that support distributed systems.
3. Learn the working of distributed mutual exclusion and deadlock detection algorithms.
4. Understand the concepts of Multiprocessors, DFS and Distributed Scheduling.

UNIT 1: (~8 Lecture Hours)
Architectures of Distributed Systems: System Architecture Types, Distributed Operating Systems, Issues in Distributed Operating Systems, Communication Primitives. Theoretical Foundations: Inherent Limitations of a Distributed System, Lamport's Logical Clocks, Vector Clocks, Causal Ordering of Messages, Termination Detection

UNIT 2: (~10 Lecture Hours)
Distributed Mutual Exclusion: The Classification of Mutual Exclusion Algorithms,
Non-Token – Based Algorithms: Lamport's Algorithm, The Ricart-Agrawala Algorithm, Maekawa's Algorithm, Token-Based Algorithms: Suzuki-Kasami's Broadcast Algorithm, Singhal's Heurisric Algorithm, Raymond's Heuristic Algorithm

UNIT 3: (~8 Lecture Hours)
Distributed Deadlock Detection: Preliminaries, Deadlock Handling Strategies in Distributed Systems, Issues in Deadlock Detection and Resolution, Control Organizations for Distributed Deadlock Detection, Centralized- Deadlock – Detection Algorithms, Distributed Deadlock Detection Algorithms, Hierarchical Deadlock Detection Algorithms

UNIT 4: (~10 Lecture Hours)
Multiprocessor System Architectures: Introduction, Motivation for multiprocessor Systems, Basic Multiprocessor System Architectures, Multi Processor Operating Systems: Introduction, Structures of Multiprocessor Operating Systems, Operating Design Issues, Threads, Process Synchronization, Processor Scheduling. Distributed File Systems: Architecture, Mechanisms for Building Distributed File Systems, Design Issues

UNIT 5: (~9 Lecture Hours)
Distributed Scheduling: Issues in Load Distributing, Components of a Load Distributed Algorithm, Stability, Load Distributing Algorithms, Requirements for Load Distributing, Task Migration, Issues in task Migration, Distributed Shared Memory: Architecture and Motivation, Algorithms for Implementing DSM, Memory Coherence, Coherence Protocols, Design Issues

Text Books:
1. Mukesh Singhal and Niranjan G. Shivaratri, Advanced Concepts in Operating Systems, Tata McGraw-Hill edition 2001.
2. George Coulouris, J Dollimore and Tim Kindberg, Distributed Systems, Concepts and Design, 4th edition, Pearson Education, 2009.

Reference Books:

1. Andrew S. Tanenbaum and Maarten Van Steen, Distributed systems, Principles and Paradigms, 2nd edition, PHI, 2006.
2. Sukumar Ghosh, Distributed Systems- An Algorithm Approach, Chapman and Hall/CRC, Taylor and Fransis Group, 2007.

Online Resources:

1. https://nptel.ac.in/courses/106106107
2. https://nptel.ac.in/courses/106106168

Course Outcomes:

After completion of the course, students will be able to

1. Understand the design approaches of advanced operating systems.
2. Compare the requirements of Distributed File System and Distributed Shared Memory.
3. Analyze the design issues of distributed operating systems.
4. Evaluate design issues of multi-processor operating systems.
5. Formulate the solutions to schedule the real time applications.

II Year M.Tech. CSE I Semester **L T P C**
Course Code: 533BY **3 0 0 3**

QUANTUM COMPUTING

(Professional Elective – V)

Prerequisites: Linear Algebra, Basic Physics

Course Objectives:
This course will enable the students to

1. Introduce the fundamentals of quantum computing.
2. Explain the problem-solving approach using finite-dimensional mathematics.
3. Explore the physical implementation of quantum bits (qubits) and quantum hardware.
4. Analyze the impact of quantum computing on modern cryptography and security protocols.

UNIT 1: (~ 9 Lecture Hours)
Mathematical Foundations for Quantum Computing:
Some Basic Algebra, Matrix Math, Vectors and Vector Spaces, Set Theory
Complex Numbers:
Definition of Complex Numbers, Algebra of Complex Numbers, Complex Numbers Graphically, Vector Representations of Complex Numbers, Pauli Matrice, Transcendental Numbers

UNIT 2: (~ 10 Lecture Hours)
Quantum physics and Entanglement:
The Journey to Quantum, Quantum Physics Essentials, Basic Atomic Structure, Hilbert Spaces, Uncertainty, Quantum States, Entanglement
Basic Quantum Theory:
Further with Quantum Mechanics, Quantum Decoherence, Quantum Electrodynamics, Quantum Chromodynamics, Feynman Diagram Quantum Entanglement and QKD, Quantum Entanglement, Interpretation, QKE

UNIT 3: (~ 9 Lecture Hours)
Quantum Hardware and Gates:
Further with Qubits, Quantum Gates, More with Gates, Quantum Circuits, The D-Wave Quantum Architecture
Quantum Hardware:
Qubits, How Many Qubits Are Needed? Addressing Decoherence, Topological Quantum Computing, Quantum Essentials

UNIT 4: (~ 9 Lecture Hours)
Quantum Algorithms:
What Is an Algorithm? Deutsch's Algorithm, Deutsch-Jozsa Algorithm, Bernstein-Vazirani Algorithm, Simon's Algorithm, Shor's Algorithm, Grover's Algorithm.

UNIT 5: (~ 9 Lecture Hours)
Current Asymmetric Algorithms: RSA, Diffie-Hellman, Elliptic Curve
The Impact of Quantum Computing on Cryptography: Asymmetric Cryptography, Specific Algorithms, Specific Applications

Text Books:

1. Nielsen M. A., Quantum Computation and Quantum Information, Cambridge University Press.
2. Dr. Chuck Easttom, Quantum Computing Fundamentals, Pearson.

Reference Books:

1. Noson S. Yanofsky and Mirco A. Mannucci, Quantum Computing for Computer Scientists, Cambridge University Pres.
2. G. Benenti, G. Casati, and G. Strini, Principles of Quantum Computation and Information, Vol. I: Basic Concepts; Vol. II: Basic Tools and Special Topics, World Scientific Publishing.
3. A. O. Pittenger, An Introduction to Quantum Computing Algorithms, Publisher, Birkhauser Veriag AG.

Online Resources:

1. https://learning.quantum.ibm.com/catalog.
2. https://www.spinquanta.com/news-detail/learn-about-quantum-computing-a-beginners-guide20250109132129.

Course Outcomes:

After completion of the course, students will be able to

1. Understand the core principles of quantum computing and the foundational mathematics of finite-dimensional spaces.
2. Explain the physical and architectural components of a quantum computer, including qubits and quantum gates.
3. Implement and analyze fundamental quantum algorithms, such as those for searching and integer factorization.
4. Assess the security risks that quantum computing poses to current cryptographic systems.
5. Comprehend the concept of quantum entanglement and its application in technologies like quantum key distribution.

II Year M.Tech. CSE I Semester **L T P C**
Course Code: 533BZ **3 0 0 3**

PROMPT ENGINEERING
(Professional Elective – V)

Prerequisites: Large Language Models

Course Objectives:
This course will enable the students to
1. Understand the architecture, capabilities, and evolution of large language models such as GPT-3.5, GPT-4, Gemini, and LLaMA.
2. Explore standard practices in structured and unstructured text generation using tools like ChatGPT.
3. Apply chunking, tokenization, and formatting techniques for improving text generation and manipulation.
4. Understand the role of embeddings, vector databases (FAISS, Pinecone), and Retrieval-Augmented Generation (RAG) in modern NLP systems.

UNIT 1: (~ 9 Lecture Hours)
Fundamentals and Principles of Prompting: Overview of the Five Principles of Prompting: Give Direction, Specify Format, Provide Examples, Evaluate Quality, Divide Labor. **Introduction to Large Language Models for Text Generation:** What Are Text Generation Models, Vector Representations: The Numerical Essence of Language

UNIT 2: (~ 9 Lecture Hours)
Transformer Architecture: Orchestrating Contextual Relationships, Probabilistic Text Generation: The Decision Mechanism, Historical Underpinnings: The Rise of Transformer Architectures, OpenAI's Generative Pretrained Transformers, GPT-3.5-turbo and ChatGPT, GPT-4, Google's Gemini, Meta's Llama and Open Source

UNIT 3: (~ 9 Lecture Hours)
Standard Practices for Text Generation with ChatGPT: Generating Lists, Hierarchical List Generation, When to Avoid Using Regular Expressions, Generating JSON, YAML Filtering YAML Payloads, Handling Invalid Payloads in YAML, Diverse Format Generation with ChatGPT, Mock CSV Data, Universal Translation Through LLMs, Ask for Context, Text Style Unbundling, Identifying the Desired Textual Features, Generating New Content with the Extracted Features, Extracting Specific Textual Features with LLMs

UNIT 4: (~ 9Lecture Hours)
Chunking Text, Benefits of Chunking Text, Scenarios for Chunking Text, Poor Chunking Example, Chunking Strategies, Sentence Detection Using SpaCy, building a Simple Chunking Algorithm in Python, Sliding Window Chunking, Text Chunking Packages, Text Chunking with Tiktoken, Encodings, Understanding the Tokenization of Strings

UNIT 5: (~ 9 Lecture Hours)
Vector Databases with FAISS and Pinecone: Retrieval Augmented Generation (RAG), Introducing Embeddings, Document Loading Memory Retrieval with FAISS, RAG with Lang Chain, Hosted Vector Databases with Pinecone, Self-Querying, Alternative Retrieval Mechanisms

Text Books:
Phoenix J, Taylor M. Prompt engineering for generative AI. O'Reilly Media, Inc., 2024

Reference Books:

1. Tunstall L, Von Werra L and Wolf T, Natural language processing with transformers, O'Reilly Media, Inc. 2022.
2. Foster D., Generative deep learning. O'Reilly Media, Inc., 2022.

Online Resources:

1. https://www.promptingguide.ai/
2. https://www.deeplearning.ai/short-courses/chatgpt-prompt-engineering-for-developers/

Course Outcomes:

After completion of the course, students will be able to

1. Apply the core principles of prompt engineering for guiding generative AI outputs effectively.
2. Understand the architecture and functionality of state-of-the-art large language models (LLMs).
3. Develop and manipulate structured outputs (JSON, YAML, CSV) using ChatGPT with advanced prompting techniques.
4. Implement text chunking, tokenization, and format control using tools like SpaCy, Tiktoken, and Python.
5. Utilize vector databases such as FAISS and Pinecone in Retrieval-Augmented Generation (RAG) pipelines for efficient information retrieval.

II Year M.Tech. CSE I Semester **L T P C**
Course Code: 533GF **3 0 0 3**

INTELLECTUAL PROPERTY RIGHTS
(Open Elective)

Course Objectives:
This course will enable the students to
1. Understand the Fundamentals and Importance of Intellectual Property Rights (IPR).
2. Learn the types of IPR and their applications in Innovation and Business.
3. Study the Legal Framework, Registration and Protection Mechanisms of IPR.
4. Give insights for Safeguarding IP Assets.
5. Explore New Developments and International Perspectives on IPR.

UNIT 1: (~9 Lecture Hours)
Introduction to Intellectual property: Introduction to Intellectual property: Introduction, Importance of IPR-Types of Intellectual Property Rights (Patents, Copyrights, Trade Marks, Industrial Designs, Geographical Indications, Plant Varieties & Farmer's Rights and Semiconductor Integrated Circuits Layout Design Act)- Overview of Patents-Patent searching process-Overview of Copy rights- Protection of Copyrights

UNIT 2: (~9 Lecture Hours)
Trade Marks: Trade Marks: Purpose and function of Trademarks-Acquisition of Trade Mark Rights- Protectable Matter-Selecting and evaluating Trade mark- Trade Mark Registration Process- Remedies for Infringement

UNIT 3: (~9 Lecture Hours)
Trade Secrets: Trade Secrets: Trade Secret Law-Determination of Trade Secrete Status- Liability for Misappropriations of Trade Secrets-Protection for Submission-Trade Secrete Litigation Unfair competition: Misappropriation right of Publicity-False Advertising

UNIT 4: (~10 Lecture Hours)
Industrial Designs and Intellectual property audits: Importance of Industrial Design– Essential requirement of Registration-Registration Process of Industrial Designs-Benefits of Registration-Assignment, Transmission and Licensing of Industrial Designs- Remedies for Infringement of Designs. Intellectual Property Audits-Types of IP Audit-Procedure of Preparing Audit-Auditing IP Assets

UNIT 5: (~10 Lecture Hours)
New Developments in Intellectual Property and International Scenario: New developments in Intellectual Property: New developments in Patent Law- Copy Right Law-Trade Mark Law-Intellectual Property Audits.International Scenario: International Overview on Intellectual Property-International Organizations, Agencies-Treaties and Conventions. International–Patent laws, Copy Right laws, Trade Mark Laws and International Development in Trade Secrets laws

Text Books:
1. Deborah. E. Bouchoux, Intellectual property right, 4th Edition, Cengage learning, 2015.
2. Prabuddha Ganguli, Intellectual property right – Unleashing the knowledge economy, 4th edition, Tata McGraw Hill Publishing company Ltd., 2017.

Reference Books:

1. S.P Satarkar, Intellectual Property Rights and Copyrights, Ess Publications, 2003.
2. Kompal Bansal, Parikshit Bansal, Fundamentals of Intellectual property for Engineers, BS Publications, 2020.

Online Resources:

1. https://onlinecourses.nptel.ac.in/noc20_hs55 (Course Title: Patent Law for Engineers and Scientists, by Dr. Feroz Ali (IIT Madras))
2. https://onlinecourses.nptel.ac.in/noc20_hs54(Course Title: Patent Drafting for Beginners, by Dr. Feroz Ali (IIT Madras))

Course Outcomes:

After completion of the course, students will be able to

1. Describe fundamental concepts and types of Intellectual Property Rights.
3. Demonstrate the Procedures, Legal frameworks, and Protection Mechanisms related to Patents, Copyrights, Trademarks, Trade Secrets and Industrial Designs.
4. Analyze issues related to infringement, misappropriation, unfair competition and legal remedies associated with different forms of Intellectual Property.
5. Apply the knowledge of IPRs in practical context with reference to Patents Copyrights, Trademarks, Trade Secrets Design Registrations and IP Audits.
6. Examine recent developments in National and International IPR laws.

II Year M.Tech. CSE I Semester **L T P C**
Course Code: 533GA **3 0 0 3**

GENERATIVE AI
(Open Elective)

Prerequisites: Artificial Intelligence

Course Objectives:
This course will enable the students to
1. Introduce the foundations, evolution, and core concepts of AI, ML, DL, NLP, and Generative AI.
2. Understand the advanced neural architectures and generative models such as GANs, VAEs, and Transformers.
3. Explore Large Language Models, prompt engineering, and their real-world applications.
4. Familiarize learners with frameworks, multimodal applications, and ethical considerations in Generative AI.

UNIT 1: (~ 10 Lecture Hours)
Foundations of AI and Generative Models:
Introduction and historical evolution to Artificial Intelligence (AI), Machine Learning (ML), Natural Language Processing (NLP)and Deep Learning (DL), Structure of Artificial Neural Networks (ANNs), Mathematical and computational foundations of generative modeling, Overview of generative models and their applications across various domains; Importance of Generative AI in modern applications, Transfer learning and in advancing Generative AI

UNIT 2: (~ 9 Lecture Hours)
Advanced Neural Architectures for Generative AI:
Variational Autoencoders (VAEs): principles and applications, Generative Adversarial Networks (GANs): architecture and working principles; Transformer architecture and attention mechanisms (in detail); Long Short-Term Memory Networks (LSTMs) and the limitations of traditional RNNs/LSTMs, Advanced Transformer architectures and techniques, Pre-training and transfer learning strategies for generative models

UNIT 3: (~ 9 Lecture Hours)
Large Language Models and Prompt Engineering:
Overview of Large Language Models (LLMs), GPT architecture, variants, and working principles, Pré-training and fine-tuning GPT models for applications (e.g., chatbots, text generation), Case study: GPT-based customer support chatbot, BERT architecture, pre-training objectives, and fine-tuning, Prompt Engineering: Designing effective prompts, controlling model behavior, and improving output quality, Fine-tuning language models for creative writing and chatbot development

UNIT 4: (~ 9 Lecture Hours)
Multi-Agent Systems and Generative AI Applications:
Introduction to Multi-Agent Systems (MAS),Types of agents: reactive, deliberative, hybrid, and learning agents, Multi-agent collaboration and orchestration for generative tasks, Use cases: autonomous research assistants, cooperative creative generation, distributed problem-solving, Frameworks and tools: AutoGen, CrewAI, Hugging GPT for LLM-powered multi-agent systems, Generative AI applications: Art, Creativity, Image/Video generation, Music composition, Healthcare, Finance, Real-world case studies and deployment challenges

UNIT 5: (~ 9Lecture Hours)

Frameworks, Multimodal Applications, and Ethics

LangChain framework: components and LLM application development, Retrieval-Augmented Generation (RAG), Embeddings, Indexing networks, and Vector databases, Generative AI across modalities: Text, Code, Image, and Video generation, Image and Video generation using GANs and VAEs, Multimodal Generative AI: integration and training strategies, Ethical considerations: bias, fairness, trust, and responsible AI deployment, Social and legal implications of Generative AI, Risk mitigation strategies and real-world ethical case studies

Text Books:

1. Altaf Rehmani, Generative AI for Everyone, Understanding the Essentials and Applications of this Breakthrough Technology.
2. Charu C. Aggarwal, Neural Networks and Deep Learning, A Textbook. Joseph Babcock, Raghav Bali, Generative AI with Python and TensorFlow 2, 2024.

Reference Books:

1. Josh Kalin, Generative Adversarial Networks Cookbook, Packt Publishing Limited.
2. Jesse Sprinter, Generative AI in Software Development: Beyond the Limitations of Traditional Coding, 2024.

Online Resources:

1. https://onlinecourses.nptel.ac.in/noc25_cs161/preview
2. https://onlinecourses.nptel.ac.in/noc25_cs159/preview.

Course Outcomes:

After completion of the course, students will be able to

1. Demonstrate knowledge of AI foundations, generative models, and advanced neural architectures.
2. Apply generative AI techniques to create solutions for text, image, video, and multimodal tasks.
3. Design, fine-tune, and optimize Large Language Models for specific applications.
4. Evaluate ethical, social, and legal implications of Generative AI deployments and propose
5. Develop and deploy generative AI-based applications by leveraging state-of-the-art frameworks, libraries, and APIs to solve real-world problems effectively.

II Year M.Tech. CSE I Semester **L T P C**
Course Code: 533GB **3 0 0 3**

INTRUSION DETECTION SYSTEMS
(Open Elective)

Prerequisites: Computer Networks, Computer Programming

Course Objectives:
This course will enable the students to
1. Understand computer and network threats, vulnerabilities, and basic security solutions.
2. Analyze cyber-attacks and study IDS/IPS models, detection techniques, and evaluation methods.
3. Explore advanced threats like malware, botnets, insider attacks, and apply defense mechanisms.
4. Develop skills to design, configure, and evaluate intrusion detection systems and apply them for securing real-world computer and network environments.

UNIT 1: (~9 Lecture Hours)
The state of threats against computers, and networked systems, Overview of computer security solutions, firewalls, why they fail -Exploits, Vulnerabilities, and Buffer Overflow Attacks, VPN's

UNIT 2: (~9 Lecture Hours)
Layers of security, threat, Harmful acts, Internet Governance, Challenges and Constraints, Computer Criminals, CIA Triad, Assets and Threat, motive of attackers, active attacks, passive attacks, Software attacks, hardware attacks, Spectrum of attacks, Taxonomy of various attacks

UNIT 3: (~ 9 Lecture Hours)
Overview of Intrusion Detection and Prevention systems, A General IDS model and taxonomy, Network and Host-based IDS, Signature-based Solutions, Snort, Snort rules, Evaluation of IDS, Cost sensitive IDS

UNIT 4: (~ 9 Lecture Hours)
Anomaly Detection Systems and Algorithms, Network Behavior Based Anomaly Detectors (rate based)-Host-based Anomaly Detector, Software Vulnerabilities, State transition, Immunology, Payload Anomaly Detection

UNIT 5: (~9 Lecture Hours)
Attack trees and Correlation of alerts, Intruders, Proxy Servers and Anonymizers, Phishing, Password Cracking, Keyloggers and Spywares, Virus and Worms, Botnets, Malware detection, Trojan Horse and Backdoors, Steganography, DoS and DDoS attacks, SQL Injection, Buffer Overflow

Text Books:
1. William Stallings, Network Security Essentials: Applications and Standards, 6th edition, Pearson, 2017.
2. N. M. Fadia, Guide to Intrusion Detection and Prevention Systems (IDPS), New Delhi, India: Firewall Media, 2009.

Reference Books:
1. Nina Godbole and Sunit Belpure, Cyber Security Understanding Cyber Crimes, Computer Forensics and Legal Perspectives, Wiley.

2. Peter Szor, The Art of Computer Virus Research and Defense, Symantec Press ISBN 0-321-30545-3.

Online Resources:

1. https://www.cybrary.it
2. https://www.netacad.com/courses/cybersecurity
3. https://www.coursera.org/courses?query=intrusion%20detection
4. https://www.edx.org/learn/cybersecurity
5. https://tryhackme.comum
6. https://cybersecurity.springeropen.com/articles/10.1186/s42400-019-0038-7

Course Outcomes:

After completion of the course, students will be able to

1. Understand computer and network threats and explain basic security solutions including firewalls, VPNs, exploits, and buffer overflow attacks.
2. Identify and classify security threats, attacker motives, and the taxonomy of various cyber-attacks across layers.
3. Illustrate the architecture of IDS/IPS, apply Snort rules, and evaluate intrusion detection models and cost factors.
4. Analyze anomaly detection systems and algorithms for identifying vulnerabilities in host and network environments.
5. Evaluate advanced cyber-attacks such as malware, phishing, and DoS/DDoS, and apply appropriate defensive mechanisms.

II Year M.Tech. CSE I Semester **L T P C**
Course Code: 533GC **3 0 0 3**

DIGITAL FORENSICS
(Open Elective)

Prerequisites: Computer Networks

Course Objectives:
This course will enable the students to
1. Provides an in-depth study of the rapidly changing and fascinating field of computer forensics.
2. Combines both the technical expertise and the knowledge required to investigate, detect and prevent digital crimes.
3. Knowledge on digital forensics legislations, digital crime, forensics processes and procedures, data acquisition and validation, e-discovery tools.
4. E-evidence collection and preservation, investigating operating systems and file systems, network.

UNIT 1: (~ 8 Lecture Hours)
Digital Forensics Science: Forensics science, computer forensics, and digital forensics
Computer Crime: Criminalistics as it relates to the investigative process, analysis of cyber criminalistics area, holistic approach to cyber-forensics

UNIT 2: (~ 8 Lecture Hours)
Cyber Crime Scene Analysis: Discuss the various court orders etc., methods to search and seizure electronic evidence, retrieved and un-retrieved communications, Discuss the importance of understanding what court documents would be required for a criminal investigation

UNIT 3: (~ 10 Lecture Hours)
Computer Forensics Evidence and capture: Data Recovery Defined-Data Back-up and Recovery-The Role of Back -up in Data Recovery-The Data -Recovery Solution
Evidence Collection and Data Seizure: Why Collect Evidence? Collection Options Obstacles-Types of Evidence-The Rules of Evidence-Volatile Evidence-General Procedure-Collection and Archiving-Methods of Collections-Art facts-Collection Steps -Controlling Contamination: The chain of custody

UNIT 4: (~ 9 Lecture Hours)
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 Consideration-Practical Implementation

UNIT 5: (~ 8 Lecture Hours)
Mobile Forensics: mobile forensics techniques, mobile forensics tools. Legal Aspects of Digital Forensics: IT Act 2000, amendment of IT Act 2008. Recent trends in mobile forensic technique and methods to search and seizure electronic evidence

Text Books:
1. John Vacca, Computer Forensics: Computer Crime Scene Investigation, Laxmi Publications, 2nd edition.

2. John Sammons, The Basics of Digital Forensics, Elsevier, 2nd edition.

Reference Books:

1. B. Nelson, A. Phillips, C. Steuart, and K. Steuart, Guide to Computer Forensics and Investigations, 6th edition. Boston, MA, USA: Cengage Learning, 2018.
2. Thomas J. Holt, Adam M. Bossler and Kathryn C. Seigfried-Spellar, Cybercrime and Digital Forensics: An Introduction, Routledge.
3. William Oettinger, Learn Computer Forensics: A beginner's guide to searching, analyzing, and securing digital evidence, Packt Publishing; 1st edition (30 April 2020), ISBN : 1838648178.

Online Resources:

1. https://onlinecourses.nptel.ac.in/noc22_cs66
2. https://www.sans.org/digital-forensics
3. https://www.dfir.training
4. https://www.cybrary.it/course/computer-hacking-and-forensics
5. https://tryhackme.com

Course Outcomes:

After completion of the course, students will be able to

1. Understand the fundamentals of forensic science and computer crime, and explain the role of digital forensics in investigating cyber-criminalistics.
2. Analyze the procedures of cyber-crime scene investigation, including legal requirements, search and seizure of evidence, and interpretation of court orders.
3. Apply techniques for evidence collection, data recovery, and ensure chain of custody while maintaining forensic soundness.
4. Demonstrate the ability to preserve, duplicate, and authenticate digital evidence while complying with legal and forensic standards.
5. Evaluate mobile forensic tools and techniques, and assess the impact of legal frameworks and emerging trends in digital forensics.

I Year M.Tech. CSE I/II Semester **L T P C**
Course Code: 53151 / 53251 **2 0 0 0**

ENGLISH FOR RESEARCH PAPER WRITING

(Audit Course - I / II)

Prerequisite: Basic English Grammar and Composition, Fundamentals of Technical Writing

Course Objectives:

This course will enable the students to

1. Improve the quality and clarity of academic writing specifically for research papers.
2. Provide students with the structure, style, and conventions of scholarly communication.
3. Help students understand how to write titles, abstracts, introductions, literature reviews, methods, results, and conclusions effectively.
4. Guide students in avoiding common grammatical, structural, and ethical mistakes in writing.
5. Build competence in reviewing and editing research manuscripts for publication.

UNIT 1: (~ 6 Lecture Hours)
Effective Writing and Sentence Structuring:
Planning and Preparation, Word Order, Breaking up Long Sentences, Structuring Paragraphs and Sentences, Being Concise and Removing Redundancy, Avoiding Ambiguity and Vagueness

UNIT 2: (~ 6 Lecture Hours)
Academic Writing and Research Ethics:
Clarifying Who Did What, Highlighting your Findings, Hedging and Criticizing, Paraphrasing and Plagiarism, Sections of a Paper, Abstracts and Introduction

UNIT 3: (~ 6 Lecture Hours)
Research Paper Structure and Finalization:
Review of the Literature, Methods, Results, Discussion, Conclusions and Final Check.

UNIT 4: (~ 7 Lecture Hours)
Essential Academic Writing Skills for Research Papers:
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 5: (~ 7 Lecture Hours)
Advanced Research Writing Skills:
Skills are needed when Writing the Methods, Skills needed when writing the Results, Skills are needed when writing the Discussion, and Skills are needed when writing the Conclusions. Useful Phrases, How to ensure Paper is as Good as it could possibly be the First Time Submission.

Text Books:

1. Robert Goldbort, Writing for Science, Yale University Press, 1st edition, 2006.
2. Robert A. Day and Barbara Gastel, How to Write and Publish a Scientific Paper, 6th edition (Cambridge University Press), 2006.

Reference Books:

1. Nicholas J. Higham, Handbook of Writing for the Mathematical Sciences, SIAM, 2nd edition, 1998.
2. Adrian Wallwork, English for Writing Research Papers, Springer, 1st edition, 2011.
3. Barbara Gastel and Robert A. Day, How to Write and Publish a Scientific Paper, Cambridge University Press, 8th edition, 2016.
4. William Strunk Jr. and E.B. White, The Elements of Style, Pearson education, 4th edition, 2000.
5. Angelika H. Hofmann, Scientific Writing and Communication: Papers, Proposals, and Presentations, Oxford University Press, 3rd edition, 2016.
6. Joshua Schimel, Writing Science: How to Write Papers That Get Cited and Proposals That Get Funded, Oxford University Press, 1st edition, 2012

Course Outcomes:

After completion of the course, students will be able to

1. Write grammatically correct, well-structured, and coherent research papers.
2. Use appropriate academic language and tone for different parts of a research article.
3. Construct effective titles, abstracts, and concise conclusions.
4. Apply standard referencing styles and avoid plagiarism.
5. Critically revise and refine research drafts for clarity and publication readiness.

I Year M.Tech. CSE I/II Semester **L T P C**
Course Code: 53158 / 53258 **2 0 0 0**

DISASTER MANAGEMENT

(AC-I & II)
(Common to PEED, DECE, CSE, CNIS, WMC)

Prerequisites: Awareness about Environmental Issues

Course Objectives:
This course will enable the students 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. Critically understand the strengths and weaknesses of disaster management approaches, preparedness, planning, management and programming in different countries, particularly their home country or the countries they work in.
4. Students will get the overview on the roles of government and non- government agencies in disaster management.
5. Describe the basic concepts of the emergency management cycle (mitigation, preparedness, response and recovery) and their application on various types of disasters.

UNIT 1: (~8 Lecture Hours)
Introduction and Repercussions of Disasters and Hazards: Definition, classification, characteristics, causes and impacts of Natural and Manmade Disasters – Industrial disasters, case studies pertaining to specific geographical areas

UNIT 2: (~5 Lecture Hours)
Disaster Risk Assessment: Concept and Elements, Disaster Risk Reduction, People's participation, Risk Assessment, Strategies for Survival, Case Studies of Global, National and Local disasters, Techniques of Risk Reduction for different disasters and Emergency Preparedness

UNIT 3: (~5 Lecture Hours)
Disaster Preparedness and Management: Monitoring of Phenomena Triggering a Disaster or Hazard; Evaluation of Risk: Application of Remote Sensing, Data from Meteorological and other Agencies, Disaster Management Communication: Role of Mass Media and Citizen Centric Approach, Government and Community Preparedness. Disaster Management Cycle

UNIT 4: (~5Lecture Hours)
Disaster Prone Areas of India: Study of Seismic Zones; Areas Prone to Floods, Cloudburst and Droughts, Landslides and Avalanches; Areas Prone to Cyclonic and Coastal Hazards with specific reference to Tsunami; Post-Disaster Diseases and Epidemics

UNIT 5: (~6 Lecture Hours)
Technological Interventions for DRR & Global Framework: Technologies for the detection of multi hazard Early Warning Systems, Global Framework: Role of UNDRR – Overview of Sendai Framework, Environmental Cooperation and Peace Building, Concept and Strategies of Disaster Mitigation, Emerging Trends in Mitigation - Structural Mitigation and Non-Structural Mitigation, Programs of Disaster Mitigation in India. Disaster Management Act, 2005, National Policy on DM Policy, NDMA in Disaster Mitigation in India

Text Books:

1. R.Nishith and Singh A.K, Disaster Management in India: Perspectives, issues and strategies New Royal book Company, 2nd edition, 2021.
2. Sahni, Pradeep Et.Al. (Eds.), Disaster Mitigation Experiences and Reflections, Prentice Hall of India, New Delhi, 3rd edition, 2018.
3. Goel S.L, Disaster Administration and Management Text and Case Studies, Deep & Deep Publication Pvt. Ltd., New Delhi, 4th edition, 2020.

Reference Books:

1. Disaster Management Guidelines. GOI-UNDP Disaster Risk Reduction Programme (2009-2012).
2. Disaster Medical Systems Guidelines. Emergency Medical Services Authority, State of California, EMSA No.214, June 2023.
3. Satapathy S. (2013). 2nd edition. Psychosocial care in Disaster management, A Training of Trainers Manual (ToT), NIDM Publication.
4. Guerisse P.2005. Basic Principles of Disaster Medical Management. Act Anaesth. Belg; 56:395-401
5. Aim and Scope of Disaster Management. Study Guide prepared by Sharman and Hansen. UW-DMC, University of Washington.UNEP.org- ECO - DRR

Online Resources:

1. https://ndma.gov.in/
2. https://www.undrr.org/
3. https://www.undrr.org/implementing-sendai-framework
4. https://onlinecourses.swayam2.ac.in/cec24_hs83/preview
5. https://www.mooc-list.com/tags/earthquake
6. https://freevideolectures.com/course/3581/earthquakes-in-your-backyard
7. https://summer.uci.edu/online/
8. http://www.open.edu/openlearn/free-courses/full-catalogue
9. https://www.edx.org
10. https://www.disasterready.org/courses
11. https://www.unep.org/explore-topics/disasters-conflicts/what-we-do/disaster-risk-reduction/ecosystem-based-disaster-risk

Course Outcomes:

After completion of the course, students will be able to

1. Acquire the knowledge of different disasters and measures to reduce the risk due to disasters.
2. Acquire the knowledge on different methodologies of Risk Assessment.
3. Gain the technical and community based strategies for disaster preparedness, mitigation and Risk Reduction.
4. Understand and evaluate disaster phenomenon and risks specific to India.
5. Apply International frameworks (UNDRR/Sendai) and National mechanisms (NDMA/SDMA) to develop technology driven solutions for DRR.

I Year M.Tech. CSE I/II Semester **L T P C**
Course Code: 53154 / 53254 **2 0 0 0**

SANSKRIT FOR TECHNICAL KNOWLEDGE

(Audit Course – I/II)

Prerequisite: Basic understanding of Indian heritage and classical languages

Course Objectives:

This course will enable the students to

1. Expose students to Sanskrit as a classical language of knowledge and science.
2. Introduce technical terms and concepts embedded in ancient Sanskrit texts.
3. Enable understanding of foundational texts related to mathematics, engineering, and philosophy.
4. Build linguistic skills for reading and interpreting original Sanskrit sources.
5. Appreciate the relevance of Sanskrit in the context of modern scientific discourse.

UNIT 1: (~6 Lecture Hours)
Alphabets in Sanskrit

UNIT 2: (~6 Lecture Hours)
Past / Present / Future Tense, Simple Sentences

UNIT 3: (~6 Lecture Hours)
Order, Introduction of Roots

UNIT 4: (~7 Lecture Hours)
Technical Information about Sanskrit Literature

UNIT 5: (~7 Lecture Hours)
Technical Concepts of Engineering-Electrical, Mechanical, Architecture, Mathematics

Text Books:

1. Dr. H. R. Vishwasa, Abhyaspustakam, Samskrita Bharati Publication, New Delhi, 1st edition, 2012.
2. Vempati Kutumbashastri, Teach Yourself Sanskrit: Prathama Diksha, Rashtriya Sanskrit Sansthanam, New Delhi, 1st edition, 2002.

Reference Books:

1. S. Balachandra Rao,, Technical Literature in Sanskrit, Rashtriya Sanskrit Vidyapeetha, 1st edition, 2005.
2. Prabhakar Apte, Sanskrit and Science, Central Institute of Indian Languages, 1st edition, 2003.
3. R. Ganapathi, Scientific Heritage of India in Sanskrit, Bharatiya Vidya Bhavan, 1st edition, 1990.
4. Rick Briggs, Sanskrit and Artificial Intelligence, AI Magazine (Journal Paper), 1st edition, 1985.
5. M. Sampath Kumar, Essentials of Sanskrit Language for Engineering Students, Sri Chandrasekharendra Saraswathi Viswa Mahavidyalaya, 1st edition, 2010.
6. R.S. Vadhyar and R.S. Vadhyar, Vyavaharika Samskritam (Functional Sanskrit), Sons, 3rd edition, 2008.

Course Outcomes:

After completion of the course, students will be able to

1. Understand the structure and grammar of Sanskrit relevant to technical usage.
2. Recognize and interpret key technical terms and concepts from ancient Sanskrit literature.
3. Translate and explain Sanskrit verses that relate to scientific and engineering disciplines.
4. Develop an interdisciplinary perspective connecting ancient wisdom with contemporary science.
5. Appreciate the contribution of Sanskrit to Indian scientific, philosophical, and cultural heritage.

I Year M.Tech. CSE I/II Semester **L T P C**
Course Code: 53157 / 53257 **2 0 0 0**

VALUE EDUCATION
(Audit Course – I/II)

Prerequisite: Basic Understanding of Ethics and Social Responsibility

Course Objectives:
This course will enable the students to
1. Understand the importance of values in personal and professional life.
2. Promote ethical behavior and decision-making based on human values.
3. Develop a sense of responsibility, empathy, and integrity.
4. Cultivate respect for diversity, equality, and sustainable living.
5. Encourage self-reflection and a commitment to lifelong value-based learning.

UNIT 1: (~6 Lecture Hours)
Values and Ethics: 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 2: (~6 Lecture Hours)
Core Personal Values: 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 3: (~6 Lecture Hours)
Personality Traits: Personality and Behavior Development, Soul and Scientific Attitude, Positive Thinking, Integrity and Discipline, Punctuality, Love and Kindness

UNIT 4: (~7 Lecture Hours)
Virtuous Living: Avoid Fault Thinking, Free from Anger, Dignity of Labour, Universal Brotherhood and Religious Tolerance, True Friendship, Happiness vs Suffering, Love for Truth, Aware of Self- Destructive Habits, Association and Cooperation, Doing Best for Saving Nature

UNIT 5: (~7 Lecture Hours)
Character and Wisdom: 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

Text Books:
1. S. K. Chakraborty, Values and Ethics for Organizations: Theory and Practice, Oxford University Press, 1st edition, 1998 paperback reprint 1999.
2. R.R. Gaur, R. Sangal and G.P. Bagaria, Value Education and Professional Ethics, Excel Books, 1st edition, 2010.

Reference Books:

1. Education in Values: A Source Book, UNESCO, NCERT Publication, 1st edition, 2002.
2. S. Ignacimuthu, Value Education: Principles and Practice, Don Bosco Publications, 1st Edition, 2009.
3. G. Rajagopalan, Value Education: Theory and Practice, Bharatiya Vidya Bhavan, 1st Edition, 2011.
4. Education for Values in Schools – A Framework, NCERT, NCERT Publication, 1st Edition, 2012.
5. A.C. Bhaktivedanta Swami Prabhupada, Education in Human Values, Bhaktivedanta Book Trust, 1st edition, 2001.
6. M. M. Goel, Teaching of Values: Some Reflections, Shipra Publications, 1st edition, 2005.

Course Outcomes:

After completion of the course, students will be able to

1. Recognize and apply core human values such as honesty, compassion, and respect.
2. Analyze ethical dilemmas and make morally sound decisions.
3. Demonstrate socially responsible behavior in both personal and professional contexts.
4. Promote harmony in relationships, society, and the environment.
5. Engage in continuous personal development guided by ethical principles.

I Year M.Tech. CSE I/II Semester **L T P C**
Course Code: 53156 / 53256 **2 0 0 0**

CONSTITUTION OF INDIA
(Audit Course – I/II)

Prerequisites: Basic knowledge of Indian history and Governance

Course Objectives:
This course will enable the students to

1. Provide a comprehensive understanding of the Indian Constitution, its structure, and significance.
2. Familiarize students with the fundamental rights, duties, and directive principles.
3. Introduce the key organs of government and their roles in a democratic system.
4. Promote awareness of constitutional values, governance mechanisms, and public responsibility.
5. To understand the relationship between the Constitution and the legal-administrative framework of India.

UNIT 1: (~6 Lecture Hours)
Constitution Drafting History, Constitutional Philosophy: History of Making of the Indian Constitution, History Drafting Committee, (Composition and Working). Philosophy of the Indian Constitution, Preamble, Salient Features

UNIT 2: (~6 Lecture Hours)
Rights and Duties Framework: Contours of Constitutional Rights and 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 3: (~7 Lecture Hours)
Organs of Governance: Parliament, Composition, Qualifications and Disqualifications, Powers and Functions, Executive, President, Governor, Council of Ministers, Judiciary, Appointment and Transfer of Judges, Qualification, Powers and Functions

UNIT 4: (~7 Lecture Hours)
Local Administration: District's Administration Head, Role and Importance. Municipalities: Introduction, Mayor and Role of Elected Representative, CEO of Municipal Corporation. Panchayat Raj: Introduction, PRI: Zila Panchayat, Elected Officials and their Roles, CEO Zila Panchayat, Position and Role, Block Level, Organizational Hierarchy (Different Departments), Village Level, Role of Elected and Appointed Officials, Importance of Grass Root Democracy

UNIT 5: (~6 Lecture Hours)
Election Commission: Role and Functioning, Chief Election Commissioner and Election Commissioners. State Election Commission: Role and Functioning. Institute and Bodies for the and Women

Text Books:

1. The Constitution of India (Bare Act), Government Publication, 1st edition, 1950.
2. Dr. S. N. Busi, Dr. B. R. Ambedkar: Framing of Indian Constitution, Ava Publications, 1st edition, 2016.

Reference Books:

1. M. P. Jain, Introduction to the Constitution of India, LexisNexis, 7th edition, 2014.
2. D. D. Basu, Introduction to the Constitution of India, LexisNexis, 22nd edition, 2015.
3. M. Laxmikanth, Indian Polity, McGraw Hill Education, 6th edition, 2021.
4. Subhash Kashyap, Our Constitution, National Book Trust, 1st edition, 2011.
5. Arun K. Thiruvengadam,The Constitution of India: A Contextual Analysis, Bloomsbury Publishing, 1st edition, 2017.
6. P.M. Bakshi, The Constitution of India, Universal Law Publishing, 17th edition, 2020.

Course Outcomes:

After completion of the course, students will be able to

1. Describe the history, evolution, and philosophy behind the Constitution of India.
2. Explain the fundamental rights and duties of citizens and the structure of the Indian government.
3. Analyze the functioning of constitutional bodies and judicial systems.
4. Understand the significance of constitutional amendments and landmark legal cases.
5. Demonstrate responsible citizenship and awareness of constitutional governance

I Year M.Tech. CSE I/II Semester **L T P C**
Course Code: 53153 / 53253 **2 0 0 0**

PEDAGOGY STUDIES
(Audit Course – I/II)

Prerequisite: Basic understanding of Teaching-Learning processes

Course Objectives:
This course will enable the students to
1. Understand the concepts, principles, and theories of pedagogy and their application.
2. Evaluate the effectiveness of different teaching approaches in varied educational contexts.
3. Analyze the impact of teacher behavior, classroom environment, and instructional strategies on learning.
4. Assess the challenges in implementing pedagogical innovations in diverse settings.
5. Enable the design of learner-centered, inclusive, and effective educational practices.

UNIT 1: (~6 Lecture Hours)
Pedagogical Foundations: Introduction and Methodology, Aims and rationale, Policy background, Conceptual framework and terminology Theories of learning, Curriculum, Teacher education. Conceptual framework, Research questions. Overview of methodology and Searching

UNIT 2: (~6 Lecture Hours)
Global Pedagogical Practices: Thematic Overview, Pedagogical Practices are being used by Teachers in Formal and Informal Classrooms in Developing Countries, Curriculum, Teacher Education

UNIT 3: (~7 Lecture Hours)
Effective Pedagogy Evidence: Evidence on the Effectiveness of Pedagogical Practices, Methodology for the Indepth Stage, Quality Assessment of Included Studies, How Can Teacher Education (Curriculum And Practicum) and the School Curriculum and Guidance Materials best Support Effective Pedagogy? Theory of Change, Strength and Nature of the body of Evidence for Effective Pedagogical Practices, Pedagogic Theory and Pedagogical Approaches, Teachers' Attitudes and Beliefs and Pedagogic Strategies

UNIT 4: (~7 Lecture Hours)
Professional development: Alignment with Classroom Practices and Follow-Up Support, Peer Support, Support from the Head Teacher and the Community, Curriculum and Assessment, Barriers to Learning, Limited Resources and LargeClass Sizes

UNIT 5: (~6 Lecture Hours)
Future Pedagogical Research: Research Gaps and Future Directions, Research Design, Contexts, Pedagogy, Teacher Education, Curriculum and Assessment, Dissemination and Research Impact

Text Books:
1. Ackers, J., and Hardman, F., Classroom interaction in Kenyan primary schools Compare a Journal of Comparative and International Education, Volume 31, 2001.
2. Agrawal. M, Curricular reform in schools: The importance of evaluation, Journal of Curriculum Studies, Volume 36, 2003.

Reference Books:

1. Akyeampong. K, Teacher training in Ghana - does it count?, Multi-site teacher education research project (MUSTER) country report 1. Department for International Development (DFID), London, 1st Edition, 2003.
2. Susan A. Ambrose, Jossey- Bass, How Learning Works: Seven Research-Based Principles for Smart Teaching, 1st edition, 2010.
3. John Hattie, Routledge, Visible Learning: A Synthesis of Over 800 Meta-Analyses Relating to Achievement 1st edition, 2009.
4. John Biggs, Catherine Tang, Teaching for Quality Learning at University, McGraw-Hill Education, 4th edition, 2011.
5. Stephen D. Brookfield, Jossey-Bass, The Skillful Teacher: On Technique, Trust, and Responsiveness in the Classroom, 3rd edition, 2015.
6. Dale H. Schunk, Learning Theories: An Educational Perspective, Pearson Education, 7th edition, 2015.

Course Outcomes:

After completion of the course, students will be able to

1. Explain key pedagogical theories and their relevance to classroom teaching.
2. Compare traditional and modern teaching strategies based on evidence from research.
3. Identify factors affecting student engagement, motivation, and learning outcomes.
4. Design instructional plans that incorporate effective pedagogical principles.
5. Critically evaluate and adapt teaching practices to meet diverse learner needs.

I Year M.Tech. CSE I/II Semester **L T P C**
Course Code: 53155 / 53255 **2 0 0 0**

STRESS MANAGEMENT BY YOGA
(Audit Course – I/II)

Prerequisite: Basic awareness of mental and physical health

Course Objectives:
This course will enable the students to
1. Introduce the concept of stress and its impact on physical and mental well-being.
2. Provide an understanding of yoga as a tool for stress relief and emotional balance.
3. Teach various yogic practices including asana, pranayama, and meditation for managing stress.
4. Cultivate self-awareness, relaxation, and resilience through regular yogic practice.
5. Promote a healthy lifestyle by integrating yogic discipline in daily life.

UNIT 1: (~6 Lecture Hours)
Definitions of Eight parts of yoga. (Ashtanga)

UNIT 2: (~6 Lecture Hours)
Yam and Niyam

UNIT 3: (~7 Lecture Hours)
Do`s and Dont`s in Life: Ahinsa, Satya, Astheya, Bramhacharya and Aparigraha. Shaucha, Santosh, Tapa, Swadhyay, Ishwarpranidhan

UNIT 4: (~6 Lecture Hours)
Aasan and Pranayam

UNIT 5: (~7 Lecture Hours)
Various Yoga Poses and their Benefits for Mind and Body. Regularization of Breathing Techniques and its Effects, Types of Pranayam

Text Books:
1. Janardan Swami Yogabhyasi Mandal, Janardan Swami Yogabhyasi Mandal, Yogic Asanas for Group Training – Part I, Nagpur, Standard Edition, 1990.
2. Swami Vivekananda, Rajayoga or conquering the Internal Nature, Advaita Ashrama (Publication Department), Kolkata, 2010.

Reference Books:
1. T.K.V. Desikachar, The Heart of Yoga: Developing a Personal Practice, Inner Traditions, 1st edition, 1999.
2. Swami Shivapremananda, Yoga for Stress Relief, Jaico Publishing House, 1st edition, 2002.
3. B.K.S. Iyengar, Light on Yoga, HarperCollins, Revised Edition, 2015.
4. Herbert Benson and HarperTorch, The Relaxation Response, Updated Edition, 2000.
5. Shri Yogendra, Yoga for Wellness, The Yoga Institute, 1st edition, 2001.
6. Timothy McCall, Yoga as Medicine, Bantam Books, 1st edition, 2007.

Course Outcomes:

After completion of the course, students will be able to

1. Understand the causes and physiological effects of stress.
2. Apply basic yogic techniques to reduce stress and enhance focus.
3. Practice breathing techniques and meditation to maintain emotional stability.
4. Demonstrate improved physical flexibility, mental clarity, and stress tolerance.
5. Incorporate yoga as a sustainable approach to managing academic, professional, and personal pressures.

I Year M.Tech. CSE I/II Semester **L T P C**
Course Code: 53152 / 53252 **2 0 0 0**

PERSONALITY DEVELOPMENT THROUGH LIFE ENLIGHTENMENT SKILLS

(Audit Course – I/II)

Prerequisite: Basic Communication and Interpersonal skills

Course Objectives:

This course will enable the students to

1. Understand the concept of personality and its development through ethical and moral grounding.
2. Enhance self-awareness, confidence, and emotional intelligence.
3. Inculcate life-enlightening values drawn from Indian wisdom and philosophy.
4. Improve communication, leadership, and decision-making abilities.
5. Promote a positive attitude and holistic approach toward life and career.

UNIT 1: (~6 Lecture Hours)

Neetisatakam-Holistic development of personality

1. Verses- 19,20,21,22 (wisdom)
2. Verses- 29,31,32 (pride & heroism)
3. Verses- 26,28,63,65 (virtue)

UNIT 2: (~6 Lecture Hours)

Neetisatakam-Holistic development of personality

1. Verses- 52,53,59 (dont’s)
2. Verses- 71,73,75,78 (do’s)

UNIT 3: (~7 Lecture Hours)

Approach to day-to-day work and duties.

1. Shrimad Bhagwad Geeta: Chapter 2-Verses 41, 47,48
2. Chapter 3-Verses 13, 21, 27, 35, Chapter 6-Verses 5,13,17, 23, 35
3. Chapter 18-Verses 45, 46, 48

UNIT 4: (~7 Lecture Hours)

Statements of basic knowledge.

1. Shrimad Bhagwad Geeta: Chapter2-Verses 56, 62, 68
2. Chapter 12 -Verses 13, 14, 15, 16,17, 18
3. Personality of Role model. Shrimad Bhagwad Geeta:

UNIT 5: (~6 Lecture Hours)

1. Chapter2-Verses 17, Chapter 3-Verses 36,37,42
2. Chapter 4-Verses 18, 38,39
3. Chapter18 – Verses 37,38,63

Text Books:

1. Swami Swarupananda, Srimad Bhagavad Gita, Advaita Ashram (Publication Department), Kolkata, 2018.
2. P. Gopinath, Bhartrihari’s Three Satakas, (Niti-sringar-vairagya), Rashtriya Sanskrit Sansthanam, New Delhi, 1st edition, 2002.

Reference Books:

1. Anthony Robbins, Awakening the Giant Within, Free Press, 1st edition, 1992.
2. Stephen R. Covey, Simon and Schuster, The 7 Habits of Highly Effective People, 30th Anniversary edition, 2020.
3. A.P.J. Abdul Kalam, Wings of Fire: An Autobiography, Universities Press, 30th Impression, 2014.
4. Swami Rama, Living with the Himalayan Masters, Himalayan Institute Press, Revised edition, 2002.
5. Robin Sharma, The Monk Who Sold His Ferrari, Jaico Publishing House, 1st edition, 1997.
6. Eckhart Tolle, The Power of Now, New World Library, 1st edition, 1999.

Course Outcomes:

After completion of the course, students will be able to

1. Explain the key elements of personality and factors influencing its growth.
2. Demonstrate improved self-confidence, empathy, and interpersonal relationships.
3. Apply principles from enlightened texts (e.g., Bhagavad Gita, Upanishads) to everyday decision-making.
4. Exhibit qualities of ethical leadership and responsible citizenship.
5. Lead a balanced, purposeful, and value-driven personal and professional life.