# ACADEMIC REGULATIONS (R20)

## B.Tech FOUR YEAR DEGREE Programme

(*Applicable for the batches admitted from the* ***A.Y. 2020-21***)

**UNIVERSITY COLLEGE OF ENGINEERING KAKINADA(A)**

**JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY KAKINADA**

**KAKINADA – 533003, ANDHRA PRADESH, INDIA**

*College: https://www.jntucek.ac.in/ University : https://www.jntuk.edu.in/*

# ACADEMIC REGULATIONS (R20) FOR B. TECH. (REGULAR)

**Applicable for the students of B. Tech. (Regular) from the Academic Year 2020-21 onwards**

## 1. Award of B. Tech. Degree

A student will be declared eligible for the award of B. Tech. Degree if he fulfils the following academic regulations:

1. A student shall be declared eligible for the award of the B. Tech Degree, if he pursues a course of study in not less than four and not more than eight academic years
2. The candidate shall register for 160 credits and secure all the 160 credits.
3. A student will be eligible to get Under Graduate degree with ***Honors or additional Minor Engineering***, if he/she completes an additional 20 credits. These could be acquired through the courses recommended by the respective Board of Studies. To award Honors / Minor Engineering degree, student should not have any **backlog history** with other requirements.

## 2. Courses of study

The following courses of study are offered at present as specializations for the B. Tech. Courses:

| S. No | Branch |
|---|---|
| 01 | Civil Engineering |
| 02 | Electrical and Electronics Engineering |
| 03 | Mechanical Engineering |
| 04 | Electronics and Communication Engineering |
| 05 | Computer Science and Engineering |
| 06 | Petroleum Engineering |
| 07 | Chemical Engineering |

## 3. Distribution and Weightage of Marks

(i) The performance of a student in each semester shall be evaluated subject – wise with a maximum of **100 marks for theory and practical subject/courses**. The project work shall be evaluated for 200 marks.

(ii) For theory subjects the distribution shall be 30 marks for Internal Evaluation and 70 marks for the End - Examinations.

(iii) For theory subjects, during the semester there shall be 2 tests/assessments. The weightage of Internal marks for 30 consists of Descriptive – 15, Assignment - 05 (Theory, Design, Analysis, Simulation, Algorithms, Drawing, etc. as the case may be and for Physics Virtual Labs to be consider as Assignments) Objective -10. The objective examination is for 20 minutes duration. The subjective examination is for 90 minutes duration conducted for 15 marks. Each subjective type test question paper shall contain **3 questions** and all questions need to be answered. The Objective examination conducted for 10 marks and subjective examination conducted for 15 marks are to be added to the assignment marks of 5 for finalizing internal marks for 30. **Internal Marks** can be calculated with 80% weightage for best of the two Mids and 20% weightage for other Mid Exam. ***As the syllabus is framed for***

***5 units, the 1st mid examination (both Objective and Subjective) is conducted in 1-2½ units and second test in 2½-5 units of each subject in a semester***.

(iv) The semester end examination/assessment is conducted covering the topics of all Units for 70 marks. **End Exam Paper containing FIVE mandatory questions** (one question from one unit) with internal choice, each carrying 14 marks gives for 70 marks.

(v) For **practical courses there shall be continuous evaluation during the semester for 30 internal marks and 70 end examination marks**. The internal 30 marks shall be awarded as follows: day to day work and record-10 marks and the remaining 20 marks to be awarded by conducting an internal laboratory test. The end examination shall be conducted by the teacher concerned and external examiner as follows:

| | *Procedure* | *Experimentation* | *Result* | *Viva-voce* | *Total* |
|---|---|---|---|---|---|
| Marks | 15 | 30 | 10 | 15 | 70 |

(vi) For the courses / subjects having design and / or drawing (such as Engineering Graphics, Engineering Drawing, Machine Drawing), *Computer Workshop* and estimation, the distribution shall be 30 marks for internal evaluation (20 marks for day-to-day work, and 10 marks for internal tests) and 70 marks for end examination. *There shall be two internal tests in a Semester and the Marks for 10 can be calculated with 80% weightage for best of the two tests and 20% weightage for other test and these are to be added to the marks obtained in day to day work*. SEE Question paper pattern also reflects the course handled procedure and different with regular course

(vii) For the seminar, each student has to be evaluated based on the presentation of any latest topic with report of 10-15 pages and a presentation (viz., ppt or any of min 10 slides). The student shall collect the information on a specialized topic and prepare a technical report, showing his understanding over the topic, and submit to the department, which shall be evaluated by the Departmental committee consisting of Head of the department, seminar supervisor and a senior faculty member. *The seminar report shall be evaluated for 50 marks. There shall be no external examination for seminar*.

(viii) There shall be 05 skill-oriented courses (*maximum of 2 credits each*) offered during III to VII semesters. Among the five skill courses, four courses shall focus on the basic and advanced skills related to the domain courses and the remaining one shall be a soft skills course. The student shall be given an option to choose either the skill courses being offered by the college or to choose a certificate course being offered by industries/Professional bodies/APSSDC or any other accredited bodies as approved by the concerned BoS

(ix) Students shall undergo *mandatory summer internships* for a minimum of six weeks duration at the end of second and third year of the Programme. *Evaluation of the summer internships shall be through the departmental committee*. A student will be required to submit a summer internship report to the concerned department and appear for an oral presentation before the departmental committee. The report and the oral presentation shall carry 40% and 60% weightages respectively

(x) In the final semester, the student should mandatorily undergo internship (*full internship in the final semester*) and in parallel he/she should work on a project with well-defined objectives. At the end of the semester the candidate shall submit an internship completion certificate and a project report. *A student shall also be permitted to submit project report on the work carried out during the internship*. The project report shall be evaluated with an external examiner.

(xi) Out of a total of 200 marks for the project work, 60 marks shall be for Internal Evaluation and 140 marks for the End Semester Examination. The End Semester Examination (Viva-Voce) shall be conducted by the committee. *The committee consists of an external examiner, Head of the Department and Supervisor of the Project*. The evaluation of project work shall be conducted at the end of the IV year. *The Internal Evaluation shall be on the basis of two seminars given by each student on the topic of his project and evaluated by an internal committee*.

**4. Attendance Requirements**

1. A student is eligible to write the External examinations if he acquires a minimum of 75% of attendance in aggregate of all the subjects.
2. Condonation of shortage of attendance in aggregate up to 10% (65% and above and below 75%) in each semester may be granted by the College Academic Committee, ***and one is eligible for condonation a maximum of THREE times during the entire course work.***
3. Shortage of Attendance below 65% in aggregate shall not be condoned.
4. A student who is short of attendance in semester may seek re-admission into that semester when offered within 4 weeks from the date of the commencement of class work.
5. Students whose shortage of attendance is not condoned in any semester are not eligible to write their end semester examination of that class.
6. A stipulated fee shall be payable towards condonation of shortage of attendance. (a) A student is eligible to write the University examinations if he acquires a minimum of 50% **in each subject/course including laboratories** and 75% of attendance in aggregate of all the subjects.
7. A student will be promoted to the next semester if he satisfies the (i) attendance requirement of the present semester and (ii) minimum required credits.
8. If any candidate fulfills the attendance requirement in the present semester, he shall not be eligible for readmission into the same class.

**5. Minimum Academic Requirements**

The following academic requirements have to be satisfied in addition to the attendance requirements mentioned in item no.4.

5.1 A student is deemed to have satisfied the minimum academic requirements if he has **earned the credits allotted to each theory/practical design/drawing subject/project and secures not less than 35% of marks in the end semester exam, and minimum 40% of marks in the sum total of the internal marks and end semester examination marks.**

5.2 A student shall be promoted from first year to second year if he fulfills the minimum attendance requirement.

5.3 A student will be **promoted from II year to III year** if he fulfills the academic requirement **of 50% of the credits from all the examinations up to II year I semester (i.e., including).**

5.4 A student shall be **promoted from III year to IV year** if he fulfils the academic requirements of **50% of the credits from all the examinations up to III year I semester (i.e., including).**

5.5 **A student shall register and put up minimum attendance in all 160 credits and earn all 160 credits.**

### 6. Course Pattern

1. The entire course of study is for four academic years, all the years are on semester pattern.
2. A student eligible to appear for the end semester examination in a subject, but absent from it or has failed in the end semester examination, may write the exam in that subject when conducted next.
3. When a student is detained for lack of credits / shortage of attendance, he may be re- admitted into the same semester / year in which he has been detained. However, the academic regulations under which he was first admitted shall continue to be applicable to him.
4. Gap Year – concept of Student Entrepreneur in Residence shall be introduced and outstanding students who wish to pursue entrepreneurship are allowed to take a break of one year at any time after I year/II year/III year to pursue entrepreneurship full time. This period shall be counted for the maximum time for graduation. An evaluation committee at university level shall be constituted to evaluate the proposal submitted by the student and the committee shall decide on permitting the student for availing the Gap Year.

### *7. Cumulative Grade Point Average (CGPA)*

After each subject is evaluated for 100 marks, the marks obtained in each subject will be converted to a corresponding letter grade as given below, depending on the range in which the marks obtained by the student fall

| Marks Range | Level | Letter Grade | Grade Point |
|---|---|---|---|
| ≥ 90 | Outstanding | A+ | 10 |
| 80-89 | Excellent | A | 9 |
| 70-79 | Very Good | B | 8 |
| 60-69 | Good | C | 7 |
| 50-59 | Fair | D | 6 |
| 40-49 | Satisfactory | E | 5 |
| < 40 | Fail | F | 0 |
| - | Absent | Ab | 0 |

***Computation of SGPA***

The following procedure is to be adopted to compute the Semester Grade Point Average (SGPA) and Cumulative Grade Point Average (CGPA):

The **SGPA** is the ratio of sum of the product of the number of credits with the grade points scored by a student in all the courses taken by a student and the sum of the number of credits of all the courses undergone by a student, i.e

***SGPA (Si) = ∑ (Ci X Gi) / ∑ Ci***

Where Ci is the number of credits of the ith course and Gi is the grade point scored by the student in the $i^{th}$ course.

***Computation of CGPA***

The **CGPA** is also calculated in the same manner taking into account all the courses undergone by a student over all the semester of a programme, i.e.

***CGPA = ∑ (Ci X Si) / ∑ Ci***

Where Si is the SGPA of the $i^{th}$ semester and Ci is the total number of credits in that semester. The SGPA and CGPA shall be rounded off to TWO decimal points and reported in the transcripts. While computing the SGPA/CGPA, the subjects in which the student is awarded Zero grade points will also be included

Conversion of CGPA into equivalent percentage as follows:

*Equivalent Percentage = (CGPA –0.75) x 10 (as per AICTE)*

## 8. Award of Class

After a student has satisfied the requirements prescribed for the completion of the program and is eligible for the award of B. Tech. Degree, he/she shall be placed in one of the following four classes:

| Class Awarded | CGPA to be Secured | |
|---|---|---|
| First Class with Distinction | $\geq$ 7.5 without backlog history | From the CGPA secured from 160 Credits. |
| First Class | $\geq$ 6.5 < 7.5 without backlog history<br>$\geq$ 6.5 with backlog history | |
| Second Class | $\geq$ 5.5 < 6.5 | |
| Pass Class | $\geq$ 4.0 < 5.5 | |

**9.** ***Honors Degree***: A student should complete an **additional 20 credits** by doing Board of Studies recommended courses and meet criteria as follows:

i. A student shall be permitted to register for Honors program at the beginning of 3rd / 4th semester provided that the student must have acquired a minimum of 8.0 SGPA upto the end of 2nd semester without any **backlog history**. In case of the declaration of the 3rd semester results after the commencement of the 4th semester and if a student fails to score the required minimum of 8.0 SGPA, his/her registration for Honors Programme stands cancelled and he/she shall continue with the regular Programme.

ii. Students can select the additional and advanced courses from their respective branch in which they are pursuing the degree and get an honors degree in the same. e.g. If a Mechanical Engineering student completes the selected advanced courses from same branch under this scheme, he/she will be awarded B.Tech. (Honors) in Mechanical Engineering.

iii. In addition to fulfilling all the requisites of a Regular B.Tech Programme, a student shall earn 20 additional credits to be eligible for the award of B. Tech (Honors) degree. This is in addition to the credits essential for obtaining the Under Graduate Degree in Major Discipline (i.e. 160 credits).

iv. Of the 20 additional Credits to be acquired, 16/15 credits shall be earned by undergoing specified courses listed as pools, with four/five courses, each carrying 4/3 credits. The remaining 4/5 credits must be acquired through two MOOCs, which shall be domain specific, with 2/3 credits and with a minimum duration of 8/12weeks as recommended by the Board of studies.

v. It is the responsibility of the student to acquire/complete prerequisite before taking the respective course. The courses offered in each pool shall be domain specific courses and advanced courses

vi. The concerned BoS shall decide on the minimum enrolments for offering Honors program by the department. If minimum enrolments criteria are not met then the students **shall be permitted to register for the equivalent MOOC courses as approved by the concerned Head of the department in consultation with BoS**.

vii. If a student drops or is terminated from the Honors program, the additional credits so far earned cannot be converted into free or core electives; they will remain extra. These additional courses will find mention in the transcript (but not in the degree certificate). In such cases, the student may choose between the actual grade or a “pass (P)” grade and also choose to omit the mention of the course as for the following: All the courses done under the dropped Minors will be shown in the transcript. None of the courses done under the dropped Minor will be shown in the transcript.

viii. In case a student fails to meet the CGPA requirement for Degree with Honors at any point after registration, he/she will be dropped from the list of students eligible for Degree with Honors and they will receive regular B.Tech degree only. However, such students will receive a separate grade sheet mentioning the additional courses completed by them.

ix. Honors must be completed simultaneously with a major degree program. A student cannot earn Honors after he/she has already earned bachelor's degree.

**10.** ***Minor Engineering***: A student should complete an **additional 20 credits** by doing respective Board of Studies recommended courses and as follows:

i. Students who are desirous of pursuing their special interest areas other than the chosen discipline of Engineering may opt for additional courses in minor specialization groups offered by a department other than their parent department. For example, If Mechanical Engineering student selects subjects from Civil Engineering under this scheme, he/she will get Major degree of Mechanical Engineering with minor degree of Civil Engineering

ii. Student can also opt for Industry relevant tracks of any branch to obtain the Minor Degree, for example, a B.Tech Mechanical student can opt for the industry relevant tracks like Data Mining track, IOT track, Machine learning track etc.

iii. The BOS concerned shall identify as many tracks as possible in the areas of emerging technologies and industrial relevance / demand. For example, the minor tracks can be the fundamental courses in CSE, ECE, EEE,CE,ME etc or industry tracks such as Artificial Intelligence (AI), Machine Learning (ML), Data Science (DS), Robotics, Electric vehicles, Robotics, VLSI etc.

iv. The list of disciplines/branches eligible to opt for a particular industry relevant minor specialization shall be clearly mentioned by the respective BoS.

v. There shall be no limit on the number of programs offered under Minor, can offer minor programs in emerging technologies based on expertise in the respective departments or can explore the possibility of collaborating with the relevant industries/agencies in offering the program.

vi. The concerned BoS shall decide on the minimum enrolments for offering Minor program by the department. If a minimum enrolments criterion is not met, then the students may be permitted to register for the equivalent MOOC courses as approved by the concerned Head of the department in consultation with BoS.

vii. A student shall be permitted to register for Minors program at the beginning of 3$^{rd}$ / 4$^{th}$ semester subject to a maximum of two additional courses per semester, provided that the student must have acquired 8 SGPA (Semester Grade point average) upto the end of 2$^{nd}$ semester without any history of backlogs. It is expected that the 3$^{rd}$ semester results may be announced after the commencement of the 4$^{th}$ semester. *If a student fails to acquire 8 SGPA upto 3$^{rd}$ semester or failed in any of the courses, his registration for Minors program shall stand cancelled*. An SGPA of 8.0 has to be maintained in the subsequent semesters without any backlogs in order to keep the Minors registration active.

viii. A student shall earn additional 20 credits in the specified area to be eligible for the award of B. Tech degree with Minor. This is in addition to the credits essential for obtaining the Under Graduate Degree in Major Discipline (i.e. 160 credits).

ix. Of the 20 additional Credits to be acquired, 16/15 credits shall be earned by undergoing specified courses listed as pools, with four/five courses, each carrying 4/3 credits. The remaining 4/5 credits must be acquired through two MOOCs, which shall be domain specific, with 2/3 credits and with a minimum duration of 8/12weeks as recommended by the Board of studies.

x. Attendance will not be monitored for MOOC courses. Student has to acquire a certificate from the agencies approved by the BOS with grading or marks. If the MOOC course is a pass/fail course without any grades, the grade to be assigned as decided by the university/academic council.

xi. If a student drops (or terminated) from the Minor program, they cannot convert the earned credits into free or core electives; they will remain extra. These additional courses will find mention in the transcript (but not in the degree certificate). In such cases, the student may choose between the actual grade or a "pass (P)" grade and also choose to omit the mention of the course as for the following: All the courses done under the dropped Minors will be shown in the transcript. None of the courses done under the dropped Minor will be shown in the transcript.

xii. In case a student fails to meet the CGPA requirement for B.Tech degree with Minor at any point after registration, he/she will be dropped from the list of students eligible for degree with Minors and they will receive B. Tech degree only. However, such students will receive a separate grade sheet mentioning the additional courses completed by them.

xiii. Minor must be completed simultaneously with a major degree program. A student cannot earn the Minor after he/she has already earned bachelor's degree.

**11.** Minimum Instruction Days: The minimum instruction days for each semester shall be 90 working days.

**12.** There shall be no branch transfers after the completion of the admission process.

**13.** There shall be no transfer from one college/stream to another within the Constituent Colleges and Units of Jawaharlal Nehru Technological University Kakinada.

**14. 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 of the student will be withheld. His degree will be withheld in such cases.

**15. TRANSITORY REGULATIONS**

1. Discontinued or detained candidates are eligible for readmission as and when next offered.
2. The readmitted students will be governed by the regulations under which the candidate has been admitted or as per University/Institute norms.

**16. General**

1. Wherever the words "he", "him", "his", occur in the regulations, they include "she", "her", "hers".
2. The academic regulation should be read as a whole for the purpose of any interpretation.
3. In case of any doubt or ambiguity in the interpretation of the above rules, University R20 regulations can be followed and / or the decision of the Vice-Chancellor is final.
4. The University may change or amend the academic regulations or syllabi at any time and the changes or amendments made shall be applicable to all the students with effect from the dates notified by the University.

## ACADEMIC REGULATIONS (R20) FOR B. TECH. (LATERAL ENTRY SCHEME)

Applicable for the students admitted into II year B. Tech. from the Academic Year **2020-21** onwards

**1 Award of B. Tech. Degree**

A student will be declared eligible for the award of B. Tech. Degree if he fulfils the following academic regulations:

1.1 A student shall be declared eligible for the award of the B. Tech Degree, if he pursues a course of study in not less than three academic years and not more than six academic years.

1.2 The candidate shall register for **121 CREDITS** and secure all the credits.

2. The attendance regulations of B. Tech. (Regular) shall be applicable to B.Tech.

3. **Promotion Rule**

A student shall be promoted from second year to third year if he fulfills the minimum attendance requirement.

A student shall be promoted from III year to IV year if he fulfils the academic requirements of 50% of the credits from all the examinations up to III year I semester.

4. **Award of Class**

After a student has satisfied the requirement prescribed for the completion of the program and is eligible for the award of B. Tech. Degree, he shall be placed in one of the following four classes:

| Class Awarded | CGPA to be Secured | |
|---|---|---|
| First Class with Distinction | ≥ 7.5 without backlog history | From the CGPA secured from 121 Credits from II Year to IV Year |
| First Class | ≥ 6.5 < 7.5 without backlog history<br>≥ 6.5 with backlog history | |
| Second Class | ≥ 5.5 < 6.5 | |
| Pass Class | ≥ 4.0 < 5.5 | |

5. All the other regulations as applicable to **B. Tech. 4-year degree course (Regular) will hold good for B. Tech. (Lateral Entry Scheme)**.

## MALPRACTICES RULES

DISCIPLINARY ACTION FOR / IMPROPER CONDUCT IN EXAMINATIONS

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

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

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

### Malpractices identified by squad or special invigilators

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

# Ragging

## Prohibition of ragging in educational institutions Act 26 of 1997

### Salient Features

- Ragging within or outside any educational institution is prohibited.
- Ragging means doing an act which causes or is likely to cause Insult or Annoyance of Fear or Apprehension or Threat or Intimidation or outrage of modesty or Injury to a student

| | Imprisonment upto | | Fine Upto |
|---|---|---|---|
| Teasing, Embarrassing and Humiliation | 6 Months | + | Rs. 1,000/- |
| Assaulting or Using Criminal force or Criminal intimidation | 1 Year | + | Rs. 2,000/- |
| Wrongfully restraining or confining or causing hurt | 2 Years | + | Rs. 5,000/- |
| Causing grievous hurt, kidnapping or Abducts or rape or committing unnatural offence | 5 Years | + | Rs. 10,000/- |
| Causing death or abetting suicide | 10 Months | + | Rs. 50,000/- |

**In Case of Emergency CALL TOLL FREE NO. : 1800 - 425 - 1288**

**LET US MAKE JNTUK A RAGGING FREE UNIVERSITY**

# Ragging

# ABSOLUTELY
# NO TO RAGGING

**1. Ragging is prohibited as per Act 26 of A.P. Legislative Assembly, 1997.**

**2. Ragging entails heavy fines and/or imprisonment.**

**3. Ragging invokes suspension and dismissal from the College.**

**4. Outsiders are prohibited from entering the College and Hostel without permission.**

**5. Girl students must be in their hostel rooms by 7.00 p.m.**

**6. All the students must carry their Identity Cards and show them when demanded**

**7. The Principal and the Wardens may visit the Hostels and inspect the rooms any time.**

## Jawaharlal Nehru Technological University Kakinada

For Constituent Colleges and Affiliated Colleges of JNTUK

**In Case of Emergency CALL TOLL FREE NO. : 1800 - 425 - 1288**

**LET US MAKE JNTUK A RAGGING FREE UNIVERSITY**

Department of Computer Science & Engineering
University College of Engineering, JNT University Kakinada

# CURRICULUM STRUCTURE

# and

# DETAILED SYLLABI

for

Four Year UG Programme

## B. Tech.
## Computer Science & Engineering

*(Applicable for batches admitted from 2020)*

**Department of Computer Science & Engineering**
**University College of Engineering (A), JNTU Kakinada**

**R20- Bachelor of Technology**
**Course Structure & Syllabi**

Department of Computer Science & Engineering
University College of Engineering, JNT University Kakinada

# Vision and Mission of the Institute

## Vision

To be a premier institute of excellence developing highly talented holistic human capital that contributes to the nation through leadership in technology and innovation through engineering education.

## Mission

1. To impart Personnel Skills and Ethical Values for Sustainable Development of the Nation.
2. To create Research & Industry oriented centers of excellence in all engineering disciplines.
3. To be a renowned IPR generator and repository for innovative technologies.
4. To develop Research and Industry oriented technical talent.
5. To benchmark globally the academic & research output.

# Vision and Mission of the Computer Science and Engineering Department

## Vision

Department of Computer Science and Engineering strives rigorously to create intellectual academic environment with global standards that fosters the search for new knowledge in a highly dynamic computing-centric society through applied research.

## Mission

1. To Provide quality education in both theoretical and applied foundations of computer science and train the students to solve the real world problems effectively thus enhancing their potential for high quality careers.
2. To facilitate the students and faculty to inculcate the research culture to advance the state-of art of computer science and integrate research innovations in multi-disciplinary fields.
3. To equip students and faculty with excellent teaching-learning capabilities through advanced learning tools and technologies
4. To produce students with critical thinking and lifelong learning capabilities for applying their knowledge to uplift the living standards of the society.
5. To produce students with enriched skill set, professional behaviour, strong ethical values and leadership capabilities so as to work with commitment for the progress of the nation.

### Programme Education Objectives (PEOs)

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**PEO 1:** Competent IT professional with sound fundamental and applied knowledge in Computer Science and Engineering with socially acceptable, high-quality technical skills and life-long learning capabilities.

**PEO 2:** Sustained learner to bring out creative and innovative ideas to meet the challenges of industry and society by applying computer engineering principles and inter-disciplinary approach with ethical responsibilities and pursuing higher studies.

**PEO 3:** Entrepreneurs with interpersonal, managerial, ethical administrative skills to succeed in multidisciplinary fields and handle critical situations.

**PEO 4:** Researcher with skills to identify research gaps and engage in continuous learning to adapt to the diverse global environment and rapidly changing technological advancements with ethics.

## Programme Outcomes (POs)

**PO1: Engineering knowledge**: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

**PO2: Problem analysis**: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

**PO3: Design/development of solutions**: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for public health and safety and cultural, societal, and environmental considerations.

**PO4: Conduct investigations of complex problems**: Use research-based knowledge and research methods, including design of experiments, analysis, interpretation of data, and synthesis of the information to provide valid conclusions.

**PO5: Modern tool usage**: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools, including prediction and modeling to complex engineering activities with an understanding of the limitations.

**PO6: The engineer and society** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**PO7: Environment and sustainability**: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

**PO8: Ethics**: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

**PO9: Individual and team work**: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

**PO10: Communication**: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

**PO11: Project management and finance**: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's work as a member and leader in a team, to manage projects and in multidisciplinary environments.

**P012: Life-long learning**: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

## Program Specific Outcomes for B. Tech (CSE)

**PSO1:** Design algorithms for real-world computational problems and analyze the complexities to optimize solutions.

**PSO2:** Apply methods and tools for data acquisition, analysis, and knowledge discovery in developing decision-making systems.

**PSO3:** Develop adaptable and self-learning models with real-world data fitting, evaluation, and performance improvement methods.

**Department of Computer Science & Engineering**
**University College of Engineering, JNT University Kakinada**

## Mapping of Mission statements to PEOs

| Key components From Department Mission | PEO 1 | PEO 2 | PEO 3 |
|---|---|---|---|
| M1. To Provide quality education in both theoretical and applied foundations of computer science and train the students to solve the real world problems effectively thus enhancing their potential for high quality careers.<br><br>M2.To facilitate the students and faculty to inculcate the research culture to advance the state-of art of computer science and integrate research innovations in multi-disciplinary fields.<br><br>M3.To equip students and faculty with excellent teaching-learning capabilities through advanced learning tools and technologies<br><br>M4.To produce students with critical thinking and lifelong learning capabilities for applying their knowledge to uplift the living standards of the society.<br><br>M5.To produce students with enriched skill set, professional behaviour, strong ethical values and leadership capabilities so as to work with commitment for the progress of the nation. | Competent IT professional with sound fundamental and applied knowledge in Computer Science and Engineering. | Sustained learner to bring out creative and innovative ideas to meet the challenges of industry and society with ethics and human values and pursue the higher studies. | Entrepreneurs in computer science acquainted interpersonal, managerial skills to make them successful in multidisciplinary fields. |
| ***Quality education*** | **High** | **High** | **Medium** |
| ***Research*** | **Medium** | **High** | **Low** |
| **Teaching- Learning** | **Medium** | **High** | **Medium** |
| ***Sustained Learning*** | **High** | **High** | **Medium** |

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### Graduate Attributes (GAs)

| |
|---|
| Scholarship of Knowledge |
| Critical Thinking |
| Problem Solving |
| Research Skills |
| Usage of Modern Tools |
| Multidisciplinary Work |
| Project Management & Finance |
| Communication |
| Lifelong Learning |
| Ethical practices & Social Responsibilities |

### Mapping of Programme Outcomes to PEOs

| PEO | | PROGRAM OUTCOMES | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | PO 1 | PO 2 | PO 3 | PO 4 | PO 5 | PO 6 | PO 7 | PO 8 | PO 9 | PO 10 | PO 11 | PO 12 |
| **PEO 1** | 3 | 3 | 3 | 2 | 3 | 3 | 1 | 2 | 2 | 2 | | |
| **PEO 2** | 2 | 2 | 3 | 3 | 2 | 3 | 1 | 2 | 3 | 3 | | |
| **PEO 3** | 2 | 1 | 1 | 1 | 2 | 3 | 3 | 3 | 1 | 1 | | |

### Mapping of Programme Outcomes to GAs

| POs | Graduate Attributes |
|---|---|
| **PO1** | Scholarship of Knowledge |
| **PO2** | Critical Thinking |
| **PO3** | Problem Solving |
| **PO4** | Research Skills |
| **PO5** | Usage of Modern Tools |
| **PO6** | Multidisciplinary Work |
| **PO7** | Project Management & Finance |
| **PO8** | Communication |
| **PO9** | Lifelong Learning |
| **PO10** | Ethical practices & Social Responsibilities |

**Department of Computer Science & Engineering**
**University College of Engineering, JNT University Kakinada**

# B. Tech (R20) Programme Course Structure & Syllabus

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

| 1st B. Tech 1st semester | | | | | | |
|---|---|---|---|---|---|---|
| **Sl. No.** | **Course Code** | **Course Title** | **Hours per week** | | | **Credits** |
| | | | **L** | **T** | **P** | **C** |
| 1 | R20BS1101 | Mathematics - I | 3 | 0 | 0 | 3 |
| 2 | R20BS1102 | Applied Chemistry | 3 | 0 | 0 | 3 |
| 3 | R20HS1101 | Communicative English | 3 | 0 | 0 | 3 |
| 4 | R20ES1101 | Computer Engineering Workshop | 1 | 0 | 4 | 3 |
| 5 | R20ES1102 | Problem solving and Programming using C | 3 | 0 | 0 | 3 |
| 6 | R20HS1102 | English Communications skills Lab | 0 | 0 | 3 | 1.5 |
| 7 | R20BS1103 | Applied Chemistry Lab | 0 | 0 | 3 | 1.5 |
| 8 | R20ES1103 | Problem solving and Programming using C lab | 0 | 0 | 3 | 1.5 |
| 9 | R20MC1101 | Yoga & Physical Fitness Activities | 0 | 0 | 2 | 0 |
| | | | | | **Total credits** | **19.5** |

| 1st B. Tech 2nd semester | | | | | | |
|---|---|---|---|---|---|---|
| **Sl. No.** | **Course Code** | **Course Title (Proposed)** | **Hours per week** | | | **Credits** |
| | | | **L** | **T** | **P** | **C** |
| 1 | R20BS1201 | Mathematics – II | 3 | 0 | 0 | 3 |
| 2 | R20BS1202 | Applied Physics | 3 | 0 | 0 | 3 |
| 3 | R20ES1201 | Computer Organization | 3 | 0 | 0 | 3 |
| 4 | R20ES1202 | Data Structures | 3 | 0 | 0 | 3 |
| 5 | R20ES1203 | Object Oriented Programming Through C++ | 3 | 0 | 0 | 3 |
| 6 | R20ES1204 | Data Structures lab | 0 | 0 | 3 | 1.5 |
| 7 | R20BS1203 | Applied Physics Lab | 0 | 0 | 3 | 1.5 |
| 8 | R20ES1205 | Object Oriented Programming Through C++ lab | 0 | 0 | 3 | 1.5 |
| 9 | R20BS1204 | Applied Physics Virtual Laboratory | 0 | 0 | 2 | 0 |
| 10 | R20MC101 | Constitution of India (AICTE suggested) | 2 | 0 | 0 | 0 |
| 11 | R20PR1201 | Engineering Exploration Project- Design Thinking | 0 | 0 | 1 | 0 |
| | | | | | **Total credits** | **19.5** |

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

| 2nd B.Tech 1st semester | | | | | | |
|---|---|---|---|---|---|---|
| **Sl. No.** | **Course Code** | **Course Title** | **Hours per week** | | | **Credits** |
| | | | **L** | **T** | **P** | **C** |
| 1 | R20BS2101 | Mathematics – III | 3 | 0 | 0 | 3 |
| 2 | R20PC2101 | Mathematical foundations of Computer Science | 3 | 0 | 0 | 3 |
| 3 | R20PC2102 | Operating Systems | 3 | 0 | 0 | 3 |
| 4 | R20PC2103 | Advanced Java Programming | 3 | 0 | 0 | 3 |
| 5 | R20PC2104 | Data Base Management Systems | 3 | 0 | 0 | 3 |
| 6 | R20PC2105 | Data Base Management Systems lab | 0 | 0 | 3 | 1.5 |
| 7 | R20PC2106 | Operating Systems & LINUX Lab | 0 | 0 | 3 | 1.5 |
| 8 | R20PC2107 | Advanced Java Programming Lab | 0 | 0 | 3 | 1.5 |
| 9 | R20SO2101 | Mobile App Development | 0 | 0 | 4 | 2 |
| 10 | R20MC2101 | Research Methodology | 2 | 0 | 0 | 0 |
| | | | | | **Total credits** | **21.5** |

| 2nd B.Tech 2nd semester | | | | | | |
|---|---|---|---|---|---|---|
| **Sl. No.** | **Course Code** | **Course Title (Proposed)** | **Hours per week** | | | **Credits** |
| | | | **L** | **T** | **P** | **C** |
| 1 | R20BS2201 | Probability and Statistics | 3 | 0 | 0 | 3 |
| 2 | R20ES2201 | Formal Languages and Automata theory | 3 | 0 | 0 | 3 |
| 3 | R20PC2201 | Computer Networks | 3 | 0 | 0 | 3 |
| 4 | R20PC2202 | Python Programming | 3 | 0 | 0 | 3 |
| 5 | R20HS2201 | Managerial Economics and Finan Accountancy | 3 | 0 | 0 | 3 |
| 6 | R20PC2203 | R Programming lab | 0 | 0 | 3 | 1.5 |
| 7 | R20PC2204 | Python Programming Lab | 0 | 0 | 3 | 1.5 |
| 8 | R20PC2205 | Computer Networks Lab | 0 | 0 | 3 | 1.5 |
| 9 | R20 SO2201 | Python Programming: NLP | 0 | 0 | 4 | 2 |
| | | | | | **Total credits** | **21.5** |
| **Internship 2 Months (Mandatory) during summer vacation** | | | | | | |

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**III B. Tech – I Semester**

| S.No | Course Code | Courses | Hours per week | | | credits |
|---|---|---|---|---|---|---|
| | | | L | T | P | C |
| 1 | R20PC3101 | Compiler Design | 3 | 0 | 0 | 3 |
| 2 | R20PC3102 | Design analysis of Algorithms | 3 | 0 | 0 | 3 |
| 3 | R20PC3103 | Data Mining Techniques | 3 | 0 | 0 | 3 |
| 4 | R20OE3101<br><br>(OpenElective /job oriented) | **Open Elective-I**<br>Open Electives offered by other departments/Optimization Techniques(Job oriented course) | 3 | 0 | 0 | 3 |
| 5 | R20PE3101 | **Professional Elective-I**<br>1. Artificial Intelligence<br>2. Software project Management<br>3. Distributed Systems<br>4. Advanced Unix Programming | 3 | 0 | 0 | 3 |
| 6 | R20PC3104 | Data Mining Techniques with R Lab | 0 | 0 | 3 | 1.5 |
| 7 | R20PC3105 | Compiler Design Lab | 0 | 0 | 3 | 1.5 |
| 8 | R20SO3101 | Animation course: Animation Design | 0 | 0 | 4 | 2 |
| 9 | R20MC3101 | Employability Skills-I | 2 | 0 | 0 | 0 |
| 10 | R20PR3101 | **Summer Internship 2 Months(Mandatory) after second year(to be evaluated during V semester** | 0 | 0 | 0 | 1.5 |
| | | | | | **Total credits** | **21.5** |
| | | **Honors/Minor courses** | 4 | 0 | 0 | 4 |
| | | **Honors/Minor courses** | 4 | 0 | 0 | 4 |

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

| III B. Tech – II Semester | | | | | | |
|---|---|---|---|---|---|---|
| **S.No** | **Course Code** | **Courses** | **Hours per week** | | | **credits** |
| | | | **L** | **T** | **P** | **C** |
| 1 | R20PC3201 | Machine Learning | **3** | **0** | **0** | **3** |
| 2 | R20PC3202 | Big Data Analytics | **3** | **0** | **0** | **3** |
| 3 | R20PCC3203 | Cryptography Network Security | **3** | **0** | **0** | **3** |
| 4 | R20PE3201 | **Professional Elective-II**<br>1.Mobile Computing<br>2.Mean Stack Development<br>3.Object oriented Analysis and Design<br>4.Network Programming | **3** | **0** | **0** | **3** |
| 5 | R20OE3201<br>(Open Elective/ job oriented) | **Open Elective-II**<br>Open Electives offered by other departments/Natural Language Processing (Job Oriented Course) | **3** | **0** | **0** | **3** |
| 6 | R20PC3204 | Big Data Analytics lab | **0** | **0** | **3** | **1.5** |
| 7 | R20PC3205 | Machine Learning using Python Lab | **0** | **0** | **3** | **1.5** |
| 8 | R20PC3206 | Cryptography Network Security Lab | **0** | **0** | **3** | **1.5** |
| 9 | R20SO3201 | Data Science: MEAN Stack Technologies Module I-AngularJs, NodeJs, MongoDb/APSSDC suggested courses | **0** | **0** | **4** | **2** |
| 10 | R20MC3201 | Employability skills-II | **2** | **0** | **0** | **0** |
| | | | | | **Total credits** | **21.5** |
| **Industrial/Research Internship(Mandatory) 2 Months during summer vacation** | | | | | | |
| | | **Honors/Minor courses** | **4** | **0** | **0** | **4** |
| | | **Honors/Minor courses through SWAYAM** | **2** | **0** | **0** | **2** |

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

| IV B. Tech –I Semester | | | | | | |
|---|---|---|---|---|---|---|
| **S.No** | **Course Code** | **Course Title** | **Hours per week** | | | **Credits** |
| | | | **L** | **T** | **P** | **C** |
| 1 | R20PE4101 | **Professional Elective-III**<br>1.Cloud Computing<br>2.Neural Networks and Soft Computing<br>3.Ad-hoc and Sensor Networks<br>4.Cyber Security & forensics | **3** | **0** | **0** | **3** |
| 2 | R20PE4102 | **Professional Elective-IV**<br>1. Deep Learning Techniques<br>2. Social Networks &Semantic Web<br>3. DevOps<br>4.MOOCS-NPTEL/SWAYAM | **3** | **0** | **0** | **3** |
| 3 | R20PE4103 | **Professional Elective-V**<br>1.Block-Chain Technologies<br>2.Wireless Network Security<br>3.Ethical Hacking<br>4.MOOCS-NPTEL/SWAYAM | **3** | **0** | **0** | **3** |
| 4 | R20OE4101<br>(OpenElective/job oriented Skills) | **Open Elective-III**<br>Open Electives offered by other departments/<br>Secure Coding Techniques (Job Oriented Course) | **2** | **0** | **2** | **3** |
| 5 | R20OE4102<br>(OpenElective/job oriented Skills) | **Open Elective-IV**<br>Open Electives offered by other departments/<br>Computational Thinking (Job Oriented Course) | **2** | **0** | **2** | **3** |
| 6 | R20HSS4101 | Universal Human Values 2: Understanding Harmony | **3** | **0** | **0** | **3** |
| 7 | R20SO4101 | PYTHON: Deep Learning /APSSDC offered Courses | **0** | **0** | **4** | **2** |
| 8 | R20PR4101 | **Industrial/Research Internship 2 months (Mandatory) after third year (to be evaluated during VII semester** | **0** | **0** | **0** | **3** |
| **Total credits** | | | | | | **23** |
| **Honors/Minor courses** | | | **4** | **0** | **0** | **4** |
| **Honors/Minor courses through SWAYAM** | | | **2** | **0** | **0** | **2** |

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

| 4th B. Tech 2nd sem | | | | | | |
|---|---|---|---|---|---|---|
| **Sl. No.** | **Course Code** | **Course Title (Proposed)** | **Hours per week** | | | **Credits** |
| | | | **L** | **T** | **P** | **C** |
| 1 | R20PR4201 | Project<br>Project work, seminar and internship in industry | 0 | 0 | 0 | 12 |
| | | | | | **Total credits** | **12** |

| Component | Remarks | APSCHE | JNTUK |
|---|---|---|---|
| HSSMS | English, MEFA, MS | **7.5+1.5(HSS Elective)** | **9** |
| BS | Maths, Physics and Chemistry | **21** | **21** |
| ES | Workshop, Drawing, Basics of Electrical/Mechanical/Computers | **22.5** | **22.5** |
| PC | Departmental Core **(PCC)** | **52.5** | **52.5** |
| PE | Departmental Electives **(PE)** | **15** | **15** |
| OE | Open Electives | **12** | **12** |
| Proj | Project Work, Internships, Seminar, APSSDC etc | **16.5** | **16.5** |
| **Skill** | | **10** | **10** |
| **Total Credits** | | **160** | **160** |
| **Minor's** | | **20(4*4+2+2)** | **20** |
| **Honor's** | | **20** | **20** |

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

***Open Electives to be offered by CSE for Other Branches:***

| **Open Elective-I:** | **Open Elective-II:** |
|---|---|
| 1. Data Structures<br>2. Java Programming<br>3. Data Base Management Systems<br>4. Computer Graphics<br>5. C++ Programming<br>6. Advanced unix programming | 1. Operating Systems<br>2. Python Programming<br>3. Web Technologies<br>4. Soft Computing<br>5. Distributed Computing<br>6. AI and ML for Robotics |
| **Open Elective-III:**<br>1. Big Data Analystics<br>2. AI Tools & Techniques<br>3. Image Processing<br>4. Information Security<br>5. Mobile Application Development<br>6. Cloud computing | **Open Elective-IV:**<br>1. Cyber Security<br>2. Deep Learning<br>3. Data Science<br>4. Block Chain Technologies<br>5. Game Theory<br>6. Internet of Things |
| | |

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

***Open Electives to be offered by Other Departments for Computer Science & Engineering***

| | |
|---|---|
| **Electronics &Communication Engineering**<br><br>1. Information Coding Theory<br>2. VLSI<br>3. Signals &Systems<br>4. Digital Signal Processing<br>5. Medical Image Processing<br>6. RFID, Sensors &Data Acquisition | **Mathematics**<br><br>1. Optimization Techniques<br>2. Statistics with R<br>3. Cryptography, number theory and Cryptanalysis<br>4. Fuzzy Sets, Logic and Systems |
| **Electronics and Electronics Engineering**<br><br>1. Network Analysis<br>2. Fuzzy Systems & Controllers<br>3. Green Energy Models<br>4. Power Systems for Data Centers<br>5. Power Safety and Management | **Civil Engineering**<br><br>1. Intelligent transportation Engineering<br>2. Geospatial Systems(GIS, Remote Sensing etc.,)<br>3. Engineering Mechanics<br>4. Smart City Planning<br>5. Smart & Safety Building Design |
| **Mechanical Engineering**<br><br>1. Industrial Management<br>2. Robotics and Autonomous Driving Systems<br>3. CAD and MATLAB<br>4. Basics of Mechatronics<br>5. Alternative Energy Systems | |

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### HONORS COURSES

**(Tentative list be appended)**

**Note**

1. **The subjects opted for HONORS should be Advanced type which are not covered in regular curriculum**
2. **Students have to acquire 16 credits with minimum one subject from each pool.**
3. **Concerned BoS can add or delete the subjects as per the decision of the board.**
4. **Pre requisites to be defined by the board for each course.**
5. **Compulsory MOOC/NPTEL Courses for 04 credits (02 courses@ 2 credits each)**

| POOL1 | POOL 2 |
|---|---|
| 1. Network Architecture and Design<br>2. Operating Systems Administration and Security<br>3. E-Commerce<br>4. Software Architecture and Design Patterns | 1. Neural Networks<br>2. Security Governance Risk and compliance<br>3. Randomized algorithms<br>4. Data Visualization |
| **POOL3** | **POOL4** |
| 1. Cloud and IoT Security<br>2. No SQL Databases<br>3. Soft computing<br>4. Social Networks & Semantic Web | 1. Digital Marketing<br>2. Wireless Networks Security<br>3. High performance Computing<br>4. Multi Agent Systems |

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### MOOCs - SWAYAM/NPTEL Honor's Course list for 04 credits (02 courses@ 2 credits each)

### (Tentative - list can be appended)

**Note: Students need to select the courses which were not offered in their regular programme and minor tracks**

| S.NO | COURSE NAME | UG-Core/Elective | Type of Course |
|---|---|---|---|
| 1 | User-centric Computing for Human Computer Interaction | Elective | Rerun |
| 2 | Information Security - 5 - Secure Systems Engineering | Elective | Rerun |
| 3 | AI: Constraint Satisfaction | Elective | Rerun |
| 4 | Privacy and Security in Online Social Media | Elective | Rerun |
| 5 | Data Science for Engineers | Elective | Rerun |
| 6 | Embedded System Design with ARM | Elective | Rerun |

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### GENERAL MINOR TRACKS

**Note:**

1. **The student can opt any 4 subjects from each pool.**
2. **Concerned BoS can add or delete the subjects as per the decision of the board.**
3. **Pre requisites to be defined by the board for each course.**
4. **Compulsory MOOC/NPTEL Courses for 04 credits (02 courses@ 2 credits each)**

**Department of Mechanical Engineering**

5. **Students need to select the courses which were not offered in their regular programme.**

| S. No | Subject | L-T-P | Credit |
|---|---|---|---|
| 1 | Operating systems | 3-1-0 | 4 |
| 2 | Data Structures Using C | 3-1-0 | 4 |
| 3 | Computer organization and Architecture | 3-1-0 | 4 |
| 4 | Software Engineering | 3-1-0 | 4 |
| 5 | Design and analysis of algorithms | 3-1-0 | 4 |
| 6 | Computer Networks | 3-1-0 | 4 |
| 7 | Database Management Systems | 3-1-0 | 4 |
| 8 | Object oriented programming | 3-1-0 | 4 |
| 9 | Data warehousing and Data mining | 3-1-0 | 4 |
| 10 | Data Science | 3-1-0 | 4 |
| 11 | Artificial Intelligence | 3-1-0 | 4 |

# Department of Computer Science & Engineering
# University College of Engineering, JNT University Kakinada

MINOR courses for SPECIALIZED TRACKS

**Note: 1. A student can opt Four subjects from any one of the specialized track @ 4 credits per subject**
**2. Concerned BoS can add or delete the subjects as per the decision of the board.**
**3. Pre requisites to be defined by the board for each course.**
**4. Compulsory MOOC/NPTEL Courses for 04 credits (02 courses@ 2 credits each)**
**5. Students need to select the courses which were not offered in their regular programme.**

| S.NO. | COURSE NAME | L-T-P | CR | PRE-REQ. | OFFERED TO |
|---|---|---|---|---|---|
| | **TRACK-1 NETWORKING & SECURITY** | | | | |
| 1 | TCP/IP Protocol Suite | 3-1-0 | 4 | | CSE/IT |
| 2 | Network Architecture and Design | 3-1-0 | 4 | | CSE/IT |
| 3 | Network Security | 3-1-0 | 4 | | CSE/IT |
| 4 | Cryptography | 3-1-0 | 4 | | CSE/IT |
| 5 | Computer Forensics | 3-1-0 | 4 | | CSE/IT |
| 6 | ethical hacking | 3-1-0 | 4 | | |
| | **TRACK-2 SOFTWARE ENGINEERING** | | | | |
| 1 | Software Metrics and Measurements | 3-1-0 | 4 | | CSE/IT |
| 2 | Software Verification and Validation | 3-1-0 | 4 | | CSE/IT |
| 3 | Software Architecture and Design Patterns | 3-1-0 | 4 | | CSE/IT |
| 4 | Software Project Management | 3-1-0 | 4 | | CSE/IT |
| 5 | Fault Tolerant Computing | 3-1-0 | 4 | | CSE/IT |
| 6 | Software Testing Methodologies | 3-1-0 | 4 | | |
| | **TRACK-3 DISTRIBUTED & CLOUD COMPUTING** | | | | |
| 1 | Enterprise Storage Systems | 3-1-0 | 4 | | CSE/IT |
| 2 | Parallel Algorithms | 3-1-0 | 4 | | CSE/IT |
| 3 | Cloud Networking | 3-1-0 | 4 | | CSE/IT |
| 4 | Cloud Computing | 3-1-0 | 4 | | CSE/IT |
| 5 | High Performance Computing | 3-1-0 | 4 | | CSE/IT |
| 6 | Advanced Computer Architecture | 3-1-0 | 4 | | |
| | **TRACK-4 COMPUTATIONAL INTELLIGENCE** | | | | |
| 1 | Artificial intelligence | 3-1-0 | 4 | | CSE/IT |
| 2 | Machine Learning | 3-1-0 | 4 | | CSE/IT |
| 3 | Natural Language Processing | 3-1-0 | 4 | | CSE/IT |
| 4 | Neural Networks | 3-1-0 | 4 | | CSE/IT |
| 5 | Multi Agent Systems | 3-1-0 | 4 | | CSE/IT |
| 6 | Deep Learning | 3-1-0 | 4 | | |

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**MOOCs- SWAYAM/NPTEL Minor Course list for 04 credits (02 courses@ 2 credits each)**

**(Tentative- list can be appended)**

**Note: Students need to select the courses which were not offered in their regular programme and minor tracks**

**LIST (ONLY 8 WEEKS)**

| S.NO | COURSE NAME | UG-Core/Elective | Type of Course |
|---|---|---|---|
| 1. | Programming in C++ | Core | Rerun |
| 2. | Data Base Management System | Core | Rerun |
| 3. | Machine Learning, ML | Elective | Rerun |
| 4. | Programming, Data Structures And Algorithms Using Python | Elective | Rerun |
| 5. | Design and analysis of algorithms | Elective | Rerun |
| 6. | Advanced Computer Architecture | Elective | Rerun |
| 7. | Advanced Graph Theory | Core/Elective | Rerun |
| 8. | Cloud Computing and Distributed Systems | Core/Elective | Rerun |
| 9. | Introduction to Soft Computing | Elective | Rerun |
| 10. | Cloud computing | Elective | Rerun |
| 11. | Data Mining | Elective | Rerun |

**Department of Computer Science & Engineering**
**University College of Engineering, JNT University Kakinada**

### Socially Relevant Projects (15 Hrs)

1. Water C/on servation Related Works
2. Swatch Bharath (Internal External)
3. Helping police
4. Traffic monitoring
5. Teaching Rural Kids (Sarva siksha Abhiyan)
6. Street light monitoring
7. Electricity Conservation
8. Solar panel utilization
9. E- policing & cyber solution
10. Pollution
11. Smart city
12. COVID awareness
13. Road Safety
14. Any suggested service project from APSCHE list

**Department of Computer Science & Engineering**
**University College of Engineering, JNT University Kakinada**

## I B. Tech I sem-R20

## COMPUTER ENGINEERING WORKSHOP
## Code: R20ES1101

**COURSE OBJECTIVES:** Skills and knowledge provided by this subject are the following:

- **PC Hardware:** Identification of basic peripherals, Assembling a PC, Installation of system software like MS Windows, device drivers, etc. Troubleshooting of PC Hardware and Software issues.
- **Internet & World Wide Web:** Different ways of hooking the PC on to the internet from home and workplace and effectively usage of the internet, web browsers, email, newsgroups and discussion forums. Awareness of cyber hygiene (protecting the personal computer from getting infected with the viruses), worms and other cyber attacks.
- **Productivity Tools:** Understanding and practical approach of professional word documents, excel spread sheets, power point presentations and personal web sites using the Microsoft suite office tools.

**COURSE OUTCOMES:** By the end of the course student will be able to

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| CO1 | Identify, assemble and update the components of a computer | K3,K5,K6 |
| CO2 | Configure, evaluate and select hardware platforms for the implementation and execution of computer applications, services and systems | K1, K3,K5 |
| CO3 | Make use of tools for converting pdf to word and vice versa | K3 |
| CO4 | Develop presentation, documents and small applications using productivity tools such as word processor, presentation tools, spreadsheets, HTML, LaTex | K3,K6 |

**#Based on suggested Revised BTL**

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 |
|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 2 | 2 | 2 | 3 | 1 | | | | 1 | |
| CO2 | 3 | 2 | 3 | 3 | 2 | 1 | | 2 | 1 | |
| CO3 | 2 | 2 | | | 3 | 2 | | 1 | 2 | |
| CO4 | 1 | | | 2 | 3 | 2 | | 3 | 2 | |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

Department of Computer Science & Engineering
University College of Engineering, JNT University Kakinada

**SYLLABUS:**

**UNIT I: Introduction to Computer:** Introduction, Digital and Analog Computers, Characteristics of a Computer, History, Classification and Applications of Computers. **Computer System Hardware:** Central Processing Unit, Memory Unit, Interconnecting the Units of a Computer, Performance of a Computer, Inside a Computer Cabinet.
**Computer Memory:** Memory hierarchy, Primary and secondary memory.

**UNIT II: Input and Output Devices:** Introduction, Human data entry devices, Source data entry devices, Output devices, I/O ports
**Interaction of user and computer:** Types of software, System software, Application Software. **Operating System:** Objectives, Types and Functions of operating system, examples of Operating systems: MS-DOS, Windows family and Linux OS

**UNIT III: Networking:** Importance of networking, transmission media, Network Types, LAN Topologies, Network devices, Wireless Networking
**The Internet and Internet services:** Internet architecture, Internet Connections, Internet Services: WWW, E-mail, FTP, Telnet
**Computer Security:** Security threat and attack, Malicious software, hacking, firewall, user identification and authentication
**Network and Internet Connections:** Start network connections, create and manage a connection, windows firewall, connect using VPN.

**UNIT IV: Office suite: MS-Word:** Introduction, MS-WORD screen and its components, examples.
**MS-Excel:** Introduction, basics of spreadsheet, MS-Excel screen and its components, examples.
**MS-PowerPoint:** Introduction, basics, MS-PowerPoint screen and its components, examples.

**UNIT V: HTML:** basic HTML tags, Introduction to HTML5 and its tags, Introduction to CSS3 and its properties. Preparation of a simple website/ homepage.
**Using LaTeX:** Introduction, Create LaTeX document, Components of a LaTeX Document, LaTeX Commands, Examples.
**Internet of Things (IoT)**: IoT fundamentals, applications, protocols, communication models, architecture, IoT devices

**List of Exercises:**

**Note: Faculty to consolidate the workshop manuals using the textbook and references**

**Task 1: Identification of the peripherals of a computer -** Prepare a report containing the block diagram of the computer along with the configuration of each component and its functionality. Describe about various I/O Devices and its usage.

**Task 2:** Practicing disassembling and assembling components of a PC

**Task 3:** Installation of Device Drivers, MS Windows, Linux Operating systems and Disk Partitioning, dual boating with Windows and Linux

**Task 4:** Introduction to Memory and Storage Devices, I/O Port, Assemblers, Compilers, Interpreters, Linkers and Loaders.

**Task 5:** Demonstration of Hardware and Software Troubleshooting

**Task 6**: Demonstrating Importance of Networking, Transmission Media, Networking Devices- Gateway, Routers, Hub, Bridge, NIC, Bluetooth Technology, Wireless Technology, Modem, DSL, and Dialup Connection.

**Task 7**: Surfing the Web using Web Browsers, Awareness of various threats on the Internet and its solutions, Search engines and usage of various search engines, Need of anti-virus, Installation of anti-virus, configuring personal firewall and windows update.
(Students should get connected to their Local Area Network and access the Internet. In the process they should configure the TCP/IP setting and demonstrate how to access the websites and email. Students customize their web browsers using bookmarks, search toolbars and pop up blockers)

**Productivity Tools:**

**Task 8:** basic HTML tags, Introduction to HTML5 and its tags, Introduction to CSS3 and its properties. Preparation of a simple website/ homepage,
*Assignment:* Develop your home page using HTML Consisting of your photo, name, address and education details as a table and your skill set as a list.
Features to be covered:- Layouts, Inserting text objects, Editing text objects, Inserting Tables, Working with menu objects, Inserting pages, Hyper linking, Renaming, deleting, modifying pages, etc.,

**Task 9**: Demonstration and Practice of various features of Microsoft Word
Assignment: 1. Create a project certificate.
2. Creating a news letter

Features to be covered:-Formatting Fonts, Paragraphs, Text effects, Spacing, Borders and Colors, Header and Footer, Date and Time option, tables, Images, Bullets and Numbering, Table of Content, Newspaper columns, Drawing toolbar and Word Art and Mail Merge in word etc.,

**Task 10:** Demonstration and Practice of various features Microsoft Excel
Assignment: 1. Creating a scheduler
2. Calculating GPA
3. Calculating Total, average of marks in various subjects and ranks of students based on marks

Features to be covered:- Format Cells, Summation, auto fill, Formatting Text, Cell Referencing, Formulae in excel, Charts, Renaming and Inserting worksheets, etc.,

**Task 11:** Demonstration and Practice of various features Microsoft Power Point Features to be covered:- Slide Layouts, Inserting Text, Word Art, Formatting Text, Bullets and Numbering, Auto Shapes, Hyperlinks Tables and Charts, Master Layouts, Types of views, Inserting – Background, textures, Design Templates, etc.,

**Task 12:** Demonstration and Practice of various features of LaTeX – document preparation, presentation (Features covered in Task 9 and Task 11 need to be explored in LaTex)

**Task 13:** Tools for converting word to pdf and pdf to word

**Task 14: Internet of Things (IoT)**: IoT fundamentals, applications, protocols, communication models, architecture, IoT devices

**REFERENCE BOOKS:**

1. Computer Fundamentals, Anita Goel, Pearson India Education, 2017
2. PC Hardware Trouble Shooting Made Easy, TMH
3. Upgrading and Repairing PCs, 18th Edition, Scott Mueller, QUE, Pearson, 2008
4. *LaTeX Companion – Leslie Lamport, PHI/Pearson*
5. Introducing HTML5, Bruce Lawson, Remy Sharp, 2nd Edition, Pearson, 2012
6. Teach yourself HTML in 24 hours, By Techmedia
7. HTML 5 and CSS 3.0 to the Real World by Alexis Goldstein, Sitepoint publication.
8. Internet of Things, Technologies, Applications, Challenges and Solutions, B K Tripathy, J Anuradha, CRC Press
9. Comdex Information Technology Course Tool Kit, Vikas Gupta, Wiley Dreamtech.
10. *IT Essentials PC Hardware and Software Companion Guide Third Edition by David Anfinson and Ken Quamme, CISCO Press, Pearson Education.*
11. Essential Computer and IT Fundamentals for Engineering and Science Students, Dr. N. B. Venkateswarlu, S. Chand Publishers

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**I B. Tech I sem-R20**

**PROBLEM SOLVING AND PROGRAMMING USING C**
**Code: R20ES1102**

**COURSE OUTCOMES**: At the end of the course, student will be able to

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Develop efficient algorithm for solving a problem. | **K3** |
| **CO2** | Experiment with various constructs of C programming language efficiently. | **K3** |
| **CO3** | Develop programs using modular approach such as functions and also able to develop programs to perform matrix and mathematical applications. | **K3** |
| **CO4** | Examine dynamic memory management, problems using pointers and solving the problems. | **K4** |
| **CO5** | Develop programs for real-life applications using Structures and also learn about handling the files for storing the data permanently. | **K3** |

***#Based on suggested Revised BTL***

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 3 | | 3 | 2 | | 2 | | 2 | 1 | | | 2 |
| **CO2** | | 2 | | 2 | 3 | | 2 | 3 | | | 1 | |
| **CO3** | 1 | | 3 | | | 3 | | 2 | | 2 | | |
| **CO4** | | 2 | 1 | | | | 2 | | 2 | | | 2 |
| **CO5** | | | 2 | | 2 | | | 3 | 1 | | 2 | |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**SYLLABUS:**

**UNIT – 1: Problem Solving:** Problem solving aspects, problem solving techniques, computer as a problem solving tool, Characteristics of Complex problem Solving methods, Collective Problem Solving, Collaborative Problem Solving, Software Development Life Cycle, algorithms-definition, features,, criteria, Flowchart definition, basic symbols, sample flowcharts, Top down design, Implementation of program verification.

**UNIT – 2: Basics of C Programming Language:** Introduction to C, Structure of a C program, Basic data types and sizes, constants, variables, Unary, Binary and Ternary operators, Expressions, Type conversions, Conditional Expressions, Precedence and Order of Evaluation, Input and Output Statements, Header files, C pre-processors, sample programs. **Selection or Branching:** Simple if, if-else, nested if, if-else ladder and switch-case, goto, break and continue statements. **Iterative or Looping:** While loop, Do-while loop and for loop, sample programs.

**UNIT – 3: Functions:** Basics, Parameter passing, storage classes, scope rules, block structure, user-defined functions, standard library functions, Recursive

functions, Recursive solutions for Fibonacci series and Towers of Hanoi. **Arrays:** Declaration, definition, Accessing elements, types of Arrays, Array applications, Matrix operations, passing 1-D Arrays and 2-D Arrays to functions.
**Strings:** Basics, string operations, string manipulation functions.

**UNIT – 4: Pointers:** Basics, Initialization of Pointer variables, Pointers and function arguments, passing by Address Dangling memory, Address Arithmetic, character pointers and functions. Pointer to Pointer, Pointers and Multi-Dimensional Arrays, Dynamic Memory Management functions, Command-line Arguments.
**Notations:** The efficiency of algorithms, Analysis of Algorithms, computational complexity of algorithms, order(O), notation, Worst-case and Average-case analysis.
**UNIT – 5: Enumerated, Structure and Union Types:** Derived Types, Structure declaration, definition and initialization, accessing Structures, nested Structures, Arrays of Structures, Structures and Functions, Pointers to Structures, Self referential Structures, Unions, typedef, bit-fields, program applications.
**File Handling:** Concepts of Files, Text files and Binary Files, formatted I/O, File I/O operations.

**TEXT BOOKS**:

1. How to Solve it by Computer, R.G.Dromey, Pearson Education, 2019.
2. Programming in C, Ashok N Kamthane, Amit Ashok Kamthane, 3rd Edition, Pearson Education, 2019.

**REFERENCE BOOKS**:

1. The C Programming Language by Dennis Richie and Brian Kernighan.
2. Programming in C, Reema Thareja, OXFORD.
3. C Programming, A Problem Solving Approach, Forouzan, Gilberg, Prasad, Cengage.

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**I B. Tech I sem-R20**

### PROBLEM SOLVING AND PROGRAMMING USING C LAB
### Code: R20ES1103

**COURSE OBJECTIVE:**

- Understand Algorithm and Syntax create Mathematical Operations in C.
- Handle Strings and Files in C.
- Understand Arrays, Pointers and Recursion in C.
- Implement Dynamic and Modular programming in C
- Learn Functions and Storage classes in C.

**COURSE OUTCOMES:**

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Examine C syntax, structure and be fluent in the use of C keywords and looping. | **K4** |
| **CO2** | Demonstrate proficiency in handling Strings and File Systems. | **K2** |
| **CO3** | Construct Matrixes creation and operations Programs using Arrays, structures like Dynamic programming. | **K3** |
| **CO4** | Interpret the concepts of Recursion Programming as used in C. | **K2** |
| **CO5** | Construct C programs using Pointers and Functions, various call by reference. | **K3** |

***Based on suggested Revised BTL***

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 3 | | 2 | 3 | 2 | 3 | | 2 | 3 | 3 | | 1 |
| **CO2** | 2 | 1 | | | 3 | | 3 | 3 | | | | 3 |
| **CO3** | 3 | | 2 | | 2 | | | 2 | 2 | | 2 | 1 |
| **CO4** | 3 | | 1 | 2 | 2 | | | 2 | 2 | 1 | 3 | |
| **CO5** | 2 | 1 | | | 2 | 1 | 2 | 3 | 2 | 2 | | |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**Department of Computer Science & Engineering**
**University College of Engineering, JNT University Kakinada**

**List of Exercises:**

**Exercise 1:**

a. Write a C program to calculate the area of a triangle.
b. Write a C program to find the largest of three numbers using ternary operator.
c. Write a C program to swap two numbers without using temporary variable.

**Exercise 2:**

a. Write a C program to find the 2's complement of a binary number.
b. Write a C program to find the roots of a quadratic equation.
c. Write a C program to implement simple calculator using switch statement.

**Exercise 3:**

a. Write a C program to find the sum of individual digits of a positive integer and also find the reverse of the given number.
b. Write a C program to generate the first n terms of the Fibonacci sequence.
c. Write a C program to generate all the prime numbers between 1 and n,where n is a value supplied by the user.

**Exercise 4:**

a. Write a C program to print the Multiplication table of a given number.
b. Write a C program to read a decimal number and find it's equivalent binary number.
c. Write a C program to check whether the given number is Armstrong number or not.

**Exercise 5:**

a. Write a C program to interchange the largest and smallest numbers in the given array.
b. Write a C program to implement Towers of Hanoi.

**Exercise 6:**

a. Write a C program to implement sorting an array of elements.
b. Write a C program to implement matrix addition and multiplication.
c. Write a C program to print the upper case matrix using Arrays.

**Exercise 7:**

Write a C program that uses functions to perform the following operations.

a. To insert a sub string into given main string at a given position.
b. To delete n characters from a given position in a given string.
c. To replace a character of a string either from beginning or ending or at a specified location.

**Exercise 8:**

Write a C program that uses functions to perform the following operations using Structure:

a. Reading a complex number
b. Writing a complex number
c. Addition of two complex numbers
d. Multiplication of two complex numbers

**Exercise 9:**
Write a C program for the following string operations without using the built-in functions.
a. To concatenate two strings
b. To append a string to another string
c. To compare two strings

**Exercise 10:**
a. Write a C program to find the number of characters in a given string including and excluding spaces.
b. Write a C program to copy the contents of one string to another string without using string handling functions.
c. Write a C program to find whether a given string is palindrome or not.

**Exercise 11:**
Write a C program using recursion for the following:
a. To display sum of digits of a given number
b. To find the factorial of a given integer
c. To find the GCD (Greatest Common Divisor) of two given integers.
d. To find Fibonacci sequence.

**Exercise 12:**
a. Write a C program to reverse a string using pointers.
b. Write a C program to compare two 2D arrays using pointers.
c. Write a C program consisting of Pointer based function to exchange value of two integers using passing by address.

**Exercise 13:**
a. Write a C program to find both the largest and smallest number of an array of integers using call by value and call by reference.
b. Write a C program to implement student details using Structures.

**Exercise 14:**
a. Write a C program which copies one file to another.
b. Write a C program to count the number of characters and number of lines in a file.
c. Write a C program to merge two files into a third file. The names of the files must be entered using command line arguments.

**Exercise 15:**
Write a C program to implement Different Storage classes.
a. Auto
b. Static
c. Register
d. External

**Department of Computer Science & Engineering**
**University College of Engineering, JNT University Kakinada**

**I B. Tech II sem-R20**

## COMPUTER ORGANIZATION
**Code: R20ES1201**

**COURSE OBJECTIVES:**
The purpose of the course is to introduce principles of computer organization and the basic architectural concepts. It provides an in depth understanding of basic organization, design, programming of a simple digital computer, computer arithmetic, instruction set design, micro programmed control unit, pipelining and vector processing, memory organization and I/O systems

**COURSE OUTCOMES:** By the end of the course student will be able to

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Demonstrate an understanding of the different number systems, codes and Relate Postulates of Boolean algebra and minimize combinational functions | **K1, K2** |
| **CO2** | Evaluate and learn different combinational circuits, sequential circuits and able to design them | **K5** |
| **CO3** | Organize, Determine and learns basic structure of components register through language, micro operations and able to write micro programs | **K3, K5** |
| **CO4** | Determine and able to write data transfer and manipulators program and students able to learn microprogramme control and central processing unit | **K5** |
| **CO5** | Able to learns the internal organization of computers and able to evaluate performance of them. | **K1, K5** |

**#based on suggested Revised BTL**
**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 2 | | 3 | 1 | | | | | | | | |
| **CO2** | 2 | 3 | 3 | 1 | | | | | | | | |
| **CO3** | 2 | 2 | 1 | 3 | | 1 | | 1 | | | | |
| **CO4** | | 3 | 2 | | | | | 1 | | | | |
| **CO5** | 1 | 2 | 3 | | | 2 | | | | | | |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**SYLLABUS:**

**UNIT I: Number System and Data Representation**: Introduction, Numbering Systems, Decimal to Binary Conversion, Binary Coded Decimal Numbers, Weighted Codes, Self-Complementing Codes, Cyclic Codes, Error Detecting Codes, Error Correcting Codes, Hamming Code for Error Correction, Alphanumeric Codes, ASCII Code

**Boolean algebra and Logic gates :** Boolean expressions and their minimization using algebraic identities; Karnaugh map representation and minimization of Boolean functions using K-maps up to 4-variable;Don't care conditions, Digital Logic gates,Two-level realizations using gates -- AND-OR, OR-AND, NAND-NAND and NOR-NOR

**UNIT II: Combinational logic circuits-I:** Design of Half adder, full adder, half subtractor, full subtractor, Design of decoder, Demultiplexer, higher order demultiplexing, encoder, multiplexer, higher order multiplexer, realization of Boolean functions using decoders and multiplexers, priority encoder.

**Sequential circuits I:** Classification of sequential circuits (synchronous and asynchronous): basic flip-flops, truth tables and excitation tables (NAND RS latch, NOR RS latch, RS flip-flop. JK flip-flop, T flip-flop, D flip-flop with reset and clear terminals).Conversion of flip-flop to flip-flop, Race around condition, Master J-K flip-flop

**Unit-III: Basic Structure Of Computers:** Computer Types, Functional unit, Basic Operational concepts, Bus structures, Software, Performance, multiprocessors and multi computers. Data Representation. Fixed Point Representation. Floating – Point Representation.

**Register Transfer Language And Micro-operations:** Register Transfer language. Register Transfer Bus and memory transfers, Arithmetic Micro-operations, Logic micro operations, shift micro operations, Arithmetic logic shift unit. Instruction codes. Computer Registers, Computer instructions, Instruction cycle.

**UNIT IV: Micro Programmed Control:** Control memory, Address sequencing, micro program example, design of control unit.

**Central Processing Unit:** General Register Organization, Instruction Formats, Addressing modes, Data Transfer and Manipulation, Program Control.

**UNIT V: Memory Organization:** Memory Hierarchy, Main Memory, Auxiliary memory, Associate Memory, Cache Memory, Cache memories performance considerations, Virtual memories Introduction to Shift registers and RAID

**Input –Output Organization** Peripheral Devices, Input-Output Interface, Asynchronous data transfer, Modes of Transfer, Priority Interrupts, DMA, Input Output Processor, Serial Communication.

**TEXT BOOKS:**

1. Digital Logic and Computer Design, Moriss Mano, 11th Edition, Pearson Education.
2. Computer Organization, 5$^{th}$ed.,Hamacher, Vranesicand Zaky,TMH,2002
3. Computer System Architecture, 3/e, MorisMano,Pearson/PHI.

**REFERENCE BOOKS:**

1. Computer System Organization &Architecture, John D.Carpinelli, Pearson, 2008
2. Computer System Organization, NareshJotwani, TMH, 2009
3. Computer Organization &Architecture: Designing for Performance, 7$^{th}$ed.,William Stallings, PHI, 2006
4. Structured Computer Organization – Andrew S. Tanenbaum, 4th Edition, PHI/Pearson.

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**I B. Tech II sem-R20**

**DATA STRUCTURES**
**Code: R20ES1202**

**COURSE OBJECTIVES:**

- Solve problems using data structures such as linear lists, stacks, queues, hash tables
- Be familiar with advanced data structures such as balanced search trees, AVL Trees, and B Trees.

**COURSE OUTCOMES:** By the end of the course student will be able to understand

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Select appropriate data structures as applied to specified problem definition | **K5** |
| **CO2** | Summarize and understand the practical applications of several advanced techniques like Hashing and Analyzing and Implement appropriate sorting/searching technique for given problems | **K2** |
| **CO3** | Demonstrate the operations such as Insertion, Deletion and Search on Data structures like Binary Search Tree and solve the problems | **K3** |
| **CO4** | Demonstrate the operations such as Insertion, Deletion and Search on Advanced Data structures like Heaps, AVL trees and B Trees. | **K3** |
| **CO5** | Comparisons of trees like Red Black trees and B-Trees etc. and priority queue operations. | **K4** |

**#based on suggested Revised BTL**

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 3 | 2 | 2 | | | | | | | | | |
| CO2 | 2 | | 3 | | | | | 3 | | | | |
| CO3 | 1 | 2 | 3 | | 3 | | | | | | | |
| CO4 | | | | 2 | 3 | 1 | | 3 | | | | |
| CO5 | 2 | | | | | | | 2 | | | | |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**SYLLABUS:**

**UNIT- I: Introduction to Data Structures:** Abstract Data Types (ADTs), The List ADT: Simple Array Implementation of Lists, Simple Linked Lists, Doubly Linked Lists, Circularly Linked Lists. The Stack ADT:The Stack Model, Implementation of Stacks, Applications of Stack. The Queue ADT: Queue
Model, Array Implementation of Queues, Application of Queues.Stacks and Queue implementation using linked list.

**UNIT-II: Searching**: List Searches, Linear and Binary Search Methods.
**Sorting:** Selection Sort, Insertion Sort, Quick Sort, Merge Sort, Heap sort.
**Hashing:** Hash Function, Separate Chaining, Collision Resolution-Separate Chaining, Open Addressing: Linear Probing, quadratic probing, Double Hashing, rehashing, Extendible Hashing.

**UNIT- III: Trees:** Binary Trees- Implementation, Expression Trees. Binary Search Trees- find, findMin and findMax, insert, delete operations.

**UNIT- IV: Trees:** AVL Trees- Single and Double Rotation, Operations. B-Tree: searching, insertion, deletion

**UNIT –V: Trees:** Introduction to Red-Black, splay trees and Comparison of Search Trees **Priority Queues:** Priority Queue Models, Simple Implementations, Binary **Heap:** Structure Property, Heap-Order Property, Basic Heap Operations. Applications: The Selection problem, Event Simulation.

**TEXT BOOKS:**

1. Data Structures and Algorithm Analysis, 4th Edition, Mark Allen Weiss,Pearson.
2. Data Structures: A PseudoCode Approach with C, 2nd Edition, Richard F.Gilberg, &Behrouz A. Forouzon, Cengage.

**REFERENCES BOOKS**:

1. Data Structures, Algorithms and Applications in java, 2/e, Sartaj Sahni,University Press.
2. Data Structures using C, 2/e, Reema Thareja

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**I B.Tech II sem-R20**

**OBJECT ORIENTED PROGRAMMING THROUGH C++**
**Code: R20ES1203**

**COURSE OBJECTIVES:** This course is designed to provide a comprehensive study of the C programming language.

- It stresses the strengths of C, which provide students with the means of writing efficient, maintainable and portable code.
- The nature of C language is emphasized in the wide variety of examples and applications.
- To know about some popular programming languages and how to choose Programming language for solving a problem.

**COURSE OUTCOMES**: At the end of the course, student will be able to

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Demonstrate basics of object oriented mode, differences between conventional and oops programming, the top-down and bottom-up approach I/O streams in C++ | **K2** |
| **CO2** | Explain Write, compile and debug programs and Use different data types, classes, objects and member functions in C++ language. | **K5** |
| **CO3** | Make use of Basic concept in C++ programming, Operators, control structures, functions, overloading, and recursion. | **K3** |
| **CO4** | Build dynamic memory management techniques using pointers, constructors, destructors, virtual functions. | **K3** |
| **CO5** | Classify inheritance with the understanding of early and late binding, usage of exception handling, generic programming. | **K2** |
| **CO6** | Apply advanced features of C++ specifically templates, operator overloading, standard template libraries (STL). | **K3** |

#***Based on suggested Revised BTL***

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 3 | | 2 | | | 2 | | | | | | |
| **CO2** | 1 | 2 | 3 | | | | | 1 | | | | |
| **CO3** | 2 | 3 | 2 | | | | | | 1 | | | |
| **CO4** | 1 | | 3 | | 2 | | | | | | | |
| **CO5** | 2 | 1 | 2 | | 3 | | | | | | | |
| **CO6** | 2 | 1 | 3 | | 2 | | | 3 | | | | |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**Department of Computer Science & Engineering**
**University College of Engineering, JNT University Kakinada**

**Syllabus:**
**UNIT – 1: Introduction to C++** : Difference between C and C++- Evolution of C++- The Object Oriented Technology- Disadvantage of Conventional Programming- Key Concepts of Object Oriented Programming- Advantage of OOP- Object Oriented Language.

**UNIT – 2: Classes and Objects &Constructors and Destructor**
Classes in C++-Declaring Objects- Access Specifiers and their Scope- Defining Member Function-Overloading Member Function- Nested class, Constructors and Destructors, Introduction- Constructors and Destructor- Characteristics of Constructor and Destructor-Application with Constructor- Constructor with Arguments (parameterized Constructor-Destructors- Anonymous Objects.

**UNIT – 3: Operator Overloading and Type Conversion & Inheritance**
The Keyword Operator- Overloading Unary Operator- Operator Return Type- Overloading Assignment Operator (=)- Rules for Overloading Operators, Inheritance, Reusability- Types of Inheritance- Virtual Base Classes- Object as a Class Member- Abstract Classes- Advantages of Inheritance-Disadvantages of Inheritance

**UNIT – 4: Pointers & Binding Polymorphisms and Virtual Functions**
Pointer, Features of Pointers- Pointer Declaration- Pointer to Class- Pointer Object- The this Pointer- Pointer to Derived Classes and Base Class, Binding Polymorphisms and Virtual Functions, Introduction- Binding in C++- Virtual Functions- Rules for Virtual Function- Virtual Destructor.
**UNIT – 5: Templates, Exception Handling**
Generic Programming with Templates, Need for Templates- Definition of class Templates- Normal Function Templates- Over Loading of Template Function-Bubble Sort Using Function Templates- Difference Between Templates and Macros- Linked Lists with Templates, Exception Handling- Principles of Exception Handling- The Keywords try throw and catch- Multiple Catch Statements – Specifying Exceptions.

**TEXT BOOKS:**
1. A First Book of C++, Gary Bronson, Cengage Learning.
2. The Complete Reference C++, Herbert Schildt, TMH.
3. Programming in C++, Ashok N Kamthane, Pearson 2nd Edition

**REFERENCE BOOKS:**
1. Object Oriented Programming C++, Joyce Farrell, Cengage.
2. C++ Programming: from problem analysis to program design, DS Malik, Cengage Learning

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**I B.Tech II sem-R20**

**DATA STRUCTURES LAB**
**Code: R20ES1204**

**COURSE OBJECTIVES:** From the course the student will
- Solve problems using data structures such as linear lists, stacks, queues, hash tables
- Be familiar with advanced data structures such as balanced search trees, AVL Trees, and B Trees.

**COURSE OUTCOMES**: After the completion of the course, student will be able

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Examine basic operations such as Insertion, Deletion, and Search on Advanced Data structures like Heaps, AVL trees, and B-Trees | **K4** |
| **CO2** | Summarize the practical applications of several advanced techniques like Hashing, Data compression techniques, and spanning trees in the domains of DBMS, Compiler design, and in Network routing. | **K2** |
| **CO3** | Identify the appropriate data structure for the given problem definition | **K3** |
| **CO4** | Analyze advanced concepts and data structures to improve the efficiency of real time systems | **K4** |

**#based on suggested Revised BTL**

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 1 | 3 | 3 | | | 2 | | 3 | | | | |
| CO2 | 2 | 2 | | 3 | | 3 | | 2 | | | | |
| CO3 | | | | | | 2 | | 2 | | | | |
| CO4 | | 2 | 2 | | | | | 3 | | | | |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**List of experiments:**
**Experiment 1:**
1. Write a C program to perform various operations on single linked list

**Experiment 2:**
2. Write a C program for the following
   a) Reverse a linked list
   b) Sort the data in a linked list

**Experiment 3:**
3. Write a C program for the following
   a) Remove duplicates
   b) Merge two linked lists

**Experiment 4:**
4. Write a C program to perform various operations on doubly linked list.

**Experiment 5:**
5. Write a C program to perform various operations on circular linked list.

**Experiment 6:**
6. Write a C program for performing various operations on stack using linked list.

**Experiment 7:**
7. Write a C program for performing various operations on queue using linked list.

**Experiment 8:**
8. Write a C program for the following using stack
   a) Infix to postfix conversion.
   b) Expression evaluation.
   c) Obtain the binary number for a given decimal number.

**Experiment 9:**
9. Write a C program to implement various operations on Binary Search Tree Using Recursive and Non-Recursive methods.

**Experiment 10:**
10. Write a C program to implement Selection Sort & Insertion Sort for given elements.

**Experiment 11:**
11. Write a C program to implement Merge & Heap Sort for given elements.

**Experiment 12:**
12. Write a C program to implement Quick Sort for given elements.

**Experiment 13:**
13. Write a C program to implement various operations on AVL trees.

**Experiment 14:**
14. Write a C program to perform the following operations:
   a) Insertion into a B-tree b) Searching in a B-tree

**Experiment 15:**
15. Write a C program to implementation of recursive and non-recursive functions to Binary tree Traversals

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**I B.Tech II sem-R20**

**OBJECT ORIENTED PROGRAMMING THROUGH C++ Lab**
**Code: R20ES1205**

**COURSE OBJECTIVES:**

- To strengthen their problem solving ability by applying the characteristics of an object- oriented approach.
- To introduce object oriented concepts in C++ and Java

**COURSE OUTCOMES:**

After the completion of the course, students will be able to

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Explain what constitutes an object-oriented approach to programming and identify potential benefits of object-oriented programming over other approaches. | **K5** |
| **CO2** | Analyze simple programs using classes and objects in C++. | **K4** |
| **CO3** | Experiment with programs involving constructors, destructors and reuse of code using inheritance. | **K3** |
| **CO4** | Examine Object Oriented Programs using templates and exceptional handling concepts. | **K4** |
| **CO5** | Apply an object-oriented approach to developing applications of varying complexity. | **K3** |

***#Based on suggested Revised BTL***

**Mapping of course outcomes with program outcomes**

| CO | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 3 | | | | | 2 | | | | | | |
| **CO2** | | 2 | | | | | | 2 | | | | |
| **CO3** | | 3 | 2 | 2 | | | | | 2 | | | |
| **CO4** | | 2 | 3 | | 2 | 2 | | | | | | |
| **CO5** | | | 2 | | 3 | | | | | | | |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**Programming Exercises:**

**Exercise 1: (Basics)**

Write a Simple Program on printing “Hello World” and “Hello Name” where name is the input from the user

a) Convert any two programs that are written in C into C++

b) Write a description of using g++ (150 Words)

**Exercise 2: (Expressions Control Flow)**

a) Write a Program that computes the simple interest and compound interest payable on principal amount (inRs.) of loan borrowed by the customer from a bank for a given period of time (in years) at specific rate of interest. Further determine whether the bank will benefit by charging simple interest or compound interest.
b) Write a Program to calculate the fare for the passenger starveling in a bus. When a Passenger enters the bus, the conductor asks "What distance will you travel?" On knowing distance from passenger (as an approximate integer), the conductor mentions the fare to the passenger according to following criteria.

**Exercise 3: (Variables, Scope, Allocation)**
a) Write a program to implement call by value and call by reference using reference variable.
b) Write a program to illustrate scope resolution, new and delete Operators. (Dynamic Memory Allocation)
c) Write a program to illustrate Storage classes
d) Write a program to illustrate Enumerations

**Exercise 4: (Functions)**
Write a program illustrating Inline Functions
a) Write a program illustrates function overloading. Write 2 overloading functions for power.
b) Write a program illustrates the use of default arguments for simple interest function.

**Exercise 5: (Functions –Exercise Continued)**
a) Write a program to illustrate function overloading. Write 2 overloading functions for adding two numbers
b) Write a program illustrate function template for power of a number.
c) Write a program to illustrate function template for swapping of two numbers

**Exercise 6: (Classes Objects)**
Create a Distance class with:
- feet and inches as data members
- member function to input distance
- member function to output distance
- member function to add two distance objects

a) Write a main function to create objects of DISTANCE class. Input two distances and output the sum.
b) Write a C++ Program to illustrate the use of Constructors and Destructors (use the above program.)
c) Write a program for illustrating function overloading in adding the distance between objects (use the above problem)
d) Write a C++ program demonstrating a Bank Account with necessary methods and variables

**Exercise 7: (Access)**
Write a program for illustrating Access Specifiers public, private, protected
a) Write a program implementing Friend Function
b) Write a program to illustrate this pointer
c) Write a Program to illustrate pointer to a class

**Exercise 8: (Operator Overloading)**
a) Write a program to Overload Unary, and Binary Operators as Member Function, and Non Member Function.
i. Unary operator as member function
ii. Binary operator as nonmember function
b) Write a c ++ program to implement the overloading assignment = operator
c) Write a case study on Overloading Operators and Overloading Functions (150 Words)

**Exercise 9: (Inheritance)**
a) Write C++ Programs and incorporating various forms of Inheritance
i) Single Inheritance ii) Hierarchical Inheritance
iii) Multiple Inheritances

**Exercise 10: (Inheritance –Continued)**
a) Write C++ Programs and incorporating various forms of Inheritance
i) Multi-level inheritance ii) Hybrid inheritance
b) Write a program to show Virtual Base Class
c) Write a case study on using virtual classes (150 Words)

**Exercise 11: (Inheritance –Continued)**
a) Write a Program in C++ to illustrate the order of execution of constructors and destructors in inheritance
b) Write a Program to *show* how *constructors* are invoked in *derived class*

**Exercise 12: (Polymorphism)**
a) Write a program to illustrate runtime polymorphism
b) Write a program to illustrate this pointer
c) Write a program illustrates pure virtual function and calculate the area of different shapes by using abstract class.
d) Write a case study on virtual functions (150 Words)

**Exercise 13: (Templates)**
a) Write a C++ Program to illustrate template class
b) Write a Program to illustrate class templates with multiple parameters
c) Write a Program to illustrate member function templates

**Exercise 14: (Exception Handling)**
a) Write a Program for Exception Handling Divide by zero
b) Write a Program to rethrow an Exception

**Exercise 15: (STL)**
a) Write a Program to implement List and List Operations
b) Write a Program to implement Vector and Vector Operations

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

## I B.Tech II sem-R20

**ENGINEERING EXPLORATION PROJECT – DESIGN THINKING**
**Code: R20PR1201**
*(COMMON FOR CE, EEE, ME, ECE, & CSE)* (15 HRS PER SEM.)

**COURSE OBJECTIVES:**

- Build mindsets & foundations essential for designers
- Learn about the Human-Centered Design methodology and understand their real-world applications
- Use Design Thinking for problem solving methodology for investigating ill-defined problems.
- Undergo several design challenges and work towards the final design challenge

Apply Design Thinking on the following Streams to
- Project Stream 1: Electronics, Robotics, IOT and Sensors
- Project Stream 2: Computer Science and IT Applications
- Project Stream 3: Mechanical and Electrical tools
- Project Stream4: Eco-friendly solutions for waste management, infrastructure, safety, alternative energy sources, Agriculture, Environmental science and other fields of engineering.

**HOW TO PURSUE THE PROJECT WORK?**
- The first part will be learning-based-masking students to embrace the methodology by exploring all the phases of design thinking through the wallet/ bag challenge and podcasts.
- The second part will be more discussion-based and will focus on building some necessary skills as designers and learning about complementary material for human- centered design.
- The class will then divide into teams and they will be working with one another for about 2 – 3 weeks. These teams and design challenges will be the basis for the final project and final presentation to be presented.
- The teams start with **Design Challenge** and go through all the phases more in depth from coming up with the right question to empathizing to ideating to prototyping and to testing.
- Outside of class, students will also be gathering the requirements, identifying the challenges, usability, importance etc
- At the end, Students are required to submit the final reports, and will be evaluated by the faculty.

**TASKS TO BE DONE:**

Task 1: Everyone is a Designer
- Understand class objectives & harness the designer mindset

Task 2: The Wallet/Bag Challenge and Podcast
- Gain a quick introduction to the design thinking methodology
- Go through all stages of the methodology through a simple design challenge
- Podcast: Observe, Listen and Engage with the surrounding environment and identify a design challenge.

Task 3: Teams & Problems (Brain Storming)
- Start Design Challenge and learn about teams & problems through this
- Foster team collaboration, find inspiration from the environment and learn how to identify problems

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

Task 4: Empathizing (Brain Storming)
- Continue Design Challenge and learn empathy
- Learn techniques on how to empathize with users
- Go to the field and interview people in their environments
- Submit Activity Card

Task 5: Ideating (Business Planning)
- Continue Design Challenge and learn how to brainstorm effectively
- Encourage exploration and foster spaces for brainstorming
- Submit Activity Card

Task 6: Prototyping (Business Model generation)
- Continue Design Challenge and learn how to create effective prototypes
- Build tangible models and use them as communication tools
- Start giving constructive feedback to classmates and teammates
- Submit Activity Card

Task 7: Testing
- Finish Design Challenge and iterate prototypes and ideas through user feedback
- Evolve ideas and prototypes through user feedback and constructive criticism
- Get peer feedback on individual and group performance
- Submit Activity Card

Task 8: Pitching
- Prepare poster to communicate your idea, how it works, why it counts, and who it benefits.
- Final Report Submission and Presentation

Task 9: Case Study
- Make students groups. Each group will generate a business model for unsolved problems in their organization and present.

**Note:** The colleges may arrange for Guest Speakers from Various Design Fields: Graphic Design, Industrial Design, Architecture, Product Design, Organizational Design, etc to enrich the students with Design Thinking Concept.

**REFERENCES:**

1. Tom Kelly, *The Art of Innovation: Lessons in Creativity From IDEO, America's Leading Design Firm* (Profile Books, 2002)
2. Tim Brown, Change by Design: How Design Thinking Transforms Organizations and Inspires Innovation (HarperBusiness, 2009)
3. Jeanne Liedtka, Randy Salzman, and Daisy Azer, Design Thinking for the Greater Good: Innovation in the Social Sector (Columbia Business School Publishing, 2017)

**OTHER USEFUL DESIGN THINKING FRAMEWORKS AND METHODOLOGIES:**

- Human-Centered Design Toolkit (IDEO); https://www.ideo.com/post/design-kit
- Design Thinking Boot Camp Bootleg (Stanford D-School); https://dschool.stanford.edu/resources/the-bootcamp-bootleg
- Collective Action Toolkit (frogdesign); https://www.frogdesign.com/wpcontent/
    - uploads/2016/03/CAT_2.0_English.pdf
- Design Thinking for Educators (IDEO); https://designthinkingforeducators.com/

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**II B. Tech – I Sem-R20**

**MATHEMATICAL FOUNDATIONS OF COMPUTER SCIENCE**

**Code: R20PC2101**

**COURSE OBJECTIVES:**

- To introduce the students to the topics and techniques of discrete methods and combinatorial reasoning.
- To introduce a wide variety of applications. The algorithmic approach to the solution of problems is fundamental in discrete mathematics, and this approach reinforces the close ties between this discipline and the area of computer science.

**COURSE OUTCOMES (COS):**

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| CO1 | **Build** skills in solving mathematical problems | K3 |
| CO2 | **comprehend** mathematical principles and logic | K4 |
| CO3 | **demonstrate** knowledge of mathematical modeling and proficien in using mathematical software | K6 |
| CO4 | **Manipulate and analyze** data numerically and/or graphically using appropriate Software | K3 |
| CO5 | **How** to communicate effectively mathematical ideas/results verbally or in writing | K1 |

**#Based on suggested Revised BTL**

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 1 | | | 2 | | 2 | | 3 | | | | |
| CO2 | 2 | | | 2 | | 3 | | 2 | | | | |
| CO3 | | | | | | 2 | | 2 | | | | |
| CO4 | | | | 3 | | | | 3 | | | | |
| CO5 | | | | | | | | | | | | |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**Syllabus:**

**UNIT-I: Mathematical Logic**: *Propositional Calculus*: Statements and Notations, Connectives, Well Formed Formulas, Truth Tables, Tautologies, Equivalence of Formulas, Duality Law, Tautological Implications, Normal Forms, Theory of Inference for Statement Calculus, Consistency of Premises, Indirect Method of Proof, *Predicate Calculus*: Predicates, Predicative Logic, Statement Functions, Variables and Quantifiers, Free and Bound Variables, Inference Theory for Predicate Calculus.

**UNIT-II: Set Theory:** *Sets*: Operations on Sets, Principle of Inclusion-Exclusion, *Relations:* Properties, Operations, Partition and Covering, Transitive Closure, Equivalence, Compatibility and Partial Ordering, Hasse
Diagrams, *Functions:* Bijective, Composition, Inverse, Permutation, and Recursive Functions, Lattice and its Properties, *Algebraic Structures:* Algebraic Systems, Properties, Semi Groups and Monoids, Group, Subgroup and Abelian group, Homomorphism, Isomorphism,

**UNIT-III: Combinatorics:** Basis of Counting, Permutations, Permutations with Repetitions, Circular and Restricted Permutations, Combinations, Restricted Combinations, Binomial and Multinomial Coefficients and Theorems, *Number Theory:* Properties of Integers, Division Theorem, Greatest Common Divisor, Euclidean Algorithm, Least Common Multiple, Testing for Prime Numbers, The Fundamental Theorem of Arithmetic, Modular Arithmetic, Fermat's and Euler's Theorems

**UNIT-IV: Recurrence Relations:** Generating Functions, Function of Sequences, Partial Fractions, Calculating Coefficient of Generating Functions, Recurrence Relations, Formulation as Recurrence Relations, Solving Recurrence Relations by Substitution and Generating Functions, Method of Characteristic Roots, Solving Inhomogeneous Recurrence Relations

**UNIT-V: Graph Theory:** Basic Concepts, Graph Theory and its Applications, Sub graphs, Graph Representations: Adjacency and Incidence Matrices, Isomorphic Graphs, Paths and Circuits, Eulerian and Hamiltonian Graphs, Multigraphs, Bipartite and Planar Graphs, Euler's Theorem, Graph Colouring and Covering, Chromatic Number, Spanning Trees, Prim's and Kruskal's Algorithms, BFS and DFS Spanning Trees

**TEXT BOOKS:**

1. Discrete Mathematical Structures with Applications to Computer Science, J. *P. Tremblay and P. Manohar*, Tata McGraw Hill.
2. Elements of Discrete Mathematics-A Computer Oriented Approach, *C. L. Liu and D. P. Mohapatra*, 3rd Edition, Tata McGraw Hill.
3. Theory and Problems of Discrete Mathematics, Schaum's Outline Series, *Seymour Lipschutz and Marc Lars Lipson*, 3rd Edition, McGraw Hill.

**REFERENCE BOOKS:**

1. Discrete Mathematics for Computer Scientists and Mathematicians, *J. L. Mott, A. Kandel and T. P. Baker*, 2nd Edition, Prentice Hall of India.
2. Discrete Mathematical Structures, *Bernand Kolman, Robert C. Busby and Sharon Cutler Ross*, PHI.
3. Discrete Mathematics, *S. K. Chakraborthy and B.K. Sarkar*, Oxford, 2011.
4. Discrete Mathematics and its Applications with Combinatorics and Graph Theory, *K. H. Rosen*, 7th Edition, Tata McGraw Hill.

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**II B. Tech – I Sem-R20**

**OPERATING SYSTEMS**
**R20PC2102**

**COURSE OBJECTIVES:**

- Study the basic concepts and functions of operating systems.
- Understand the structure and functions of OS.
- Learn about Processes, Threads and Scheduling algorithms.
- Understand the principles of concurrency and Deadlocks.
- Learn various memory management schemes.
- Study I/O management and File systems.
- Learn the basics of Linux system and perform administrative tasks on Linux Servers.

**COURSE OUTCOMES:**

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Describe Computer Operating System Functions, Structures and System Calls. | **K1** |
| **CO2** | Demonstrate various Process Management Concepts and CPU Scheduling Algorithms and Process Synchronization Techniques. | **K3** |
| **CO3** | Illustrate Memory Management Techniques and Page Replacement Algorithms. | **K2** |
| **CO4** | Apply Deadlock Prevention and Avoidance Techniques | **K3** |
| **CO5** | Demonstrate File System Concepts and Mass Storage Structures | **K3** |

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 1 | | | 2 | | 2 | | 3 | | | | |
| CO2 | 2 | | | 2 | | 3 | | 2 | | | | |
| CO3 | | | | | | 2 | | 2 | | | | |
| CO4 | | | | 3 | | | | 3 | | | | |
| CO5 | | | | | | | | | | | | |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**SYLLABUS:**

**UNIT-I: Introduction to Operating System Concept:** Types of operating systems, operating systems concepts, Evaluation of operating systems, operating systems services, functions of OS, structure of OS, Introduction to System call, System call types.

**UNIT-II: Process Management** – Process concept, The process, Process State Diagram, Process control block, Process Scheduling- Scheduling Queues, Schedulers, Operations on Processes, Interprocess Communication, Threading Issues, Scheduling-Basic Concepts, Scheduling Criteria, Scheduling Algorithms.

**UNIT-III: Memory Management:** Swapping, Contiguous Memory Allocation, Paging, structure of the Page Table, Segmentation **Virtual Memory Management:** Virtual Memory, Demand Paging, Page-Replacement Algorithms, Thrashing

**UNIT-IV: Concurrency:** Process Synchronization, The Critical- Section Problem, Synchronization Hardware, Semaphores, Classic Problems of Synchronization, Monitors, Synchronization examples **Principles of deadlock**: System Model, Deadlock Characterization, Deadlock Prevention, Detection and Avoidance, Recovery form Deadlock

**UNIT-V: File System Interface:** Concept of a file, Access Methods, Directory structure, File system mounting, file sharing, protection. **File System Implementation:** File system structure, allocation methods, Disk scheduling, **Linux System**: Components of LINUX, Inter-process Communication, Synchronization, Interrupt, Exception and System Call. **Case studies: Android, UNIX, Windows**

**TEXT BOOK:**

1. Operating System Concepts, Abraham Silberschatz, Peter Baer Galvin and Greg Gagne 9th Edition, John Wiley and Sons Inc., 2012.
2. Operating Systems – Internals and Design Principles, William Stallings, 7th Edition, Prentice Hall, 2011.
3. Operating Systems-S Halder, Alex A Aravind Pearson Education Second Edition 2016.

**REFERENCES:**

1. Modern Operating Systems, Andrew S. Tanenbaum, Second Edition, Addison Wesley, 2001.
2. Operating Systems: A Design-Oriented Approach, Charles Crowley, Tata McGraw Hill Education, 1996.
3. Operating Systems: A Concept-Based Approach, D M Dhamdhere, Second Edition, Tata McGraw-Hill Education, 2007.

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**II B. Tech I Sem-R20**

**ADVANCED JAVA PROGRAMMING**
**Code: R20PC2103**

**COURSE OUTCOMES:**

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| CO1 | Explain Basics concept of Object Oriented Programming | K3 |
| CO2 | Discuss the Inheritance, Packages, Interfaces: | K5 |
| CO3 | Explain How to handle Exception handling | K2 |
| CO4 | How to create application Applets, Swings | K3 |
| CO5 | Apply and develop the Database Access using java Beans, EJB | K5,K4 |

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 1 | | | 2 | | 2 | | 3 | | | | |
| CO2 | 2 | | | 2 | | 3 | | 2 | | | | |
| CO3 | | | | | | 2 | | 2 | | | | |
| CO4 | | | | 3 | | | | 3 | | | | |
| CO5 | | | | | | | | | | | | |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**SYLLABUS:**

**UNIT-I Basics of Object Oriented Programming (OOP):** The history and evolution of java, java Buzzwords, The OOP principles-Encapsulation, Inheritance, Polymorphism, Data types, variables, scope and life time of variables, arrays, operators, expressions, control statements, type conversion and casting, sample java program and compilation, classes and objects, concepts of classes, objects, constructors, access control, this keyword, garbage collection, overloading methods and overloading constructors, parameter passing, recursion, Exploring the string handling functions

**UNIT-II: Inheritance:** Hierarchical abstractions, Base class object, subclass, subtype, substitutability, forms of inheritance- specialization, specification, construction, extension, Method binding, method overriding, Member access rules, super uses, using final keyword, polymorphism, abstract classes.
**Packages**: Defining a Package, CLASSPATH, Access protection, importing packages. java.io package, java.util package.
**Interfaces:** Defining an interface, implementing interfaces, Nested interfaces, applying interfaces, variables in interfaces and extending interfaces.

**UNIT-III: Exception handling:** Concepts of exception handling, benefits of exception handling, exception hierarchy, usage of try, catch, throws and finally, built in exceptions, creating own exception sub classes.
**Multithreading:** Differences between multithreading and multitasking, thread life cycle, creating threads, synchronizing threads, Daemon threads, Thread Group.

**UNIT-IV: Applets:** Concept of Applets, differences between applets and applications, lifecycle of an applet, local and remote applets, creating applets, passing parameters to applets.
**Swings:** Introduction, MVC architecture, components, containers, exploring swing-JApplet, JFrame and JComponent, Icons and Labels, JTextField, JButton, JCheckBox, JRadioButton, JComboBox, JTabbedPane, JScrollPane, JTree and JTable.

**UNIT-V: Database Access:** Database Programming using JDBC, Studying Javax.sql.*package.
**Java Beans:** Introduction, Advantages of Java Beans, BDK Introspection.
**Enterprise Java Beans:** Introduction, EJB Architecture and Design: EJB Container and its services, Working with EJBs, Design of the EJB Tier

**TEXT BOOKS:**
1. Java: The Complete Reference, Eleventh Edition, Herbert Schildt.
2. Internet and World Wide Web: How to program,6/e, Dietel, Dietel, Pearson.
3. Java Server Faces, Hans Bergstan, O'reilly.

**REFERENCE BOOKS:**
1. Core Java 2, Vol 1(Vol 2) Fundamentals(Advanced), 7/e, Cay.S.Horstmann, Gary Cornell, Pearson.
2. Professional Java Server Programming, J2EE 1.3 edition, APRESS publications, 2007
3. Web Programming, building internet applications, 2/e, Chris Bates, Wiley Dreamtech
4. Programming World Wide Web, Sebesta, PEA

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**II B. Tech I Sem-R20**

**DATA BASE MANAGEMENT SYSTEMS**
**Code: R20PC2104**

**Pre-requisite: Basic Knowledge on Data Structures**

**COURSE OUTCOMES:** At the end of the course, student will be able to

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | **Illustrate** the concept of databases, database management systems, database languages, database structures and their work | **K2** |
| **CO2** | **Apply** ER modeling and Relational modeling for designing simple databases. | **K3** |
| **CO3** | **Summarize** the concepts related to relational model and SQL and **Write** database queries using relational algebra and structured query language. | **K2** |
| **CO4** | **Design and develop** databases from the real world by applying the concepts of Normalization. | **K6** |
| **CO5** | **Outline** the issues associated with Transaction Management and Recovery, Tree Structured Indexing | **K2** |

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 1 | | | 2 | | 2 | | 3 | | | | |
| **CO2** | 2 | | | 2 | | 3 | | 2 | | | | |
| **CO3** | | | | | | 2 | | 2 | | | | |
| **CO4** | | | | 3 | | | | 3 | | | | |
| **CO5** | | | | | | | | | | | | |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**SYLLABUS:**

**UNIT – 1: Overview of Database System**: Database System Applications, Purpose of Database Systems, View of Data, Database Languages, Relational Databases, Database Design, Data Storage and Querying, Transaction Management, Database Architecture, Data Mining and Informational Retrieval, Specialty Databases, Database Users and Administrators, History of Database Systems. **[Text Book -2]**
**Introduction to Database Design:** Database Design and ER Diagrams, Entities, Attributes and Entity Sets, Relationships and Relationship Sets, Additional Features of the ER Model, Conceptual Design with the ER Model, Extended ER features **[Text Book -1]**

**UNIT – 2: Relational Model:** Introduction to the Relational Model, Integrity Constraints over Relations, Enforcing Integrity Constraints, Querying Relational Data, Logical Database Design: ER to Relational, Introduction to Views, Destroying/Altering Tables and Views **[Text Book -1]**

**Relational Algebra:** Selection and Projection, Set Operations, Renaming, Joins, Division, More Examples of Algebra Queries **[Text Book -1]**

**UNIT – 3: SQL: Queries, Constraints, Triggers:** The Form of a Basic SQL Query, UNION, INTERSECT and EXCEPT, Nested Queries, Aggregate Operators, Null Values, Complex Integrity Constraints in SQL, Triggers, Exceptions, Procedures, Functions **[Text Book -1]Normal Forms:** Introduction to Schema Refinement, Functional Dependencies, Reasoning about FDs, Normal Forms, Properties of Decompositions, Normalization. **[Text Book -1]**

**UNIT – 4: Schema Refinement:** Multi-valued dependencies, Fourth Normal form, Join Dependencies, Fifth Normal Form, Loss less join, dependency preservation. **[Text Book -1] Transaction Management:** Transaction Concepts, Transaction state, Implementation of Atomicity and Durability, Concurrent Execution, Serializability, Recoverability. **[Text Book -2] Concurrency Control:** Lock-based Protocols: Locks, Granting of Locks, Two Phase Locking Protocol, Implementation of locking; Timestamp-Based Protocols: Time Stamps, Time Stamp Ordering protocol, Thomas Write Rule, Validation-Based Protocols **[Text Book -2]**

**UNIT – 5: Overview of Storage and Indexing:** Data on External Storage, File organization and indexing: Clustered Indexes, Primary and Secondary Indexrs; Index Data Structures: Hash and Tree based indexing; Comparison of File organizations. **[Text Book -1] Tree Structured Indexing:** Intuitions for Tree Indexes, Indexed Sequential Access Method (ISAM), B+ Trees: A Dynamic Index Structure, Search, Insert, Delete, Duplicates, B+ Trees in Practice **[Text Book -1]**

**TEXT BOOKS**:

1. Data base Management Systems, 3/e, Raghurama Krishnan, Johannes Gehrke, Mc Graw-Hill
2. Data base System Concepts, 6/e, Abraham Silberschatz, Henry F. Korth, S. Sudarshan, Mc Graw-Hill
3. Database Systems, 9/e, Carlos Coronel, Steven Morris, Peter Rob, Cengage

**REFERENCE BOOKS**:

1. Database Systems, 6/e Ramez Elmasri, Shamkant B. Navathe, Pearson
2. Introduction to Database Systems, 8/e, C J Date, Pearson

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**II B. Tech I Sem-R20**

### DATABASE MANAGEMENT SYSTEMS LAB
### Code: R20PC2105

**Pre-requisite: Knowledge on Database, Basic knowledge on SQL**

**COURSE OUTCOMES**: At the end of the course, student will be able to

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | **Utilize** SQL to execute queries for creating database and performing data manipulation operations | K3 |
| **CO2** | **Examine** integrity constraints to build efficient databases | K4 |
| **CO3** | **Apply** Queries using Advanced Concepts of SQL | K3 |
| **CO4** | **Build** PL/SQL programs including stored procedures, functions, cursors and triggers. | K6 |

**List of Experiments:**

**Experiment- 1:** To implement Data Definition language
1. Create, Alter, Drop, Truncate
2. To implement Constraints.
   (a) Primary key, (b) Foreign Key, (c) Check, (d) Unique, (e) Null, (f) Not null, (g) Default, (h) Enable Constraints, (i) Disable Constraints (j). Drop Constraints

**Experiment- 2:** To implementation on DML, TCL and DRL
(a) Insert (b) Select (c) Update d) Delete (e) Commit (f) Rollback (g) Save point (i) Like'%' (j) Relational Operator.

**Experiment- 3:**
Implement In-Built Functions Numeric , Date , String, Group and Time functions

**Experiment- 4:** To implement Nested Queries & Join Queries
4.1. To implementation of Nested Queries
4.2. (a) Conditional Join (b) Equi Join (c) Inner join, (d).Left outer join (e).Right outer join (f).Full outer join

**Experiment- 5:** To implement Views
(a) View (b) Force View (c) View with Check, Update Option

**Experiment- 6:**
To implement Queries on Group By & Having Clauses, ALIAS, Sequence By, Order By

**Experiment- 7: (**Control Structure)

7.1 To write a PL/SQL block for Addition of Two Numbers
7.2 To write a PL/SQL block for IF Condition
7.3 To write a PL/SQL block for IF and else condition
7.4 To write a PL/SQL block for greatest of three numbers using IF AND ELSEIF
**7.5** To write a PL/SQL block for summation of odd numbers using FOR LOOP

**Experiment- 8: (Exception Handling)**

8.1 Raising Exceptions
8.2 User Defined Exceptions Pre-Defined Exceptions

**Experiment- 9: (Procedures)**

9.1 To write a PL/SQL Procedure using Positional Parameters
9.2 To write a PL/SQL Procedure using notational parameters
9.3 To write a PL/SQL Procedure for GCD Numbers
9.4 To write a PL/SQL Procedure for cursor implementation
9.5 To write a PL/SQL Procedure for explicit cursors implementation
9.6 To write a PL/SQL Procedure for implicit cursors implementation

**Experiment- 10: (Functions)**

1. To write a PL/SQL block to find factorial of a given number using function
2. To write a PL/SQL function to search an address from the given address database

**Experiment- 11: (Trigger)**

11.1 To write a Trigger to pop-up the DML operations
11.2 To write a Trigger to check the age valid or not Using Message Alert.
11.3 Create a Trigger to raise appropriate error code and error message.
11.4 Create a Trigger for a table it will update another table while inserting values

**TEXT BOOKS**:

1. Oracle: The Complete Reference by Oracle Press
2. Nilesh Shah, "Database Systems Using Oracle", PHI, 2007
3. Rick F Vander Lans, "Introduction to SQL", Fourth Edition, Pearson Education, 2007

**WEB REFERENCES**:

1. https://www.geeksforgeeks.org/dbms/
2. https://www.w3schools.com/sql/
3. https://www.tutorialspoint.com/dbms/index.htm

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**II B. Tech I Sem-R20**

### OPERATING SYSTEMS & LINUX LAB
### Code: R20PC2106

**COURSE OBJECTIVES:**

- To write programs in Linux environment using system calls.
- To implement the scheduling algorithms.
- To implement page replacement algorithms
- To implement file allocation methods.
- To understand and implement IPC mechanism using named and unnamed pipes.
- To develop solutions for synchronization problems using semaphores.

**COURSE OUTCOMES:** At the end of the course, student will be able to

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| CO1 | Ability to develop application programs using system calls in UNIX. | K3 |
| CO2 | Ability to implement inter process communication between two processes. | K4 |
| CO3 | Ability to design and solve synchronization problems | K3 |
| CO4 | Ability to simulate and implement operating system concepts such as scheduling, deadlock management, file management, and memory management. | K6 |

**Use Linux operating system and GNU C compiler.**

**List of Experiments:**

1. Write C programs to simulate the following CPU scheduling algorithms:
   a) Round Robin
   b) SJF
2. Write C programs to simulate the following CPU scheduling algorithms:
   a) FCFS
   b) Priority
3. Write C programs to simulate the following File organization techniques:
   a) Single level directory
   b) Two level
   c) Hierarchical

4. Write C programs to simulate the following File allocation methods:
   a) Contiguous
   b) Linked
   c) Indexed

5. Write a C program to copy the contents of one file to another using system calls.

6. Write a C program to simulate Bankers Algorithm for Dead Lock Avoidance
7. Write a C program to simulate Bankers Algorithm for Dead Lock Prevention
8. Write C programs to simulate the following page replacement algorithms:
   a) FIFO
   b) LRU
   c) LFU
9. Write C programs to simulate the following techniques of memory management:
   a) Paging
   b) Segmentation
10. Write a C program to implement the ls | sort command. (Use unnamed Pipe)
11. Write a C program to solve the Dining- Philosopher problem using semaphores.
12. Write C programs to implement ipc between two unrelated processes using named pipe.

**<u>Linux Lab</u>**

1. Write a Shell program to check whether given number is prime or not.
2. Write a shell script which will display Fibonacci series up to the given range.
3. Write a shell script to check whether the given number is Armstrong or not.
4. Write a shell script to calculate the value of $n_c^r$
5. Write a shell script to accept student number, name, marks in 5 subjects. Find total, average and grade using the following rules:
   Avg>=80 then grade A
   Avg<80 && Avg>=70 then grade B
   Avg<70 && Avg>=60 then grade C
   Avg<60 && Avg>=50 then grdae D
   Avg<50 && Avg>=40 then grade E else grade F
6. Write a shell script to find minimum and maximum elements in the given list ofelements.
7. Write a shell program to check whether the given string is palindrome or not.
8. Write an awk program to print sum, avg of students marks list

**Department of Computer Science & Engineering**
**University College of Engineering, JNT University Kakinada**

**II B. Tech – I Sem-R20**

**ADVANCED JAVA PROGRAMMING LAB**
**Code: R20PC2107**

**COURSE OUTCOMES:** At the end of the course, student will be able to

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Create Java program to implement the java based Programs | **K3** |
| **CO2** | Apply Java Program to check Overloading methods | **K5** |
| **CO3** | Implement a program to execute selected query using JDBC | **K4** |
| **CO4** | Develop Create a simple visual bean with an area filled with a color | **K5** |

**List of Experiments:**

1. The Fibonacci sequence is defined by the following rule. The first 2 values in the sequence are 1, 1. Every subsequent value is the sum of the 2 values preceding it. Write a Java Program that uses both recursive and non-recursive functions to print the nth value of the Fibonacci sequence.
2. Write a Java Program that prompts the user for an integer and then prints out all the prime numbers up to that integer.
3. Write a java program to implement call by value and call by reference mechanisms.
4. Write a Java Program, using String Tokenizer class, which reads a line of integers and then displays each integer and the sum of all integers.
5. Write a Java Program that checks whether a given string is a palindrome or not.
6. Write a Java Program to check the compatibility for multiplication, if compatible multiply two matrices and find its transpose.
7. Write a Java program to implement constructor overloading and method overloading.
8. Write a Java Program that illustrates how runtime polymorphism is achieved.
9. Write a Java Program that illustrates the use of super keyword.
10. Write a Java Program to create and demonstrate packages.
11. Write a Java Program that displays the number of characters, lines and words in a text/text file.

**Department of Computer Science & Engineering**
**University College of Engineering, JNT University Kakinada**

12. Write a Java Program to implement a Queue, using user defined Exception Handling (also make use of throw, throws).

13. Write a Java Program that creates 3 threads by extending Thread class. First thread displays "Good Morning" every 1 sec, the second thread displays "Hello" every 2 seconds and the third displays "Welcome" every 3 seconds. (Repeat the same by implementing Runnable).

14. Write a Java Program for handling mouse events.

15. Write a Java Program for handing keyboard events.

16. Write a Java Program that lets users create Pie charts. Design your own user interface (with Swings).

17. Implement a program to execute selected query using JDBC.

18. Write a program to update the customer information using JDBC.

19. Write a Program to implement the SQL commands using JDBC

20. Create a simple visual bean with an area filled with a color. The shape of the area depends on the property shape. If it is set to true then the shape of the area is Square and it is Circle, if it is false. The color of the area should be changed dynamically for every mouse click. The color should also be changed if we change the color in the "property window".

**Department of Computer Science & Engineering**
**University College of Engineering, JNT University Kakinada**

**I B. Tech I Sem-R20**

**MOBILE APP DEVELOPMENT LAB**
**Code: R20SO2101**

**COURSE OUTCOMES:** By the end of the course,

- Student will be able to write simple GUI applications, use built-in widgets and components, work with the database to store data locally, and much more.
- Android Application Development course is designed to quickly get you up to speed with writing apps for Android devices. The student will learn the basics of Android platform and get to understand the application lifecycle

**Programming Experiments:**

**Exercise 1:** Write a J2ME program to show how to change the font size and colour.

**Exercise 2:** Write a J2ME program which creates the following kind of menu.

- cut
- copy
- past
- delete
- select all
- unselect all

**Exercise 3:** Create a J2ME menu which has the following options (Event Handling):

- cut - can be on/off
- copy - can be on/off
- paste - can be on/off
- delete - can be on/off
- select all - put all 4 options on
- unselect all - put all

**Exercise 4:** Create a MIDP application, which draws a bar graph to the display. Data values can be given at int[] array. You can enter four data (integer) values to the input text field.

**Exercise 5:** Create an MIDP application which examines, that a phone number, which a user has entered is in the given format (Input checking):

1. Area code should be one of the following: 040, 041, 050, 0400, 044
2. There should 6-8 numbers in telephone number (+ area code)

**Exercise 6:** Write a sample program to show how to make a SOCKET Connection from J2ME phone. This J2ME sample program shows how to how to make a SOCKET Connection from a J2ME Phone. Many a times there is a need to connect backend HTTP server from the J2ME application. Show how to make a SOCKET connection from the phone to port 80.

**Exercise 7:** Login to HTTP Server from a J2ME Program. This J2ME sample program shows how to display a simple LOGIN SCREEN on the J2ME phone and how to authenticate to a HTTP server. Many J2ME applications for security reasons require the authentication of the user. This free J2ME sample program, shows how a J2ME application can do authentication to the backend server. Note: Use Apache Tomcat Server as Web Server and MySQL as Database Server.

**Exercise 8:** The following should be carried out with respect to the given set of application domains: (Assume that the Server is connected to the well-maintained database of the given domain. Mobile Client is to be connected to the Server and fetch the required data value/information)

1. Students Marks Enquiry
2. Town/City Movie Enquiry
3. Railway/Road/Air (For example PNR) Enquiry/Status
4. Sports (say, Cricket) Update
5. Town/City Weather Update
6. Public Exams (say Intermediate or SSC)/ Entrance (Say EAMCET) Results Enquiry
7. Divide Student into Batches and suggest them to design database according to their domains and render information according the requests.

**Exercise 9:** Write an Android application program that displays Hello World using Terminal.

**Exercise 10:** Write an Android application program that displays Hello World using Eclipse.

**Exercise 11:** Write an Android application program that accepts a name from the user and displays the hello name to the user in response as output using Eclipse.

**Exercise 12:** Write an Android application program that demonstrates the following:

1. LinearLayout
2. RelativeLayout
3. TableLayout
4. GridView layout

**Exercise 13:** Write an Android application program that converts the temperature in Celsius to Fahrenheit.

**Exercise 14:** Write an Android application program that demonstrates intent in mobile application development.

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**II B. Tech II Sem- R20**

**FORMAL LANGUAGES & AUTOMATA THEORY**
**Code: R20ES2201**

**COURSE OBJECTIVES:**
- Introduce student to the concepts of Theory of Computation in Computer Science
- The students should acquire insights into the relationship among formal languages, formal Grammars and automata.

**COURSE OUTCOMES:** By the end of the course students can

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Classify machines by their power to recognize languages. | **K4** |
| **CO2** | Attains the knowledge of language classes & grammars relationship among them with the help of Chomsky hierarchy | **K2** |
| **CO3** | Employ finite state machines to solve problems in computing | **K3** |
| **CO4** | Explain deterministic and non-deterministic machines | **K2** |
| **CO5** | Comprehend the hierarchy of problems arising in the computer science | **K5** |

**#based on suggested Revised BTL**

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 3 | 2 | 3 | 1 | 2 | | | 2 | | | 2 | |
| **CO2** | 3 | 2 | 2 | 1 | 2 | | | | | | | |
| **CO3** | 2 | 1 | 2 | | 3 | | | 1 | | | 1 | |
| **CO4** | 3 | 2 | 3 | | 1 | | | 1 | | | | |
| **CO5** | | 3 | 2 | 3 | 1 | 5 | | | | | 1 | |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**Syllabus:**

**UNIT-I: Finite Automata:** Theory of computer science introduction (Automata theory, Formal Languages and Computation), Central Concepts of Automata Theory, Automation, Finite Automation (FA), Transition Systems, Acceptance of a String, DFA, Design of DFAs, NFA, Design of NFA, Equivalence of DFA and NFA, Conversion of NFA into DFA, Finite Automata with $\in$-Transitions, Minimization of Finite Automata, Finite Automata with output-Mealy and Moore Machines Equivalence of Moore and Mealy machines, Applications and Limitation of Finite Automata

**UNIT-II: Regular Expressions:** Regular Expressions(RE), Regular Sets, Identity Rules, Equivalence of two RE, Manipulations of REs, Finite Automata and Regular Expressions, Inter Conversion, Equivalence between FA and RE, Pumping Lemma of Regular Sets, Closure Properties of Regular Sets, Decision algorithms for regular sets, Phrase structure Grammars, classification of grammars Chomsky Hierarchy, Right and Left Linear Regular Grammars, Equivalence between Regular Grammars and FA, Inter Conversion

**UNIT-III: Context Free Grammars:** Formal Languages, Context Free Grammar, Leftmost and Rightmost Derivations, Parse Trees, Ambiguous Grammars, Simplification of Context Free Grammars-Elimination of Useless Symbols, ∊-Productions and Unit Productions, construction of reduced grammar , Normal Forms-Chomsky Normal Form and Greibach Normal Form, Pumping Lemma for Context free languages, Closure Properties, Applications of Context Free Grammars

**UNIT-IV: Pushdown Automata:** Pushdown Automata, Definition, Model, Graphical Notation, Instantaneous Description, Language Acceptance of Pushdown Automata, Design of Pushdown Automata, Deterministic and Non – Deterministic Pushdown Automata, Equivalence of Pushdown Automata and Context Free Grammars, Conversion, Two Stack Pushdown Automata, Application of Pushdown Automata

**UNIT-V: Turning Machine(TM):** Definition, Model, Representation of TMs-Instantaneous Descriptions, Transition Tables and Transition Diagrams, Language of a TM, Design of TMs, Types of TMs, Church's Thesis, Universal and Restricted TM, Decidable and Un-decidable Problems, Halting Problem of TMs, Post's Correspondence Problem(PCP) , Modified PCP, Classes of P and NP, NP-Hard and NP-Complete Problems.

**TEXT BOOKS:**

1. Introduction to Automata Theory, Languages and Computation, J. E. Hopcroft, R. Motwani and J. D. Ullman, 3rd Edition, Pearson, 2008
2. Theory of Computer Science-Automata, Languages and Computation, K. L. P. Mishra and N. Chandrasekharan, 3rd Edition, PHI, 2007
3. Elements of Theory of Computation, Lewis H.P. &Papadimitiou C.H., Pearson /PHI

**REFERENCE BOOKS:**

1. Formal Language and Automata Theory, K. V. N. Sunitha and N. Kalyani, Pearson, 2015
2. Introduction to Automata Theory, Formal Languages and Computation, ShyamalenduKandar, Pearson, 2013
3. Theory of Computation, V. Kulkarni, Oxford University Press, 2013
4. Theory of Automata, Languages and Computation, Rajendra Kumar, McGraw Hill, 2014

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**II B. Tech II Sem- R20**

**COMPUTER NETWORKS**
**Code: R20PC2201**

**Course Objectives:**

- To provide insight about networks, topologies, and the key concepts.
- To gain comprehensive knowledge about the layered communication architectures (OSI and TCP/IP) and its functionalities.
- To understand the principles, key protocols, design issues, and significance of each layers in ISO and TCP/IP.
- To know the basic concepts of network services and various network applications.

**Course Outcomes (COs)**: At the end of the course, student will be able to

| Course Outcomes | | Knowledge Level (K)# |
|---|---|---|
| **CO1** | **Demonstrate** different network models for networking links OSI, TCP/IP get knowledge about various communication techniques, methods and protocol standards. | **K2** |
| **CO2** | **Discuss** different transmission media and different switching networks. | **K6** |
| **CO3** | **Analyze** data link layer services, functions and protocols like HDLC and PPP. | **K4** |
| **CO4** | **Compare and Classify** medium access control protocols like ALOHA, CSMA, CSMA/CD, CSMA/CA, Polling, Token passing, FDMA, TDMA, CDMA protocols. | **K4,K2** |
| **CO5** | **Determine** application layer services and client server protocols working with the client server paradigms like WWW, HTTP, FTP, e-mail and SNMP etc. | **K5** |

***#Based on suggested Revised BTL***

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 1 | 3 | 2 | | | | | | | 1 | | |
| **CO2** | 2 | 1 | 1 | 3 | | | | | | 1 | | |
| **CO3** | 2 | 3 | | 1 | | | | | | | | |
| **CO4** | 1 | 2 | 3 | | | | | | | | | |
| **CO5** | | 1 | 3 | | | 2 | | | | | | |

**(Please fill the above with Levels of Correlation, viz., L-1, M-2, H-3)**

**Department of Computer Science & Engineering**
**University College of Engineering, JNT University Kakinada**

**Syllabus:**

**UNIT I: Introduction:** Network Types, LAN, MAN, WAN, Network Topologies Reference models- The OSI Reference Model- the TCP/IP Reference Model - A Comparison of the OSI and TCP/IP Reference Models, OSI Vs TCP/IP, Lack of OSI models success, Internet History.

**Physical Layer –**Introduction to Guided Media- Twisted-pair cable, Coaxial cable and Fiber optic cable and unguided media: Wireless-Radio waves, microwaves, infrared.

**UNIT II: Data link layer:** Design issues, **Framing**: fixed size framing, variable size framing, flow control, error control, error detection and correction codes, CRC, Checksum: idea, one's complement internet checksum, services provided to Network Layer, **Elementary Data Link Layer protocols**: simplex protocol, Simplex stop and wait, Simplex protocol for Noisy Channel.

**Sliding window protocol:** One bit, Go back N, Selective repeat-Stop and wait protocol, Data link layer in HDLC: configuration and transfer modes, frames, control field, point to point protocol (PPP): framing transition phase, multiplexing, multi link PPP.

**UNIT – III: Media Access Control: Random Access**: ALOHA, Carrier sense multiple access (CSMA), CSMA with Collision Detection, CSMA with Collision Avoidance, **Controlled Access:** Reservation, Polling, Token Passing, **Channelization:** frequency division multiple Access(FDMA), time division multiple access(TDMA), code division multiple access(CDMA).

**Wired LANs:** Ethernet, Ethernet Protocol, Standard Ethernet, Fast Ethernet(100 Mbps), Gigabit Ethernet, 10 Gigabit Ethernet.

**UNIT – IV: The Network Layer Design Issues** – Store and Forward Packet Switching-Services Provided to the Transport layer- Implementation of Connectionless Service-Implementation of Connection Oriented Service- Comparison of Virtual Circuit and Datagram Networks, Routing Algorithms-The Optimality principle-Shortest path, Flooding, Distance vector, Link state, Hierarchical, Congestion Control algorithms-General principles of congestion control, Congestion prevention polices, Approaches to Congestion Control-Traffic Aware Routing- Admission Control-Traffic Throttling-Load Shedding. Traffic Control Algorithm-Leaky bucket & Token bucket.

**Internet Working:** How networks differ- How networks can be connected- Tunnelling, internetwork routing-, Fragmentation, network layer in the internet – IP protocols-IP Version 4 protocol-IPV4 Header Format, IP addresses, Class full Addressing, CIDR, NAT-, Subnets-IP Version 6-The main IPV6 header, Transition from IPV4 to IPV6, Comparision of IPV4 & IPV6- Internet control protocols- ICMP-ARP-DHCP

**UNIT –V: The Transport Layer:** Transport layer protocols: Introduction-services-port number-User data gram protocol-User datagram-UDP services-UDP applications-Transmission control protocol: TCP services- TCP features- Segment-A TCP connection- windows in TCP- flow control-Error control, Congestion control in TCP.

**Application Layer** –- World Wide Web: HTTP, Electronic mail-Architecture- web based mail- email security- TELENET-local versus remote Logging-Domain Name System: Name Space, DNS in Internet ,- Resolution-Caching- Resource Records-DNS messages- Registrars-security of DNS Name Servers, SNMP.

**Text Books:**

1. Computer Networks — Andrew S Tanenbaum, Fifth Edition. Pearson Education/PHI
2. Data Communications and Networks – Behrouz A. Forouzan, Fifth Edition TMH.

**References Books:**

1. Data Communications and Networks- Achut S Godbole, Atul Kahate
2. Computer Networks, Mayank Dave, CENGAGE

**Department of Computer Science & Engineering**
**University College of Engineering, JNT University Kakinada**

**II B. Tech II Sem- R20**

**PYTHON PROGRAMMING**
**Code: R20PC2202**

**COURSE OUTCOMES**: At the end of the course, student will be able to

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| CO1 | Understand and comprehend the basics of python programming. Demonstrate the principles of structured programming and be able to describe, design, implement, and test structured programs using currently accepted methodology | K2 |
| CO2 | Demonstrate the principles of structured programming and be able to describe, design, implement, and test structured programs using currently accepted methodology. Explain the use of the built-in data structures list, sets, tuples and dictionary | K3 |
| CO3 | Understand of functions and its applications, Modules. | K3 |
| CO4 | Identify real-world applications using oops, files and exception handling provided by python. | K3 |
| CO5 | Formulate and implement a program to solve a real-world problem using GUI and Turtle graphics. | K3 |

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 1 | | | 2 | | 2 | | 3 | | | | |
| CO2 | 2 | | | 2 | | 3 | | 2 | | | | |
| CO3 | | | | | | 2 | | 2 | | | | |
| CO4 | | | | 3 | | | | 3 | | | | |
| CO5 | | | | | | | | | | | | |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**Syllabus:**

**UNIT – 1 Introduction and Strings:** History of Python, Need of Python Programming, Applications Basics of Python Programming Python Basics, Objects- Python Objects, Standard Types, Other Built-in Types, Internal Types, Standard Type Operators, Standard Type Built-in Functions, Categorizing the Standard Types, Unsupported Types Numbers - Introduction to Numbers, Integers, Floating Point Real Numbers, Complex Numbers, Operators, Built-in Functions, Related Modules Sequences - Strings, Lists, and Tuples, Mapping and Set Types. Introduction to Numbers, Integers, Floating Point Real Numbers, Complex Numbers, Operators, Built-in Functions, Related Modules Sequences - Strings, Lists, and Tuples, Mapping and Set Types.

**UNIT – 2: FILES**: File Objects, File Built-in Function [ open() ], File Built-in Methods, File Built-in Attributes, Standard Files, Command-line Arguments, File System, File Execution, Persistent Storage Modules.

**Errors and Exceptions**: Difference between an error and Exception, Handling Exception, try except block, Raising Exceptions, User Defined Exceptions Detecting and Handling Exceptions, Context Management, **Modules**: Modules and Files, Namespaces, Importing Modules, Importing Module Attributes, Module Built-in Functions, Packages, Other Features of Modules

**UNIT – 3: OOP in Python and Regular Expressions:** Classes and objects, Inheritance, Overriding Methods, Data-hiding, Polymorphism, introduction to regular expressions Introduction, Special Symbols and Characters, Res and Python Multithreaded Programming: Introduction, Threads and Processes, Python, Threads, and the Global Interpreter Lock, Thread Module, Threading Module.

**UNIT – 4: GUI Programming and Testing:** Introduction, Tkinter and Python Programming, Brief Tour of Other GUIs, Related Modules and Other GUIs WEB Programming: Introduction, Wed Surfing with Python, Creating Simple Web Clients, Advanced Web Clients, CGI-Helping Servers Process Client Data, Building CGI Application Advanced CGI, Web (HTTP) Servers, **Testing:** Why testing is required? Basic concepts of testing, Unit testing in Python, Writing Test cases, Running Tests

**UNIT – 5 Database Programming:** Introduction, Python Database Application Programmer's Interface (DB-API), Object Relational Managers (ORMs), Related Modules

**TEXT BOOKS:**

1. Core Python Programming, W.Chun, Pearson.
2. Fundamentals of Python First Programs, Kenneth. A. Lambert, Cengage.

**REFERENCE BOOKS:**

1. Python Programming: A Modern Approach, Vamsi Kurama, Pearson.
2. Think Python, Allen Downey, Green Tea Press
3. Introduction to Python Programming, Gowrishankar.S, Veena A, CRC Press.

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**II B. Tech II Sem- R20**

**R PROGRAMMING LAB**
**Code: R20PC2203**

**Exercise 1:** a) Write an R program to print Hello World
b) Write a program to demonstrate the basic Arithmetic in R
c) Write a program to demonstrate the Variable assignment in R

**Exercise 2: a)** Write a program to demonstrate the data types in R
b) Write a program to demonstrate the creating and naming a vector in R
c) Write a program to demonstrate the create a matrix and naming matrix in R

**Exercise 3:**
**a)** Write a program to demonstrate the Add column and Add a Row in Matrix in R
**b)** Write a program to demonstrate the selection of elements in Matrixes in R
**c)** Write a program to demonstrate the Performing Arithmetic of Matrices

**Exercise 4: a)** Write a program to demonstrate the Factors in R
b) Implement the Factor in R
c) Write a program to illustrate Ordered Factors in R

**Exercise 5:**
**a)** Write an R program to take input from the user.
b) Write an R program to Check if a Number is Odd or Even
c) Write an R program to check if the given number is a Prime Number

**Exercise 6:**
a) Write an R program to Find the Factorial of a Number
b) Write an R program to Find the Factors of a Number
c) Write an R program to Find the Fibonacci sequence Using Recursive Function

**Exercise 7:**
**a)** Write an R program to Make a Simple Calculator
b) Write an R program to Find L.C.M of two numbers
c) Write an R program to create a Vector and to access elements in a Vector

**Exercise 8:**
a) Write an R program to create a Matrix and access rows and columns using functions *colnames()* and *rownames()* .
b) Write an R Program to create a Matrix using *cbind()* and *rbind()* functions.

**Department of Computer Science & Engineering**
**University College of Engineering, JNT University Kakinada**

**Exercise 9:**

**a)** Write an R Program to create a Matrix from a Vector using the ***dim()*** function.
b) Write an R Program to create a List and modify its components.
c) Write an R Program to create a Data Frame.

**Exercise 10:**

**a)** Write an R Program to access a Data Frame like a List.
b) Write an R Program to access a Data Frame like a Matrix.
c) Write an R Program to create a Factor.

**Exercise 11:**

**a)** Write an R Program to Access and Modify Components of a Factor.
b) Write an R Program to create an S3 Class and S3 Objects.
c) Write an R Program to write an own generic function in S3 Class.

**Exercise 12:**

**a)** Write an R Program to create an S4 Class and S4 Objects.
b) Write an R Program to write an own generic function in S4 Class.
c) Write an R Program to create a reference class and modify its methods.

**Exercise 13:**

**a)** Write an R program to create a scatter plot for the data frame columns.
b) Write an R program to create a bar plot for the data frame columns.
c) Write an R program to create a box plot for the data frame columns.

**Exercise 14:**

**a)** Write an R program to add the legend to the plot.
b) Write an R program to change the width and height of the plot layout
c) Write an R program to calculate mean, mode, median and standard devotion

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**II B. Tech II Sem- R20**

**PYTHON PROGRAMMING LAB**
**Code: R20PC2204**

**COURSE OUTCOMES**: At the end of the course, student will be able to

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| CO1 | Understand the Numbers, Math functions, Strings, List, Tuples and Dictionaries in Python | K2 |
| CO2 | Demonstrate different Decision-Making statements and Functions. | K4 |
| CO3 | Interpret Object oriented programming in Python. | K3 |
| CO4 | Demonstrate and summarize different File handling operations. | K4 |
| CO5 | Identify how to design GUI Applications in Python and evaluate different database operations | K3 |
| CO6 | Design and develop Client Server network applications using Python. | K4 |

***#Based on suggested Revised BTL***

**Syllabus:**
**List of Experiments*:***
**Exercise- 1**:
a. Write a program to demonstrate different number data types in Python.
b. Write a program to perform different Arithmetic Operations on numbers in Python.
c. Write a program to create, concatenate and print a string and accessing sub-string from a given string.

**Exercise- 2**:
a. Write a python script to print the current date in the following format "WED 09 02:26:23 IST 2020"
b. Write a Python program to convert temperatures to and from Celsius, Fahrenheit [Formula: c/5 = f-32/9]
c. Write a Python script that prints prime numbers less than 20.

**Exercise- 3**:
a. Write a python program to find factorial of a number using Recursion.
b. Write a program that accepts the lengths of three sides of a triangle as inputs. The program output should indicate whether or not the triangle is a right triangle (Recall from the Pythagorean Theorem that in a right triangle, the square of one side equals the sum of the squares of the other two sides).
c. Write a python program to define a module to find Fibonacci Numbers and import the module to another program.

**Department of Computer Science & Engineering**
**University College of Engineering, JNT University Kakinada**

**Exercise- 4**:

a. Write a script named copyfile.py. This script should prompt the user for the names of two text files. The contents of the first file should be input and written to the second file.
b. Write a program that inputs a text file. The program should print all of the unique words in the file in alphabetical order.
c. Write a Python class to convert an integer to a roman numeral.

**Exercise- 5:**

a. Write a Python class to implement pow(x, n)
b. Write a Python class to reverse a string word by word.

**Exercise- 6:**

a. Write a program to count the numbers of characters in the string and store them in a dictionary data structure
b. Write a program to use split and join methods in the string and trace a birthday with a dictionary data structure.

**Exercise- 7** :

a. Write a program combine lists that combines these lists into a dictionary.
b. Write a program to count frequency of characters in a given file. Can you use character frequency to tell whether the given file is a Python program file, C program file or a text file?

**Exercise- 8:**

a. Write a function ball collides that takes two balls as parameters and computes if they are colliding. Your function should return a Boolean representing whether or not the balls are colliding.
   **Hint:** Represent a ball on a plane as a tuple of (x, y, r), r being the radius If (distance between two balls centers) <= (sum of their radii) then (they are colliding)
b. Find mean, median, mode for the given set of numbers in a list

**Exercise- 9:**

a. Write a function nearly equal to test whether two strings are nearly equal. Two strings a and b are nearly equal when a can be generated by a single mutation on b.
b. Write a function dups to find all duplicates in the list.
c. Write a function unique to find all the unique elements of a list.

**Exercise- 10:**

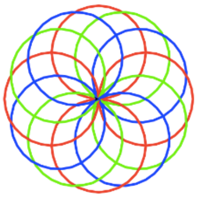
a. Write a program that defines a matrix and prints
b. Write a program to perform addition of two square matrices
c. Write a program to perform multiplication of two square matrices

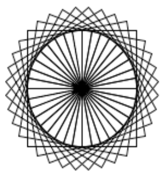
**Exercise- 11:**

a. Install packages requests, flask and explore them. using (pip)
b. Write a script that imports requests and fetch content from the page. Eg. (Wiki)
c. Write a simple script that serves a simple HTTP Response and a simple HTML Page

**Exercise- 12:**

a. Write a GUI for an Expression Calculator using tk .
b. Write a program to implement the following figures using turtle





**Exercise- 13:**

a. Write a test-case to check the function even numbers which return True on passing a list of all even numbers
b. Write a test-case to check the function reverse string which returns the reversed string

**Exercise- 14:**

a. In algebraic expressions, the symbol for multiplication is often left out, as in 3x+4y or 3(x+5). Computers prefer those expressions to include the multiplication symbol, like 3*x+4*y or 3*(x+5). Write a program that asks the user for an algebraic expression and then inserts multiplication symbols where appropriate.
b. Write a program that asks the user to enter a length in feet. The program should then give the user the option to convert from feet into inches, yards, miles, millimeters, centimeters, meters, or kilometers. Say if the user enters a 1, then the program converts to inches, if they enter a 2, then the program converts to yards, etc. While this can be done with if statements, it is much shorter with lists and it is also easier to add new conversions if you use lists.

**Exercise- 15:**

a. Write a program that reads a file consisting of email addresses, each on its own line. Your program should print out a string consisting of those email addresses separated by semicolons.
b. Write a program that reads a list of temperatures from a file called temps.txt, converts those temperatures to Fahrenheit, and writes the results to a file called ftemps.txt.

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**II B. Tech II Sem- R20**

**COMPUTER NETWORKS LAB**
**Code: R20PC2205**

**List of Experiments:**

1. Study of Network devices in detail and connect the computers in Local Area Network.
2. Write a Program to implement the data link layer farming methods such as i) Character stuffing ii) bit stuffing.
3. Write a Program to implement data link layer farming method checksum.
4. Write a program for Hamming Code generation for error detection and correction.
5. Write a Program to implement on a data set of characters the three CRC polynomials – CRC 12, CRC 16 and CRC CCIP.
6. Write a Program to implement Sliding window protocol for Goback N.
7. Write a Program to implement Sliding window protocol for Selective repeat.
8. Write a Program to implement Stop and Wait Protocol.
9. Write a program for congestion control using leaky bucket algorithm
10. Write a Program to implement Dijkstra‘s algorithm to compute the Shortest path through a graph.
11. Write a Program to implement Distance vector routing algorithm by obtaining routing table at each node (Take an example subnet graph with weights indicating delay between nodes).
12. Write a Program to implement Broadcast tree by taking subnet of hosts.
13. Wireshark
    i. Packet Capture Using Wire shark
    ii. Starting Wire shark
    iii. Viewing Captured Traffic
    iv. Analysis and Statistics & Filters.
14. How to run Nmap scan
15. Operating System Detection using Nmap
16. Do the following using NS2 Simulator
    i. NS2 Simulator-Introduction
    ii. Simulate to Find the Number of Packets Dropped
    iii. Simulate to Find the Number of Packets Dropped by TCP/UDP
    iv. Simulate to Find the Number of Packets Dropped due to Congestion
    v. Simulate to Compare Data Rate& Throughput.

**Compiler Design Lab**

1. Write a C program to identify different types of Tokens in a given Program.
2. Implementation of Lexical Analyzer using Lex tool
3. Write a C program to Simulate Lexical Analyzer to validating input String.
4. Write a program to implement Brute force technique of Top down parsing.
5. Implementation of Recursive Descent Parser

6. Write program to calculate First and Follow Functions Sets for the given grammar
7. Write a program to check the validity of input string using Predictive Parser.
8. Implement a Shift Reduce Parser using Stack Data Structure for a given grammar.
9. Implementation of calculator using LEX and YACC tool
   1. Generate YACC specification for a few syntactic categories.
   2. Write a program for generating various intermediate code forms
      i) Three address code    ii) Polish notation
10. Write a program to perform loop unrolling and constant propagation

**Department of Computer Science & Engineering**
**University College of Engineering, JNT University Kakinada**

**II B. Tech II Seme-R20**

**PYTHON PROGRAMMING: NLP**
**Code: R20SO2201**

## Pre-requisites:

- Basic Knowledge of Natural Language Processing
- Hands-on practice of Python
- Basic idea of NLP using the library- SpaCy

**Experiment-1:**

a) Getting started with NLTK, install NLTK using PIP
b) Try using the Python interpreter as a calculator, and typing expressions like 12 / (4 + 1).

**Experiment-2:**

a) Define a string and assign it to a variable, e.g., my_string = 'My String' (but put something more interesting in the string). Print the contents of this variable in two ways, first by simply typing the variable name and pressing enter, then by using the print statement.
b) Try adding the string to itself using my_string + my_string, or multiplying it by a number, e.g., my_string * 3. Notice that the strings are joined together without any spaces. How could you fix this?

**Experiment-3:** Define a variable my_sent to be a list of words, using the syntax my_sent = ["My", "sent"] (but with your own words, or a favorite saying).

a. Use ' '.join(my_sent) to convert this into a string.
b. Use split() to split the string back into the list form you had to start with.

**Experiment-4:** Write expressions for finding all words in text6 that meet the conditions listed below. The result should be in the form of a list of words: ['word1', 'word2', ...].

a. Ending in *ise*
b. Containing the letter *z*
c. Containing the sequence of letters *pt*
d. Having all lowercase letters except for an initial capital (i.e., titlecase)

**Department of Computer Science & Engineering**
**University College of Engineering, JNT University Kakinada**

**Experiment-5:** Define sent to be the list of words ['she', 'sells', 'sea', 'shells', 'by', 'the', 'sea', 'shore']. Now write code to perform the following tasks:

a. Print all words beginning with *sh*
b. Print all words longer than four characters

**Experiment-6:** Write a program by using list addition, and the set and sorted operations, compute the vocabulary of the sentences sent1 ... sent8.

**Experiment-7:** Write a program by using text9. index() to find the index of the word *sunset*. You'll need to insert this word as an argument between the parentheses. By a process of trial and error, find the slice for the complete sentence that contains this word.

**Experiment-8:** Write a program for defining a function percent (word, text) that calculates how often a given word occurs in a text, and expresses the result as a percentage.

**Experiment-9:** Write a program for defining a function called vocab_size (text) that has a single parameter for the text, and which returns the vocabulary size of the text.

**Experiment-10:** Write a program to find all the four-letter words in the Chat Corpus (text5). With the help of a frequency distribution (FreqDist), show these words in decreasing order of frequency.

**Experiment-11:** Tom wants to help his sister Mary to pass through the aptitude exam, so in order to help her he wants to test her skills in English .So he decided to assign some sentences and want the key root words in the sentences. So implement a python code that helps Mary to get the root words in the given sentences? (Note: Use stemming and tokenization process)

**Experiment-12:** Tom is a comic editor in a X company one day while editing a particular script he became enthusiastic about the story of the script so there is no time to read the whole script he decided to understand the total story line by learning about the characters so he wants to separate the words in the sentence to know the characters as the whole story is complex ,So implement a python program that splits the words and display both splitted words and count of the words in the given sentence using tokenizer function?

**REFERENCES:**

1. https://www.nltk.org/book/ch01.html
2. https://www.kaggle.com/akasireddy99/nlp-lab-manual

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**III B. Tech I Sem-R20**

### COMPILER DESIGN
### Code: R20PC3101

**COURSE OUTCOMES:** At the end of the course, student will be able to

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | **Demonstrate** phases in the design of compiler | **K2** |
| **CO2** | **Organize** Syntax Analysis, Top Down and LL(1) grammars | **K3** |
| **CO3** | **Design** Bottom Up Parsing and Construction of LR parsers | **K6** |
| **CO4** | **Analyze** synthesized, inherited attributes and syntax directed translation schemes | **K4** |
| **CO5** | **Determine** algorithms to generate code for a target machine | **K5** |

**#Based on suggested Revised BTL**

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 2 | 2 | 3 | | 2 | 1 | 1 | 1 | 1 | 2 | | |
| **CO2** | 3 | 3 | 2 | | 3 | | 1 | | 2 | 3 | | |
| **CO3** | 2 | 3 | 3 | | 2 | 1 | 1 | | 1 | 2 | | |
| **CO4** | 3 | 3 | 2 | | 3 | 1 | 2 | | 3 | 2 | | |
| **CO5** | 2 | 3 | 2 | | | | 2 | | 2 | 3 | | |

**(Please fill the above with Levels of Correlation, viz., L-1, M-2, H-3)**

**SYLLABUS:**

**UNIT-1: Lexical Analysis:** Language Processors, Structure of a Compiler, Lexical Analysis, The Role of the Lexical Analyzer, Bootstrapping, Input Buffering, Specification of Tokens, Recognition of Tokens, Lexical Analyzer Generator-LEX, Finite Automata, Regular Expressions and Finite Automata, Design of a Lexical Analyzer Generator.

**UNIT-2: Syntax Analysis:** The Role of the Parser, Context-Free Grammars, Derivations, Parse Trees, Ambiguity, Left Recursion, Left Factoring, **Top Down Parsing:** Pre Processing Steps of Top Down Parsing, Backtracking, Recursive Descent Parsing, LL (1) Grammars, Non-recursive Predictive Parsing, Error Recovery in Predictive Parsing.

**UNIT-3: Bottom Up Parsing:** Introduction, Difference between LR and LL Parsers, Types of LR Parsers, Shift Reduce Parsing, SLR Parsers, Construction of SLR Parsing Tables, More Powerful LR Parses, Construction of CLR (1) and LALR Parsing Tables, Dangling Else Ambiguity, Error Recovery in LR Parsing, Handling Ambiguity Grammar with LR Parsers.

**UNIT-4: Syntax Directed Translation:** Syntax-Directed Definitions, Evaluation Orders for SDD's, Applications of Syntax Directed Translation, Syntax-Directed Translation Schemes, Implementing L-Attributed SDD's. **Intermediate Code Generation:** Variants of Syntax Trees, Three Address Code, Types and Declarations, Translation of Expressions, Type Checking, Control Flow, Backpatching, Intermediate Code for Procedures.

**UNIT-5: Run Time Environments:** Storage Organization, Run Time Storage Allocation, Activation Records, Procedure Calls, Displays, **Code Optimization:** The Principle Sources of Optimization, Basic Blocks, Optimization of Basic Blocks, Structure Preserving Transformations, Flow Graphs, Loop Optimization, Data-Flow Analysis, Peephole Optimization**, Code Generation:** Issues in the Design of a Code Generator, Object Code Forms, Code Generation Algorithm, Register Allocation and Assignment.

**TEXT BOOKS:**

1. Compilers: Principles, Techniques and Tools, Second Edition, Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffry D. Ullman, Pearson Publishers, 2007.

**REFERENCE BOOKS:**

1. Compiler Construction, Principles and Practice, Kenneth C Louden, Cengage Learning, 2006
2. Modern compiler implementation in C, Andrew W Appel, Revised edition, Cambridge University Press.
3. Optimizing Compilers for Modern Architectures, Randy Allen, Ken Kennedy, Morgan Kauffmann, 2001.
4. Levine, J.R., T. Mason and D. Brown, Lex and Yacc, edition, O'Reilly & Associates, 1990

**Department of Computer Science & Engineering**
**University College of Engineering, JNT University Kakinada**

**III B. Tech I Sem-R20**

## DESIGN AND ANALYSIS OF ALGORITHMS
**Code: R20PC3102**

**COURSE OBJECTIVES:**

Upon completion of this course, students will be able to do the following:

- Ability to understand, analyze and denote time complexities of algorithms
- To introduce the different algorithmic approaches for problem solving through numerous example problems
- Describe the dynamic-programming paradigm and explain when an algorithmic design situation calls for it. Recite algorithms that employ this paradigm. Synthesize dynamic-programming algorithms, and analyze them.
- To provide some theoretical grounding in terms of finding the lower bounds of algorithms and the NP-completeness

**COURSE OUTCOMES:** After the completion of the course, student will be able to

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Analyze the performance of a given algorithm, denote its time complexity using the asymptotic notation for recursive and non-recursive algorithms | **K4** |
| **CO2** | List and describe various algorithmic approaches and Solve problems using divide and conquer &greedy Method | **K2** |
| **CO3** | Synthesize efficient algorithms dynamic programming approaches to solve in common engineering design situations. | **K3** |
| **CO4** | Organize important algorithmic design paradigms and methods of analysis: backtracking, branch and bound algorithmic approaches | **K4** |
| **CO5** | Demonstrate NP- Completeness theory ,lower bound theory and String Matching | **K6** |

**#Based on suggested Revised BTL**

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 1 | 3 | 2 | 2 | | 1 | | | | | | |
| **CO2** | 2 | 3 | 2 | | 3 | | | | | | | |
| **CO3** | 3 | 1 | 3 | | 2 | | | | | | | |
| **CO4** | 3 | 1 | 2 | | 3 | 1 | | | | | | |
| **CO5** | 1 | 2 | 3 | 3 | | | | | | | | |

**(Please fill the above with Levels of Correlation, viz., L-1, M-2, H-3)**

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**SYLLABUS:**

**UNIT I: Introduction:** Algorithm- introduction, Algorithm characteristics, Pseudo code Conventions Recursive Algorithm, Performance Analysis, Space Complexity, Time Complexity, Probabilistic Complexity, Asymptotic Notation, Practical Complexities, Big and little- O notations.

**UNIT II: Divide and Conquer:** General Method, Defective chessboard, Binary Search, finding the maximum and minimum, Merge sort, Quick sort. **The Greedy Method:** The general Method, knapsack problem, Job sequencing with deadlines, minimum-cost spanning Trees, Optimal Merge Patterns, Single Source Shortest Paths.

**UNIT III: Dynamic Programming:** General Method, All pairs-shortest paths, String Editing, 0/1 knapsack, Reliability Design, Floyd-Warshall algorithm for shortest path.

**UNIT IV: Backtracking:** The General Method, The 8-Queens problem, sum of subsets, Graph coloring, Hamiltonian cycles, knapsack problem.

**Branch and Bound:** The Method, Least cost (LC) Search, The 15- Puzzle: an Example, Control Abstraction for LC-Search.

**UNIT V:** Bounding, FIFO Branchand-Bound, LC Branch and Bound, 0/1 Knapsack Problem, Traveling Salesperson problem,Introduction to NP-Hard and NP- Completeness - Decision Trees - P, NP and NP - CompleteProblems - Cook's theorem.

**TEXT BOOKS:**

1. Ellis Horowitz, SartajSahni, Sanguthevar Rajasekaran, "Fundamentals of Computer Algorithms", 2ndEdition, Universities Press.
2. Introduction to Algorithms Thomas H. Cormen, PHI Learning

**REFERENCE BOOKS:**

1. Harsh Bhasin, "Algorithms Design & Analysis", Oxford University Press.
2. Alfred V. Aho, John E. Hopcroft, Jeffrey D. Ullman, "The Design and Analysis of Computer Algorithms", Pearson Education India, 1 January 2004.
3. S. Sridhar, "Design and Analysis of Algorithms", Oxford University Press.

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**III B. Tech I Sem-R20**

**DATA MINING TECHNIQUES**
**Code: R20PC3103**

**Pre-requisites:** Data Structures, Algorithms, Probability & Statistics, Data Base Management Systems

**COURSE OBJECTIVES:** The main objective of the course is to

- Introduce basic concepts and techniques of data warehousing and data mining
- Examine the types of the data to be mined and apply pre-processing methods on raw data
- Discover interesting patterns, analyze supervised and unsupervised models and estimate the accuracy of the algorithms.

**COURSE OUTCOMES:** By the end of the course student will be able to

| CO | Course Outcomes | Knowledge Level(#) |
|---|---|---|
| CO1 | Illustrate the importance of Data Warehousing, Data Mining and its functionalities and Design schema for real time data warehousing applications. | K3, K6 |
| CO2 | Demonstrate on various Data Preprocessing Techniques viz. data cleaning, data integration, data transformation and data reduction and Process raw data to make it suitable for various data mining algorithms. | K3 |
| CO3 | Choose appropriate classification technique to perform classification, model building and evaluation. | K4 |
| CO4 | Make use of association rule mining techniques viz. Apriori and FP Growth algorithms and analyze on frequent item-sets generation. | K3, K4 |
| CO5 | Identify and apply various clustering algorithm (with open source tools), interpret, evaluate and report the result. | K2 |

**Mapping of course outcomes with program outcomes**

| | PO 1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO1 0 | PO1 1 | PO1 2 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 2 | 2 | 2 | 1 | 2 | | | | | 2 | | |
| CO2 | 3 | 3 | 2 | 2 | 3 | | | | | 1 | | |
| CO3 | 3 | 3 | 3 | 3 | 3 | | | | | 2 | | |
| CO4 | 1 | 2 | 2 | 3 | 3 | | | | | 1 | | |
| CO5 | 3 | 3 | 2 | 3 | 3 | | | | | 2 | | |

**(Please fill the above with levels of Correlation, viz., L-1,M- 2, H-3)**

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**SYLLABUS:**

**UNIT–I: Data Warehousing and Online Analytical Processing:** Basic concepts, Data Warehouse Modeling: Data Cube and OLAP, Data Warehouse Design and Usage, Data Warehouse Implementation, Cloud Data Warehouse, Data Mining and Patten Mining, Technologies, Applications, Major issues, Data Objects & Attribute Types, Basic Statistical Descriptions of Data, Data Visualization, Measuring Data Similarity and Dissimilarity.

**UNIT II: Data Preprocessing:** An Overview, Data Cleaning, Data Integration, Data Reduction, Data Transformation and Data Discretization.

**UNIT–III: Classification:** Basic Concepts, General Approach to solving a classification problem, Decision Tree Induction: Attribute Selection Measures, Tree Pruning, Scalability and Decision Tree Induction, Visual Mining for Decision Tree Induction, Bayesian Classification Methods: Bayes Theorem, Naïve Bayes Classification, Rule-Based Classification, Model Evaluation and Selection.

**UNIT–IV: Association Analysis:** Problem Definition, Frequent Itemset Generation, Rule Generation: Confident Based Pruning, Rule Generation in Apriori Algorithm, Compact Representation of frequent item sets, FP-Growth Algorithm.

**UNIT–V: Cluster Analysis:** Overview, Basics and Importance of Cluster Analysis, Clustering techniques, Different Types of Clusters; K-means: The Basic K-means Algorithm, K-means Additional Issues, Bi-secting K Means, Agglomerative Hierarchical Clustering: Basic Agglomerative Hierarchical Clustering Algorithm DBSCAN: Traditional Density Center-Based Approach, DBSCAN Algorithm, Strengths and Weaknesses.

**TEXT BOOKS:**

1. Data Mining concepts and Techniques, 3rd edition, Jiawei Han, Michel Kamber, Elsevier, 2011.
2. Introduction to Data Mining: Pang-Ning Tan & Michael Steinbach, Vipin Kumar, Pearson, 2012.

**REFERENCE BOOKS:**

1. Data Mining: VikramPudi and P. Radha Krishna, Oxford Publisher.
2. Data Mining Techniques, Arun K Pujari, 3rd edition, Universities Press,2013.
3. Data Warehousing Data Mining & OLAP, Alex Berson, Stephen Smith, TMH.
   http://onlinecourses.nptel.ac.in/noc18_cs14/preview
4. (NPTEL course by Prof.PabitraMitra)
   http://onlinecourses.nptel.ac.in/noc17_mg24/preview
5. (NPTEL course by Dr. NandanSudarshanam& Dr. BalaramanRavindran)
   http://www.saedsayad.com/data_mining_map.htm

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**III B. Tech I Sem-R20**

**OPTIMIZATION TECHNIQUES**
**Code: R20OE3101**

**COURSE OBJECTIVES:**

- To define an objective function and constraint functions in terms of design variables, and then state the optimization problem.
- To state single variable and multi variable optimization problems, without and with constraints.
- To explain linear programming technique to an optimization problem, define slack and surplus variables, by using Simplex method.
- To state transportation and assignment problem as a linear programming problem to determine Simplex method.
- To study and explain nonlinear programming techniques, unconstrained or constrained, and define exterior and interior penalty functions for optimization problems.

**COURSE OUTCOMES**: At the end of the course, student will be able to

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | State and formulate the optimization problem, without and with constraints, by using design variables from an engineering design problem. | **K2** |
| **CO2** | Apply classical optimization techniques to minimize or maximize a multi-variable objective function, without or with constraints, and arrive at an optimal solution. | **K3** |
| **CO3** | **Apply** and Solve transportation and assignment problem by using Linear programming Simplex method. | **K3** |
| **CO4** | Apply gradient and non-gradient methods to nonlinear optimization problems and use interior or exterior penalty functions for the constraints to derive the optimal solutions | **K3** |
| **CO5** | Formulate and apply Dynamic programming technique to inventory control, production planning, engineering design problems etc. to reach a final optimal solution from the current optimal solution. | **K5,K3** |

***#Based on suggested Revised BTL***

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 1 | 3 | 2 | | 2 | | | | | | | 2 |
| **CO2** | | 3 | 1 | 2 | | 2 | | | | | 2 | 2 |
| **CO3** | 1 | 2 | 2 | 2 | | | 2 | | | | | 3 |
| **CO4** | | 2 | 2 | | 3 | 2 | | | | | 3 | |
| **CO5** | 2 | 3 | 3 | 3 | 3 | 3 | 3 | | | | | 3 |

**(Please fill the above with Levels of Correlation, viz., L, M, H)**

**Department of Computer Science & Engineering**
**University College of Engineering, JNT University Kakinada**

**SYLLABUS:**
**UNIT – I: Introduction and Classical Optimization Techniques:** Statement of an Optimization problem – design vector – design constraints – constraint surface – objective function – objective function surfaces – classification of Optimization problems.
**Classical Optimization Techniques:** Single variable Optimization – multi variable Optimization without constraints – necessary and sufficient conditions for minimum/maximum – multivariable Optimization with equality constraints. Solution by method of Lagrange multipliers – multivariable Optimization with inequality constraints – Kuhn – Tucker conditions

**UNIT – II: Linear Programming :**Standard form of a linear programming problem – geometry of linear programming problems – definitions and theorems – solution of a system of linear simultaneous equations – pivotal reduction of a general system of equations – motivation to the simplex method – simplex algorithm – Duality in Linear Programming – Dual Simplex method.

**UNIT – III: Transportation Problem:** Finding initial basic feasible solution by north – west corner rule, least cost method and Vogel's approximation method – testing for optimality of balanced transportation problems – Special cases in transportation problem.
**UNIT – IV: Nonlinear Programming:** Unconstrained cases – One – dimensional minimization methods: Classification, Fibonacci method and Quadratic interpolation method – Univariate method, Powell's method and steepest descent method.
Constrained cases– Characteristics of a constrained problem, Classification, Basic approach of Penalty Function method; Basic approaches of Interior and Exterior penalty function methods, Introduction to convex Programming Problem.

**UNIT – V: Dynamic Programming:** Dynamic programming multistage decision processes – types – concept of sub optimization and the principle of optimality – computational procedure in dynamic programming – examples illustrating the calculus method of solution – examples illustrating the tabular method of solution.
**TEXT BOOKS:**

1. "Engineering optimization: Theory and practice"-by S. S.Rao, New Age International (P) Limited, 3rd edition, 1998.
2. "Introductory Operations Research" by H.S. Kasene & K.D. Kumar, Springer (India), Pvt. LTd.

**REFERENCE BOOKS:**

1. "Optimization Methods in Operations Research and systems Analysis" – by K.V. Mital and C. Mohan, New Age International (P) Limited, Publishers, 3rd edition, 1996.
2. Operations Research – by Dr. S.D.Sharma, Kedarnath, Ramnath & Co

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**III B. Tech I Sem-R20**

**ARTIFICIAL INTELLIGENCE**
**Code: R20PE3101**

**COURSE OBJECTIVES:** The student will be able to
- Know the methodology of Problem solving
- Implement basic AI algorithms
- Design and carry out an empirical evolution of different algorithms on a problem formalization

**COURSE OUTCOMES**: At the end of the course, student will be able to

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| CO1 | **Demonstrate** the fundamental concepts in Artificial Intelligence | K2 |
| CO2 | **Analyze** the applications of search strategies and problem reductions. | K6 |
| CO3 | **Apply** the mathematical logic concepts | K3 |
| CO4 | **Develop** the Logic Reasoning Systems in Artificial Intelligence. | K4 |
| CO5 | **Analyze** the Probabilistic Reasoning Systems | K5 |

***#Based on suggested Revised BTL***

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 1 | 3 | 2 | | 2 | | | | | | | 2 |
| CO2 | | 3 | 1 | 2 | | 2 | | | | | 2 | 2 |
| CO3 | 1 | 2 | 2 | 2 | | | 2 | | | | | 3 |
| CO4 | | 2 | 2 | | 3 | 2 | | | | | 3 | |
| CO5 | 2 | 3 | 3 | 3 | 3 | 3 | 3 | | | | | 3 |

**(Please fill the above with Levels of Correlation, viz., L, M, H)**

**SYLLABUS:**

**UNIT – I: Introduction:** Introduction to AI, Foundations of AI, History of AI, **Intelligent Agents:** Introduction, Structure of Intelligent Agents, Environments.

**UNIT – II: Problem Solving:** Problem-Solving Agents, Formulating Problems, Searching for Solutions, Generating action sequences, Data Structures for Search Trees, Search Strategies, **Informed Search Methods:** Best-First Search, Heuristic Functions, Memory Bounded Search, Iterative Improvement Algorithms.

**UNIT – III: Agents that Reason Logically:** A Knowledge-Based Agent, Wumpus World Environment, Representation, Reasoning, and Logic, Prepositional Logic: Syntax, Semantics, Validity and inference, Agent for the Wumpus World.

**First-Order Logic:** Syntax and Semantics, Extensions and Notational Variations, Using First-Order Logic, Logical Agents for the Wumpus World, A Simple Reflex Agent.

**UNIT – IV: Inference in First-Order Logic:** Inference Rules Involving Quantifiers, Generalized Modus Ponens, Forward and Backward Chaining algorithms, Completeness, A Complete Inference Procedure. **Logical Reasoning Systems:** Introduction, Indexing, Retrieval, and Unification, Logic Programming Systems, The Prolog language, Theorem Provers, Design of a theorem prover, Extending Prolog, Forward-Chaining Production Systems,

**UNIT – V: Probabilistic Reasoning Systems:** Probabilistic Reasoning Systems, The Semantics of Belief Networks, Inference in Belief Networks, Inference in Multiply Connected Belief Networks, Knowledge Engineering for Uncertain Reasoning. **Making Simple Decisions:** Combining Beliefs and Desires Under Uncertainty, Utility Functions, Multi attribute utility functions, Decision Networks, Markov Decision Processes.

**TEXT BOOK:**

1. Artificial Intelligence – A Modern Approach, Stuart J. Russell and Peter Norvig, Prentice Hall, 1995.
2. AI and Machine Learning for Coders: A Programmer's Guide to Artificial Intelligence — Laurence Moroney

**REFERENCE BOOKS:**

1. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, 2nd Edition, O'Reilly Publications, 2019

**Web Link:**

1. Swayam NPTEL: An Introduction to Artificial Intelligence: **https://onlinecourses.nptel.ac.in/noc22_cs56/preview**

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**III B. Tech I Sem-R20**

### SOFTWARE PROJECT MANAGEMENT
**Code: R20PE3101**

**Pre-requisites:** Software Engineering

**COURSE OBJECTIVES:**

The main objectives of the course is to make the student

- understand the basic concepts and issues of software project management
- plan and manage projects at each stage of the software development life cycle (SDLC) and to complete a specific project in time with the available budget

**COURSE OUTCOMES:**

After finishing this course student will be able to:

| CO | Course Outcomes | Knowlede Level (K)# |
|---|---|---|
| CO1 | Identify the Problem Effectively & Efficiently with proper documentation for the use in different software teams and organization and also Outline various artifacts sets for better understanding of software development | K2 |
| CO2 | effectively Plan the software projects that address real-world management challenges and Design various estimation levels of cost and effort | K6 |
| CO3 | Adapt and apply Project Scheduling, tracking, Risk analysis, Quality management and Project Cost estimation using different techniques | K3 |
| CO4 | Organize activities necessary to successfully complete and close the Software projects | K4 |
| CO5 | Comprehend and be able to plan the activities within time schedules with CPM and PERT Analysis and Practice the role of professional ethics in successful software development | K3,K5 |

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 2 | 3 | 2 | | 2 | | | | | | | 2 |
| CO2 | 3 | 3 | 1 | 2 | | 2 | | | | | 2 | 2 |
| CO3 | 3 | 2 | 2 | 2 | | | 2 | | | | | 3 |
| CO4 | 1 | 2 | 2 | | 3 | 2 | | | | | 3 | |
| CO5 | 3 | 3 | 3 | 3 | 3 | 3 | 3 | | | | | 3 |

**(Please fill the above with Levels of Correlation, viz., L-1, M-2, H-3)**

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**SYLLABUS:**

**UNIT-I: Conventional Software Management:** The waterfall model, conventional software Management performance. **Evolution of Software Economics:** Software Economics, pragmatic software cost estimation. **Improving Software Economics:** Reducing Software product size, improving software processes, improving team effectiveness, improving automation, Achieving required quality, peer inspections.

**UNIT-II: The Old Way and The New:** The principles of conventional software Engineering, principles of modern software management, transitioning to an iterative process. **Life Cycle Phases:** Engineering and production stages, inception, elaboration, construction, transition phases. **Artifacts of The Process:** The artifact sets, Management artifacts, Engineering artifacts, programmatic artifacts.

**UNIT- III: Model Based Software Architectures:** A Management perspective and technical perspective. **Work Flows of The Process:** Software process workflows, Iteration workflows. **Checkpoints of the Process:** Major mile stones, Minor Milestones, Periodic status assessments.

**UNIT- IV: Iterative Process Planning:** Work breakdown structures, planning guidelines, cost and schedule estimating, Iteration planning process, Pragmatic planning. **Project Organizations and Responsibilities:** Line-of-Business Organizations, Project Organizations, evolution of Organizations.

**UNIT-V: Process Automation:** Automation Building blocks, The Project Environment. **Project Control and Process Instrumentation:** The seven core Metrics, Management indicators, quality indicators, life cycle expectations, pragmatic Software Metrics, Metrics automation. **Project Estimation and Management:** COCOMO model, Critical Path Analysis, PERT technique, Monte Carlo approach

**TEXT BOOKS:**

1. Software Project Management, Walker Royce, PEA, 2006.
2. Software Project Management, Bob Hughes,3/e, Mike Cotterell, TMH

**REFERENCE BOOKS:**

1. Software Project Management, Joel Henry, PEA, 2006
2. Software Project Management in practice, Pankaj Jalote, PEA, 2005,
3. Effective Software Project Management, Robert K.Wysocki, Wiley,2006
4. Project Management in IT, Kathy Schwalbe, Cengage
5. Quality Software Project Management, Futrell, Donald F. Shafer, Donald I. Shafer, PEA

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**III B. Tech I Sem-R20**

**DISTRIBUTED SYSTEMS**
**Code: R20PE3101**

**COURSE OBJECTIVES:** From the course the student will learn

- To learn the principles, architectures, algorithms and programming models used in distributed systems.
- To examine state-of-the-art distributed systems, such as Google File System.
- To design and implement sample distributed systems.
- The structure of distributed systems using multiple levels of software is emphasized.
- To design of distributed systems and algorithms that support distributed computing.

**COURSE OUTCOMES:**

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| CO1 | Explain the Fundamentals of Distributed System. | K2 |
| CO2 | Determine Concepts and access System Basic Distributed System Architecture. | K4 |
| CO3 | Examine Communication and Operating System Architecture. | K3 |
| CO4 | Discuss in detail the system level and support required for distributed system. | K5 |
| CO5 | Discuss the issues involved in transactions and Recoveries. | K3 |

***#Based on suggested Revised BTL***

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 1 | 1 | 2 | 2 | | 2 | | | | | | |
| CO2 | 1 | 1 | 2 | | 2 | | | | | 1 | | |
| CO3 | | | 1 | 2 | 3 | | | | | 3 | | 2 |
| CO4 | 1 | 1 | 2 | | 1 | | | | | | 2 | |
| CO5 | 1 | 2 | 3 | 2 | | | | | | | | |

**(Please fill the above with Levels of Correlation, viz., L-1, M-2, H-3)**

**Department of Computer Science & Engineering**
**University College of Engineering, JNT University Kakinada**

**SYLLABUS:**

**UNIT-I: Characterization of Distributed Systems:** Introduction, Examples of Distributed Systems, Challenges. **System Models:** Introduction, Physical Models, Architectural Models, Fundamental Models

**UNIT-II: Interprocess Communication:** Introduction, The API for the Internet Protocols, External Data Representation and Marshalling, Multicast Communication, Network Virtualization: Overlay Networks **Remote Invocation:** Introduction, Request-reply Protocols, Remote Procedure Call, Remote Method Invocation

**UNIT-III: Indirect Communication:** Introduction, Group Communication, Message Queues, Shared Memory Approaches **Operating System Support:** Introduction, The Operating System Layer, Protection, Processes and Threads, Communication and Invocation, Operating System Architecture, Virtualization at the Operating System Level.

**UNIT-IV: Distributed File Systems:** Introduction, File Service Architecture, Case Study: Sun Network File System, Enhancements and Further Developments. **Coordination and Agreement:** Introduction, Distributed Mutual Exclusion, Elections, Coordination and Agreement in Group Communication

**UNIT-V: Transactions and Concurrency Control:** Introduction, Transactions, Nested Transactions, Locks **Distributed Transactions:** Introduction, Flat and Nested Distributed Transactions, Atomic Commit Protocols, Concurrency Control in Distributed Transactions, Distributed Dead Locks, Transaction Recovery

**TEXT BOOKS:**

1. George Coulouris, Jean Dollimore, Tim Kindberg, Gordon Blair "Distributed Systems- Concepts and Design", Fifth Edition, Pearson Publications

**REFERENCE BOOKS:**

1. Ajay D Kshemkalyani, MukeshSinghal, "Distributed Computing-Principles, Algorithms and Systems", Cambridge
2. Andrew S Tanenbaum, Maarten Van Steen, "Distributed Systems-Principles and Paradigms-PHI

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**III B. Tech I Sem-R20**

**ADVANCED UNIX PROGRAMMING**
**Code: R20PE3101**

**COURSE OBJECTIVES:**
Understating the shell commands, shell programming, system calls of files and processes, signals, inter-process communication concepts and programming, TCP and UDP.

**COURSE OUTCOMES:** After finishing this course student will be able to:

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Gain good knowledge on Unix commands AND Awareness of shell programming | **K2** |
| **CO2** | Know about different system calls for files and directories | **K6** |
| **CO3** | Ability to know the working of processes and signals | **K3** |
| **CO4** | Application of client server program for IPC | **K4** |
| **CO5** | Knowledge about socket programming | **K3,K5** |

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 3 | 2 | | 2 | | | | | | | 2 | |
| **CO2** | 3 | 1 | 2 | | 2 | | | | | 2 | 2 | |
| **CO3** | 2 | 2 | 2 | | | 2 | | | | | 3 | |
| **CO4** | 2 | 2 | | 3 | 2 | | | | | 3 | | |
| **CO5** | 3 | 3 | 3 | 3 | 3 | 3 | | | | | 3 | |

**(Please fill the above with Levels of Correlation, viz., L-1, M-2, H-3)**

Department of Computer Science & Engineering
University College of Engineering, JNT University Kakinada

**SYLLABUS:**

**UNIT-1:** Introduction, Architecture of unix, Responsibilities of shell, unix file system, vi editor. **Unix commands**: Some Basic Commands, file utilities, process utilities, text processing utilities, network utilities, disk utilities, backup utilities , Security by file permissions.

**UNIT-2: Shell Programming**: shell variables, The Export command, The Profile File a Script Run During starting, The First Shell Script, The read command, Positional Parameters, The $? Variable , Knowing the exit Status- More about the Set Command, The Exit command, Branching Control Structures, Loop Control Structures, The Continue and Break Statement- The Expr Command, Performing Integer Arithmetic- Real Arithmetic in Shell Programs- The here Document(<<), The Sleep Command, Debugging Scripts, The Script command, The Eval command, The Exec Command, Sample programs. **Files** - Introduction, file descriptors, 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, links-soft and hard links-symlink, link, unlink.

**UNIT-3: Directories-**Creating, removing and changing Directories-mkdir, rmdir, chdir, obtaining current working directory-getcwd, Directory contents, Scanning Directories-opendir, readdir, closedir, rewinddir functions. **Process Control**: process identifiers, fork function, vfork function, exit function, wait and waitpid functions, exec functions, user identification. **Signals**: signal handling using signal function, kill and raise, alarm, pause, abort and sleep functions.

**UNIT-4: IPC**: introduction, pipes, FIFO's, client –server examples for pipes and FIFO's **message queues:** message queue structure in kernel, system calls of message queue, client-server example for message queue. **Semaphores:** definition, system calls of semaphores, semaphores structure in kernel, file locking using semaphores

**UNIT-5: Shared memory**-system calls of shared memory, semaphore structure in kernel, client server example. **Sockets:** Introduction, overview, elementary socket system calls, TCP Echo program, UDP Echo program

**TEXT BOOKS:**

1. Unix the ultimate guide, 3rd edition, sumitabha Das, TMH.
2. Advanced programming in the unix environment by W. Richard Stevens.
3. Unix network programming by W. Richard Stevens.

**REFERENCE BOOKS:**

1. Introduction to Unix and shell programming, Venkateshmurhty
2. Unix and shell programming by B.M. Harwani, OXFORD university press.

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**III B. Tech I Sem-R20**

**Data Mining Techniques with R Lab**
**Code: R20PC3104**

**Pre-requisites:** Data Base Management Systems, Python Programming
**COURSE OBJECTIVES:** The main objective of the course is to

- Inculcate Conceptual, Logical, and Physical design of Data Warehouses OLAP applications and OLAP deployment
- Design a data warehouse or data mart to present information needed by management in a form that is usable
- Emphasize hands-on experience working with all real data sets.
- Test real data sets using popular data mining tools such as WEKA, Python Libraries
- Develop ability to design various algorithms based on data mining tools.

**COURSE OUTCOMES:** By the end of the course student will be able to

| CO | Course Outcomes | Knowledge Level |
|---|---|---|
| CO1 | Design a data mart or data warehouse for any organization | K6 |
| CO2 | Extract knowledge using data mining techniques and enlist various algorithms used in information analysis of Data Mining Techniques | K2 |
| CO3 | Demonstrate the working of algorithms for data mining tasks such as association rule mining, classification for realistic data | K3 |
| CO4 | Implement and Analyze on knowledge flow application on data sets and Apply the suitable visualization techniques to output analytical results | K3 |

**Mapping of course outcomes with program outcomes**

| | PO 1 | PO2 | PO 3 | PO 4 | PO 5 | PO 6 | PO 7 | PO 8 | PO 9 | PO 10 | PO 11 | PO 12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 1 | 2 | 3 | 3 | 3 | | | | | | 2 | 2 |
| CO2 | 1 | 2 | 1 | 2 | 3 | | | | | | 1 | 2 |
| CO3 | 1 | 2 | 3 | 2 | 3 | | | | | | 2 | 1 |
| CO4 | 2 | 2 | 3 | 3 | 3 | | | | | | 1 | 1 |

**(Please fill the above with levels of Correlation, viz., 1, 2, 3)**

**Software Requirements:** WEKA Tool/Python/R-Tool/Rapid Tool/Oracle Data mining

**List of Experiments:**

1. Creation of a Data Warehouse.
   - Build Data Warehouse/Data Mart (using open source tools like Pentaho Data Integration Tool, Pentaho Business Analytics; or other data warehouse tools like Microsoft-SSIS, Informatica, Business Objects,etc.,)

- Design multi-dimensional data models namely Star, Snowflake and Fact Constellation schemas for any one enterprise (ex. Banking, Insurance, Finance, Healthcare, manufacturing, Automobiles, sales etc).
- Write ETL scripts and implement using data warehouse tools.
- Perform Various OLAP operations such slice, dice, roll up, drill up and pivot

2. Explore machine learning tool "WEKA"
   - Explore WEKA Data Mining/Machine Learning Toolkit.
   - Downloading and/or installation of WEKA data mining toolkit.
   - Understand the features of WEKA toolkit such as Explorer, Knowledge Flow interface, Experimenter, command-line interface.
   - Navigate the options available in the WEKA (ex. Select attributes panel, Preprocess panel, Classify panel, Cluster panel, Associate panel and Visualize panel)
   - Study the arff file format Explore the available data sets in WEKA. Load a data set (ex. Weather dataset, Iris dataset, etc.)
   - Load each dataset and observe the following:
     1. List the attribute names and they types
     2. Number of records in each dataset
     3. Identify the class attribute (if any)
     4. Plot Histogram
     5. Determine the number of records for each class.
     6. Visualize the data in various dimensions

3. Perform data preprocessing tasks and Demonstrate performing association rule mining on data sets
   - Explore various options available in Weka for preprocessing data and apply Unsupervised filters like Discretization, Resample filter, etc. on each dataset
   - Load weather. nominal, Iris, Glass datasets into Weka and run Apriori Algorithm with different support and confidence values.
   - Study the rules generated. Apply different discretization filters on numerical attributes and run the Apriori association rule algorithm. Study the rules generated.
   - Derive interesting insights and observe the effect of discretization in the rule generation process.
4. Demonstrate performing classification on data sets Weka/R

   - Load each dataset and run 1d3, J48 classification algorithm. Study the classifier output. Compute entropy values, Kappa statistic.
   - Extract if-then rules from the decision tree generated by the classifier, Observe the confusion matrix.
   - Load each dataset into Weka/R and perform Naïve-bayes classification and k-Nearest Neighbour classification. Interpret the results obtained.
   - Plot RoC Curves
   - Compare classification results of ID3, J48, Naïve-Bayes and k-NN classifiers for each dataset, and deduce which classifier is performing best and poor for each dataset and justify.

5. Demonstrate performing clustering of data sets
   - Load each dataset into Weka/R and run simple k-means clustering algorithm with different values of k (number of desired clusters).
   - Study the clusters formed. Observe the sum of squared errors and centroids, and derive insights.
   - Explore other clustering techniques available in Weka/R.
   - Explore visualization features of Weka/R to visualize the clusters. Derive interesting insights and explain.
6. Demonstrate knowledge flow application on data sets into Weka/R
   - Develop a knowledge flow layout for finding strong association rules by using Apriori, FP Growth algorithms
   - Set up the knowledge flow to load an ARFF (batch mode) and perform a cross validation using J48 algorithm
   - Demonstrate plotting multiple ROC curves in the same plot window by using j48 and Random forest tree
7. Demonstrate ZeroR technique on Iris dataset (by using necessary preprocessing technique(s)) and share your observations
8. Write a java program to prepare a simulated data set with unique instances.
9. Write a Python program to generate frequent item sets / association rules using Apriori algorithm
10. Write a program to calculate chi-square value using Python/R. Report your observation.
11. Write a program of Naive Bayesian classification using Python/R programming language.
12. Implement a Java/R program to perform Apriori algorithm
13. Write a R program to cluster your choice of data using simple k-means algorithm using JDK
14. Write a program of cluster analysis using simple k-means algorithm Python/R programming language.
15. Write a program to compute/display dissimilarity matrix (for your own dataset containing at least four instances with two attributes) using Python
16. Visualize the datasets using matplotlib in python/R.(Histogram, Box plot, Bar chart, Pie chart etc.,)

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**III B. Tech I Sem-R20**

**COMPILER DESIGN LAB**
**Code: R20PC3105**

**COURSE OUTCOMES:** The end of the course student will be able to

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Design simple lexical analyzers | **K6** |
| **CO2** | Determine predictive parsing table for a CFG | **K5** |
| **CO3** | Apply Lex and Yacc tools | **K3** |
| **CO4** | Examine LR parser and generating SLR Parsing table | **K4** |
| **CO5** | Relate Intermediate code generation for subset C language | **K2** |

**# Based on suggested Revised BTL**

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 3 | 2 | | 2 | | | | | | | 2 | |
| **CO2** | 3 | 1 | 2 | | 2 | | | | | 2 | 2 | |
| **CO3** | 2 | 2 | 2 | | | 2 | | | | | 3 | |
| **CO4** | 2 | 2 | | 3 | 2 | | | | | 3 | | |
| **CO5** | 3 | 3 | 3 | 3 | 3 | 3 | | | | | 3 | |

**(Please fill the above with Levels of Correlation, viz., L-1, M-2, H-3)**

**List of Experiments:**

**Experiment 1:** Write a C program to identify different types of Tokens in a given Program.

**Experiment 2:** Write a Lex Program to implement a Lexical Analyzer using Lex tool.

**Experiment 3:** Write a C program to Simulate Lexical Analyzer to validating a given input String.

**Experiment 4:** Write a C program to implement the Brute force technique of Top down Parsing.

**Experiment 5:** Write a C program to implement a Recursive Descent Parser.

**Experiment 6:** Write C program to compute the *First* and *Follow* Sets for the given Grammar.

**Experiment 7:** Write a C program for eliminating the left recursion and left factoring of a given grammar

**Experiment 8:** Write a C program to check the validity of input string using Predictive Parser.

**Experiment 9:** Write a C program for implementation of LR parsing algorithm to accept a given input string.

**Experiment10:** Write a C program for implementation of a Shift Reduce Parser using Stack Data Structure to accept a given input string of a given grammar.

**Experiment 11:** Simulate the calculator using LEX and YACC tool.

**Experiment 12:** Generate YACC specification for a few syntactic categories.

**Experiment 13:** Write a C program for generating the three address code of a given expression/statement.

**Experiment 14:** Write a C program for implementation of a Code Generation Algorithm of a given expression/statement.

**Text Books / Reference Books / Online Resources:**

1. Compilers: Principles, Techniques and Tools, Second Edition, Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffry D. Ullman, Pearson Publishers, 2007.
2. John R Levine, Tony Mason, Doug Brown, "Lex and Yacc", Orielly, 2nd Edition, 2009.

**Department of Computer Science & Engineering**
**University College of Engineering, JNT University Kakinada**

**III B. Tech I Sem-R20**

**ANIMATION DESIGN**
**Code: R20SO3101**

**COURSE OUTCOMES:**

- Design layouts for web pages, Paper Adverts, Broachers, CD Covers and package designing
- Event and Exhibition stall Designs,
- Color corrections

**Software Requirements: Adobe CorelDraw & Illustrator**

**List of Experiments:**

**Experiment 1:**

1. Design a logo, brochure, cover letter, visiting cards.
2. Prepare a cutout of some images using Photoshop.
3. Place nice background for those images.

**Experiment 2:**

1. Prepare nice background using gradient tool.
2. Creating Web Banners in Adobe Flash
3. Creating a Logo Animation in Adobe Flash

**Experiment 3:**

1. Draw Cartoon Animation using reference.
2. Create Lip Sink to Characters
3. Using filters & Special effects

**Experiment 4:**

1. Create a scene by using Mask layers animation
2. Create any Model of Cars or Bike,
3. Create any model of the male or female character.

**Experiment 5:**

1. Create a natural outdoor or indoor scene.
2. Render a frame and video of indoor and outdoor scenes.
3. Advance lighting using mental ray render.

**Experiment 6:**

1. Animate day and night scene of a street with the help of lighting.
2. Title Graphics
3. Video – Audio synchronization
4. 30 Second Commercial AD

**Experiment 7:**

1. Creating an effect of snow or rain or smoke or water.
2. Creating an effect of bomb/explosion.

**Experiment 8:**

1. Create a natural outdoor or indoor scene.
2. Set light for Day, Night and Morning
3. Render a frame and video of indoor and outdoor scenes.

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**III B. Tech I Sem-R20**

## Employability Skills-I
**Code: R20MC3101**

**COURSE OUTCOMES:** The end of the course student will be able to

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| CO1 | Understand the corporate etiquette. | K2 |
| CO2 | Make presentations effectively with appropriate body language | K3 |
| CO3 | Be composed with positive attitude | K3 |
| CO4 | Understand the core competencies to succeed in professional and personal life | K2 |

**# Based on suggested Revised BTL**

**UNIT – I:**
**1. Analytical Thinking & Listening Skills**: Self-Introduction, Shaping Young Minds - A Talk by Azim Premji (Listening Activity), Self – Analysis, Developing Positive Attitude, Perception.
**2.Communication Skills**: Verbal Communication; Non Verbal Communication (Body Language)

**UNIT – II:**
**3.Self-Management Skills:** Anger Management, Stress Management, Time Management, Six Thinking Hats, Team Building, Leadership Qualities
**4. Etiquette:** Social Etiquette, Business Etiquette, Telephone Etiquette, Dining Etiquette

**UNIT – III:**
**5. Standard Operation Methods:** Note Making, Note Taking, Minutes Preparation, Email & Letter Writing
**6 Verbal Ability**: Synonyms, Antonyms, One Word Substitutes-Correction of Sentences-Analogies, Spotting Errors, Sentence Completion, Course of Action - Sentences Assumptions, Sentence Arguments, Reading Comprehension, Practice work

**UNIT-IV:**
**7. Job-Oriented Skills –I:** Group Discussion, Mock Group Discussions
**8. Job-Oriented Skills –II**: Resume Preparation, Interview Skills, Mock Interviews

**Text books and Reference books:**

1. Barun K. Mitra, Personality Development and Soft Skills, Oxford University Press, 2011.
2. S.P. Dhanavel, English and Soft Skills, Orient Blackswan, 2010.
3. R.S.Aggarwal, A Modern Approach to Verbal & Non-Verbal Reasoning, S.Chand & Company Ltd., 2018.
4. Raman, Meenakshi & Sharma, Sangeeta, Technical Communication Principles and Practice, Oxford University Press, 2011.

**E-resources:**

1. www. Indiabix.com
2. www.freshersworld.com

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**III B. Tech II Sem-R20**

**MACHINE LEARNING**
**Code: R20PC3201**

**COURSE OBJECTIVES:**

- Identify problems that are amenable to solution by ANN methods, and which ML methods may be suited to solving a given problem.
- Formalize a given problem in the language/framework of different ANN methods (e.g., as a search problem, as a constraint satisfaction problem, as a planning problem, as a Markov decision process, etc).

**COURSE OUTCOMES:** After the completion of the course, student will be able to

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Explain the fundamental usage of the concept Machine Learning system | **K2** |
| **CO2** | Demonstrate on various regression Technique | **K2** |
| **CO3** | Analyze the Ensemble Learning Methods | **K4** |
| **CO4** | Illustrate the Clustering Techniques and Dimensionality Reduction Models in Machine Learning. | **K5** |
| **CO5** | Discuss the Neural Network Models and Fundamentals concepts of Deep Learning | **K6** |

**#based on suggested Revised BTL**

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 2 | 3 | 2 | 1 | 2 | | | | 1 | | 1 | 2 |
| **CO2** | 1 | 2 | 3 | 2 | 1 | 1 | 1 | | | | 1 | 2 |
| **CO3** | 2 | 3 | 2 | 2 | 3 | 1 | | | | | 1 | 3 |
| **CO4** | 1 | 3 | 2 | 2 | 2 | 2 | | | | | 2 | 2 |
| **CO5** | 2 | 2 | 3 | 3 | 1 | 1 | 1 | | | | 2 | 2 |

**(Please fill the above with Levels of Correlation, viz., L, M, H)**

**Department of Computer Science & Engineering**
**University College of Engineering, JNT University Kakinada**

**SYLLABUS:**

**Unit-I: Introduction-** Artificial Intelligence, Machine Learning, Deep learning, Types of Machine Learning Systems, Main Challenges of Machine Learning.
**Statistical Learning**: Introduction, Supervised and Unsupervised Learning, Training and Test Loss, Tradeoffs in Statistical Learning, Estimating Risk Statistics, Sampling distribution of an estimator, Empirical Risk Minimization.

**Unit-II: Supervised Learning**(Regression/Classification):Basic Methods: Distance based Methods, Nearest Neighbours, Decision Trees, Naive Bayes, **Linear Models:** Linear Regression, Logistic Regression, Generalized Linear Models, Support Vector Machines, **Binary Classification:** Multiclass/Structured outputs, MNIST, Ranking.

**Unit-III: Ensemble Learning and Random Forests:** Introduction, Voting Classifiers, Bagging and Pasting, Random Forests, Boosting, Stacking.
**Support Vector Machine:** Linear SVM Classification, Nonlinear SVM Classification SVM Regression, Naïve Bayes Classifiers.
.
**Unit-IV: Unsupervised Learning Techniques:** Clustering, K-Means, Limits of K-Means, Using Clustering for Image Segmentation, Using Clustering for Preprocessing, Using Clustering for Semi-Supervised Learning, DBSCAN, Gaussian Mixtures.
Dimensionality Reduction: The Curse of Dimensionality, Main Approaches for Dimensionality Reduction, PCA, Using Scikit-Learn, Randomized PCA ,Kernel PCA.

**Unit-V: Neural Networks and Deep Learning:** Introduction to Artificial Neural Networks with Keras, Implementing MLPs with Keras, Installing TensorFlow 2, Loading and Preprocessing Data with TensorFlow.

**TEXT BOOKS:**

1. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, 2nd Edition, O'Reilly Publications, 2019
2. Data Science and Machine Learning Mathematical and Statistical Methods,Dirk P. Kroese, Zdravko I. Botev, Thomas Taimre, Radislav Vaisman,25th November 2020
3. Machine Learning Probabilistic Approach, Kevin P. Murphy, MIT Press, 2012.

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**III B. Tech II Sem-R20**

**BIG DATA ANALYTICS**
**Code: R20PC3202**

**COURSE OBJECTIVES:**

- Optimize business decisions and create competitive advantage with Big Data analytics
- Introducing Java concepts required for developing map reduce programs
- Derive business benefit from unstructured data
- Imparting the architectural concepts of Hadoop and introducing map reduce paradigm
- To introduce programming tools PIG & HIVE in Hadoop echo system.

**COURSE OUTCOMES:** At the end of the course, student will be able to

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | **Preparing** for data summarization, query, and analysis. | **K2** |
| **CO2** | **Applying** data modelling techniques to large data sets | **K3** |
| **CO3** | **Creating** applications for Big Data analytics. | **K3** |
| **CO4** | **Building** a complete business data analytic solution | **K4** |
| **CO5** | **Demonstrate and Develop** the pig and Latin Scripts | **K2** |

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 2 | 3 | 2 | 1 | 2 | | | | 1 | | 1 | 2 |
| **CO2** | 1 | 2 | 3 | 2 | 1 | 1 | 1 | | | | 1 | 2 |
| **CO3** | 2 | 3 | 2 | 2 | 3 | 1 | | | | | 1 | 3 |
| **CO4** | 1 | 3 | 2 | 2 | 2 | 2 | | | | | 2 | 2 |
| **CO5** | 2 | 2 | 3 | 3 | 1 | 1 | 1 | | | | 2 | 2 |

**(Please fill the above with Levels of Correlation, viz., L-1, M-2, H-3)**

**SYLLABUS:**

**UNIT-I: Data structures in Java:** Linked List, Stacks, Queues, Sets, Maps; Generics: Generic classes and Type parameters, Implementing Generic Types, Generic Methods, Wrapper Classes, Concept of Serialization

**UNIT-II: Working with Big Data:** Google File System, Hadoop Distributed File System (HDFS) Building blocks of Hadoop (Namenode, Datanode, Secondary Namenode, Job Tracker, Task Tracker), Introducing and Configuring Hadoop cluster (Local, Pseudo-distributed mode, Fully Distributed mode), Configuring XML files.

**UNIT-III: Writing Map Reduce Programs:** A Weather Dataset, Understanding Hadoop API for Map Reduce Framework (Old and New), Basic programs of Hadoop Map Reduce: Driver code, Mapper code, Reducer code, Record Reader, Combiner, Practitioner

**UNIT-IV: StreamMemory and Spark:** Introduction to Streams Concepts– Stream Data Model and Architecture , Stream computing, Sampling Data in a Stream , Filtering Streams ,Counting Distinct Elements in a Stream , Introduction to Spark Concept , Spark Architecture and components , Spark installation , Spark RDD(Resilient Distributed Dataset) – Spark RDD operations.

**UNIT-V**: **Pig:** Hadoop Programming Made Easier Admiring the Pig Architecture, Going with the Pig Latin Application Flow, Working through the ABCs of Pig Latin, Evaluating Local and Distributed Modes of Running Pig Scripts, Checking out the Pig Script Interfaces, Scripting with Pig Latin.
**Applying Structure to Hadoop Data with Hive:** Saying Hello to Hive, Seeing How the Hive is Put Together, Getting Started with Apache Hive, Examining the Hive Clients, Working with Hive Data Types, Creating and Managing Databases and Tables, Seeing How the Hive Data Manipulation Language Works, Querying and Analysing data

**TEXT BOOKS:**
1. Wiley & Big Java 4th Edition, Cay Horstmann, Wiley John Sons, INC
2. Hadoop: The Definitive Guide by Tom White, 3rd. Edition, O'reilly

**REFERENCE BOOKS:**
1. Hadoop in Action by Chuck Lam, MANNING Publ.
2. Hadoop for Dummies by Dirk deRoos, Paul C.Zikopoulos, Roman B.Melnyk,Bruce Brown, Rafael Coss
3. Hadoop in Practice by Alex Holmes, MANNING Publ.
4. Big Data Analytics by Dr. A.Krishna Mohan and Dr.E.Laxmi Lydia
5. Hadoop Map Reduce Cookbook, SrinathPerera, ThilinaGunarathne

**Software Links:**
1. Hadoop:http://hadoop.apache.org/
2. Hive: https://cwiki.apache.org/confluence/display/Hive/Home
3. Piglatin: http://pig.apache.org/docs/r0.7.0/tutorial.html

**Department of Computer Science & Engineering**
**University College of Engineering, JNT University Kakinada**

**III B. Tech II Sem-R20**

**CRYPTOGRAPHY & NETWORK SECURITY**
**Code: R20PC3203**

**COURSE OBJECTIVES:**

- Explain the objectives of information security
- Explain the importance and application of each of confidentiality, integrity, authentication and availability
- Understand the basic categories of threats to computers and networks
- Discusses the Mathematics of Cryptography
- Discuss the fundamental ideas of Symmetric and Asymmetric cryptographic Algorithms
- Discusses the Network layer, Transport Layer and Application layer Protocols Enhanced security mechanisms

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

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Student will be able to understand security issues related to computer networks and learn different symmetric key techniques | **K2** |
| **CO2** | Students will be able learn mathematic of cryptography for symmetric and Asymmetric algorithms and apply this knowledge to understand the Cryptographic algorithms | **K3** |
| **CO3** | Students will be able learn different types of symmetric and Asymmetric algorithms | **K3** |
| **CO4** | Students will be able learn different algorithms of Hash functions, message authentication and digital signature and their importance to the security | **K4** |
| **CO5** | Students will be able learn different Enhanced security protocols of Application Layer, Transport Layer and Network layer | **K4** |

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 1 | 2 | 1 | 2 | 1 | 1 | 1 | 1 | | 1 | | |
| CO2 | 3 | 1 | 1 | 2 | 2 | 2 | 1 | 2 | | 3 | 3 | |
| CO3 | 2 | 2 | 2 | 1 | 2 | 1 | 1 | 1 | | 2 | | |
| CO4 | 3 | 2 | 3 | 2 | 3 | 2 | 1 | 1 | | 2 | 1 | |
| CO5 | 3 | 2 | 3 | 1 | 2 | 2 | 1 | 1 | | 2 | 2 | |

**(Please fill the above with Levels of Correlation, viz., L-1, M-2, H-3)**

**SYLLABUS:**

**UNIT – I: Security Concepts:** Introduction, The need for security, Security approaches, Principles of security, Types of Security attacks, Security services, Security Mechanisms, A model for Network Security Cryptography. Classical Encryption Techniques-symmetric cipher model, Substitution techniques, Transposition techniques, Rotor Machines, Stegnography.

**Introduction to Mathematics of Cryptography:** Integer Arithmetic, Euclidean Algorithm, The Extended Euclidean Algorithm, Linear Diophantine Equations, Modular Arithmetic, Inverses, Matrices, Linear Congruence.

**UNIT – II: Mathematics of Symmetric Cryptography:** Algebraic Structures-Groups, Rings, Fields, GF($2^n$) fields, Polynomials. **Mathematics of Asymmetric cryptography:** Primes, Checking For Primness, Eulers phi-functions, Fermat's Little Theorem, Euler's Theorem, Generating Primes, Primality Testing, Factorization, Chinese Remainder Theorem, Quadratic Congruence, Exponentiation And Logarithm.

**UNIT – III: Symmetric key Ciphers:** Block Cipher principles, DES, AES, Blowfish, IDEA, Block cipher operation, Stream ciphers: RC4, RC5

**Asymmetric key Ciphers:** Principles of public key cryptosystems, RSA algorithm, Diffie-Hellman Key Exchange, Elgamal Cryptographic system, Elliptic Curve Arithmetic, Elliptic Curve Cryptography.

**UNIT – IV: Cryptographic Hash Functions:** Applications of Cryptographic Hash Functions, Two Simple Hash Functions, Requirements and Security, Hash Functions Based on Cipher Block Chaining, Secure Hash Algorithms (SHA)

**Message Authentication Codes:** Message Authentication Requirements, Message Authentication Functions, Requirements for Message Authentication Codes, Security of MAC'S,MAC'S Based On Hash Functions: HMAC, MAC'S Based On Block Ciphers: DAA And CMAC

**Digital Signatures:** Digital Signatures, Elgamal Digital Signature Scheme, Elliptic Curve Digital Signature Algorithm, RSA-PSS Digital Signature Algorithm.

**UNIT – V: Network and Internet Security: Transport-Level Security:** Web Security Considerations, Transport Level Security, HTTPS, SSH.

**IP Security:** IP Security Overview, IP Security Policy, Encapsulating Security Payload, Authentication Header Protocol.

**Electronic-Mail Security:** Internet-mail Security, Email Format, Email Threats and Comprehensive Email Security, S/MIME, PGP.

**TEXT BOOKS:**

1. Cryptography and Network Security - Principles and Practice: William Stallings, Pearson Education, 7th Edition, 2017
2. Cryptography and Network Security: Behrouz A. Forouzan Debdeep, Mc Graw Hill, 3rd Edition, 2015

**REFERENCE BOOKS:**

1. Cryptography and Network Security: Atul Kahate, Mc Graw Hill, 3rd Edition
2. Introduction to Cryptography with Coding Theory: Wade Trappe, Lawrence C. Washington, Pearson.
3. Modern Cryptography: Theory and Practice By Wenbo Mao. Pearson

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**III B. Tech II Sem-R20**

**MOBILE COMPUTING**
**Code: R20PE3201**

**COURSE OBJECTIVES:**

- To understand the fundamentals of mobile communication
- To understand the architecture of various Wireless Communication Networks
- To understand the significance of different layers in mobile system Course Contents

**COURSE OUTCOMES:** Upon completion of the course, the students will be able to:

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Develop a strong grounding in the fundamentals of mobile Networks | **K2** |
| **CO2** | Apply knowledge in MAC, Network, and Transport Layer protocols of Wireless Network | **K3** |
| **CO3** | Comprehend, design, and develop a lightweight network stack | **K3** |
| **CO4** | Analyze the Mobile Network Layer system working | **K4** |
| **CO5** | Explain about the WAP Model | **K5** |

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 3 | 2 | | 2 | | | | | | | 2 | |
| **CO2** | 3 | 1 | 2 | | 2 | | | | | 2 | 2 | |
| **CO3** | 2 | 2 | 2 | | | 2 | | | | | 3 | |
| **CO4** | 2 | 2 | | 3 | 2 | | | | | 3 | | |
| **CO5** | 3 | 3 | 3 | 3 | 3 | 3 | | | | | 3 | |

**(Please fill the above with Levels of Correlation, viz., L-1, M-2, H-3)**

**SYLLABUS:**

**UNIT- I: Introduction to Wireless Networks** - Applications - History - Simplified Reference Model - Wireless transmission - Frequencies - Signals - Antennas - Signal propagation - Multiplexing - Modulation - Spread spectrum - Cellular Systems: Frequency Management and Channel Assignment - types of hand-off and their characteristics.

**UNIT- II: MAC** - Motivation - SDMA - FDMA - TDMA - CDMA - Telecommunication Systems - GSM: Architecture Location tracking and call setup - Mobility management - Handover - Security - GSM - SMS –

International roaming for GSM - call recording functions - subscriber and service data management - DECT - TETRA - UMTS - IMT-2000.

**UNIT- III: Wireless LAN** - Infrared vs. Radio transmission - Infrastructure - Adhoc Network - IEEE 802.11WLAN Standards - Architecture - Services - HIPERLAN - Bluetooth Architecture & protocols.

**UNIT- IV: Mobile Network Layer** - Mobile IP - Dynamic Host Configuration Protocol - Mobile Transport Layer - Traditional TCP - Indirect TCP - Snooping TCP - Mobile TCP - Fast retransmit/Fast recovery - Transmission/Time-out freezing - Selective retransmission - Transaction Oriented TCP.

**UNIT-V: Support for Mobility** – Wireless Application Protocol: Architecture, Wireless Datagram Protocol, Wireless Transport Layer Security, Wireless Transaction Protocol, Wireless Session Protocol, Wireless Application Environment, Wireless Markup Language, WML Scripts, Wireless Telephone Application.

**TEXT BOOKS:**

1. Jochen Schiller, “Mobile Communication”, Second Edition, Pearson Education, 2008.

**REFERENCES:**

1. William Stallings, “Wireless Communications and Networks”, Second Edition, Pearson Education, 2004.
2. C. Siva Ram Murthy, B. S. Manoj, “Adhoc Wireless Networks: Architectures and Protocols”, Second Edition, Pearson Education, 2008.

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**III B. Tech II Sem-R20**

**MEAN STACK DEVELOPMENT**
**Code: R20PE3201**

**COURSE OBJECTIVES:**

- To Learn the basics of Web Designing using HTML, DHTML, and CSS
- To learn the basics about Client side scripts and Server side scripts

**COURSE OUTCOMES (COS)**: At the end of the course, student will be able to

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Describe basics of Web Designing using HTML, DHTML, and CSS | **K2** |
| **CO2** | Build real world applications using client side and server side scripting languages | **K3** |
| **CO3** | Design and develop applications using web servers | **K5** |
| **CO4** | Analyze the basics of PHP programming | **K4** |
| **CO5** | Apply Database connectivity with case study for student Information System and Health Management system | **K3** |

#***Based on suggested Revised BTL***

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 3 | 2 | | 2 | | | | | | | 2 | |
| **CO2** | 3 | 1 | 2 | | 2 | | | | | 2 | 2 | |
| **CO3** | 2 | 2 | 2 | | | 2 | | | | | 3 | |
| **CO4** | 2 | 2 | | 3 | 2 | | | | | 3 | | |
| **CO5** | 3 | 3 | 3 | 3 | 3 | 3 | | | | | 3 | |

**(Please fill the above with Levels of Correlation, viz., L-1, M-2, H-3)**

**SYLLABUS:**

**UNIT- I: HTML & DHTML :-** Introduction, HTML Formatting, Hyper-Links, Lists, Tables, Images, Forms, Frames, Cascading Style sheets, Types, XML, Document type definition, XML Schemas, Document Object model- HTML and Scripting Access, Rollover Buttons, Moving objects with DHTML, Ramifications of DHTML.

**UNIT- II: Introduction to Client Side scripting**: JavaScript, Control statements, Functions, Arrays, Objects, Events, Dynamic HTML with Java Script - AJAX: Ajax Client Server Architecture - XML Http Request Object - Call Back Methods.

**UNIT- III: Web Application-** Web servers – IIS (XAMPP - LAMPP) and Tomcat Servers - Server Side Scripting - Java Servlets - Java Server Pages - Java Server Faces - JSF Components - Session Tracking - Cookies.

**UNIT- IV: PHP Programming** - Basic Syntax - Defining variable and constant - PHP Data types - Operator and Expression - Operator Precedence - Decisions and Loop - Functions & Recursion - String Processing and Regular Expressions - Form Processing - Working with file and Directories - Cookies.

**UNIT- V: JDBC:** Database Connectivity with MySQL - Servlets - JSP - PHP - MongoDB - NOSQL Database - Fundamentals of JQuery and Bootstrap.
Case Studies - Student information system - Health Management System

**TEXT BOOKS**:

1. Paul J. Deitel, Harvey M. Deitel, Abbey Deitel, “Internet & World Wide Web How to Program”, Fifth Edition, Deitel Series, 2012.
2. Jason Gilmore, “Beginning PHP and MySQL from Novice to Professional”, Fourth Edition, Apress Publications, 2010.

**REFERENCES:**

1. Brown, Ethan, “Web Development with Node and Express: Leveraging the JavaScript Stack”, O'Reilly Media, 2019. CSE Dept. Flexible Curriculum NITTUGCSE19 95.
2. Anthony, Accomazzo, Murray Nathaniel, Lerner Ari, “Fullstack React: The Complete Guide to React JS and Friends”, Fullstack.io, 2017.
3. Kozlowski, Pawel, “Mastering Web Application Development with Angular JS”, Packt Publishing Ltd., 2013.
4. Robert W. Sebesta, “Programming with World Wide Web”, Fourth Edition, Pearson, 2008.
5. David William Barron, “The World of Scripting Languages”, Wiley Publications, 2000.
6. Dayley B., “Node.js, MongoDB, and AngularJS Web Development”, Addison-Wesley Professional, 2014.
7. Vainikka J., “Full-Stack Web Development using Django REST Framework and React”, 2018

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**III B. Tech II Sem-R20**

**OBJECT ORIENTED ANALYSIS AND DESIGN**
**Code: R20PE3201**

**Pre-requisites:** Basic knowledge of object oriented methods, Software Engineering Concepts.

**COURSE OBJECTIVES:** The main objective is the students become familiar with all phases of OOAD.

- Master the main features of the UML.
- Master the main concepts of Object Technologies and how to apply them at work and develop the ability to analyze and solve challenging problem in various domains.
- Learn the Object design Principles and understand how to apply them towards Implementation.

**COURSE OUTCOMES:** After finishing this course student will be able to:

| CO | Course Outcome | Knowledge Level |
|---|---|---|
| **CO1** | Analyze the nature of complex system and its solutions. | **K4** |
| **CO2** | Illustrate & relate the conceptual model of the UML, identify & design the classes and relationships | **K6** |
| **CO3** | Analyze &Design Class and Object Diagrams that represent Static Aspects of a Software System and apply basic and Advanced Structural Modeling Concepts for designing real time applications. | **K6** |
| **CO4** | Analyze & Design behavioral aspects of a Software System using Use Case, Interaction and Activity Diagrams. | **K6** |
| **CO5** | Analyze & Apply techniques of State Chart Diagrams and Implementation Diagrams to model behavioral aspects and Runtime environment of Software Systems. | **K6** |

**Mapping of course outcomes with program outcomes**

| | PO 1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO1 0 | PO1 1 | PO1 2 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 2 | 3 | 2 | | 1 | 1 | 1 | 1 | 2 | | | |
| CO2 | 3 | 2 | 3 | | 2 | 3 | 3 | 2 | 3 | | | |
| CO3 | 3 | 2 | 2 | 1 | 3 | 1 | 2 | 1 | 3 | | | |
| CO4 | 1 | 1 | 3 | 2 | 1 | 2 | 2 | 3 | 3 | | | |
| CO5 | 3 | 2 | 3 | 1 | 3 | 2 | 2 | 3 | 3 | | | |

**(Please fill the above with Levels of Correlation, viz., L-1, M-2, H-3)**

**Department of Computer Science & Engineering**
**University College of Engineering, JNT University Kakinada**

**SYLLABUS:**

**UNIT-I: Introduction:** The Structure of Complex systems, The Inherent Complexity of Software, Attributes of Complex System, Organized and Disorganized Complexity, Bringing Order to Chaos, Designing Complex Systems. **Case Study:** System Architecture: Satellite-Based Navigation

**UNIT-II: Introduction to UML:** Importance of modeling, principles of modeling, object oriented modeling, conceptual model of the UML, Architecture, and Software Development Life Cycle. **Basic Structural Modeling:** Classes, Relationships, common Mechanisms, and diagrams. **Case Study:** Control System: Traffic Management.

**UNIT- III: Class & Object Diagrams:** Terms, concepts, modeling techniques for Class & Object Diagrams. **Advanced Structural Modeling:** Advanced classes, advanced relationships, Interfaces, Types and Roles, Packages. **Case Study:** AI: Cryptanalysis.

**UNIT-IV: Basic Behavioral Modeling-I:** Interactions, Interaction diagrams Use cases, Use case Diagrams, Activity Diagrams. **Case Study:** Web Application: Vacation Tracking System

**UNIT-V: Advanced Behavioral Modeling:** Events and signals, state machines, processes and Threads, time and space, state chart diagrams. **Architectural Modeling:** Component, Deployment, Component diagrams and Deployment diagrams
**Case Study:** Weather Forecasting

**TEXT BOOKS:**

1. Grady BOOCH, Robert A. Maksimchuk, Michael W. ENGLE, Bobbi J. Young, Jim Conallen, Kellia Houston , "Object- Oriented Analysis and Design with Applications", 3rd edition, 2013, PEARSON.
2. Grady Booch, James Rumbaugh, Ivar Jacobson: The Unified Modeling Language User Guide, Pearson Education.

**REFERENCE BOOKS:**

1. Meilir Page-Jones: Fundamentals of Object Oriented Design in UML, Pearson Education.
2. Pascal Roques: Modeling Software Systems Using UML2, WILEY- Dreamtech India Pvt. Ltd.
3. Atul Kahate: Object Oriented Analysis & Design, The McGraw-Hill Companies.
4. Appling UML and Patterns: An introduction to Object – Oriented Analysis and Design and Unified Process, Craig Larman, Pearson Education.

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**III B. Tech II Sem-R20**

**NETWORK PROGRAMMING**
**Code: R20PE3201**

**COURSE OBJECTIVES:**
Having successfully completed this course, the student will be able to:

- Demonstrate mastery of main protocols comprising the Internet.
- Develop skills in network programming techniques.
- Implement network services that communicate through the Internet.
- Apply the client-server model in networking applications.
- Practice networking commands available through the operating system

**COURSE OUTCOMES:** At the end of the course, student will be able to

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | **Identifying** different models and sockets | **K2** |
| **CO2** | **Demonstrate** different TCP Echo server functions and I/O models | **K3** |
| **CO3** | **Rationalize** IPV4 and IPV6 Socket options | **K4** |
| **CO4** | **Identifying** daemon processing and Advanced input and output functions | **K2** |
| **CO5** | **Analyze** Broadcasting and multicasting | **K3** |

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 3 | 2 | | 2 | | | | | | | 2 | |
| CO2 | 3 | 1 | 2 | | 2 | | | | | 2 | 2 | |
| CO3 | 2 | 2 | 2 | | | 2 | | | | | 3 | |
| CO4 | 2 | 2 | | 3 | 2 | | | | | 3 | | |
| CO5 | 3 | 3 | 3 | 3 | 3 | 3 | | | | | 3 | |

**(Please fill the above with Levels of Correlation, viz., L-1, M-2, H-3)**

**SYLLABUS:**

**UNIT-1: Introduction to Network Programming:** Introduction to Network Programming: OSI model, UNIX standards, TCP and UDP & TCP connection establishment and Format, Buffer sizes and limitation, standard internet services, Protocol usage by common internet application Elementary Sockets: Sockets introduction, Elementary TCP sockets.

**Department of Computer Science & Engineering**
**University College of Engineering, JNT University Kakinada**

**UNIT-2: TCP client server:** Introduction, TCP Echo server functions, Normal startup, terminate and signal handling server process termination, Crashing and Rebooting of server host shutdown of server host.I/O Multiplexing: I/O Models, the select and poll functions, Batch input and buffering, shutdown function.

**UNIT-3: UDP and Socket options: Elementary UDP sockets:** Introduction UDP Echo server functions, lost datagram, summary of UDP example, Lack of flow control with UDP. **Socket options:** getsockopt and setsockopt functions. Socket states, Generic socket options IPV4 socket options, IPV6 socket options, ICMPV6 socket options and TCP socket options, SCTP socket options, fcntl function.

**UNIT-4: Advanced Sockets and Daemon Processes:** IPV4 and IPV6 interoperability, introduction, IPV4 client: IPV6 server, IPV6 client: IPV4 Server, IPV6 Address-testing macros. Daemon Processes and inetdSuperserver – Introduction, syslogd Daemon, syslog Function, daemon_init Function, inetd Daemon, daemon_inetd. Advanced I/O functions: Socket timeouts, recv and send functions, ready and writev functions, recvmsg and send msg functions, Ancillary data.

**UNIT-5: Broadcasting and Multicasting:** Broadcasting introduction, broadcast addresses, unicast versus Broadcast, dg_cli function using broadcasting, race conditions, Multicasting addresses, multicasting versus broadcasting on a LAN, multicasting on a WAN, source-specific multicast, multcast socket options. **Raw Sockets**: Introduction, Raw Socket Creation, Raw Socket Output, Raw Socket Input, Ping Program, Traceroute Program

**TEXT BOOKS:**

1. UNIX Network Programming, by W. Richard Stevens, Bill Fenner, Andrew M. Rudoff, Pearson Education
2. UNIX Network Programming, 1st Edition, - W. Richard Stevens. PHI.

**REFERENCES:**

1. UNIX Systems Programming using C++ T CHAN, PHI.
2. UNIX for Programmers and Users, 3rd Edition Graham GLASS, King abls, Pearson Education
3. Advanced UNIX Programming 2nd Edition M. J. ROCHKIND, Pearson Education

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**III B. Tech II Sem-R20**

**NATURAL LANGUAGE PROCESSING**
**Code: R20OE3201**

**COURSE OBJECTIVES:**

- This course introduces the fundamental concepts and techniques of natural language processing (NLP).
- Students will gain an in-depth understanding of the computational properties of natural languages and the commonly used algorithms for processing linguistic information.
- The course examines NLP models and algorithms using both the traditional symbolic and the more recent statistical approaches.
- Enable students to be capable to describe the application based on natural language processing and to show the points of syntactic, semantic and pragmatic processing.

**COURSE OUTCOMES:** After completion of this course

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| CO1 | Demonstrate a given text with basic Language features | K4 |
| CO2 | To design an innovative application using NLP components | K6 |
| CO3 | Explain a rule based system to tackle morphology/syntax of a language | K3 |
| CO4 | To design a tag set to be used for statistical processing for real-time applications | K2 |
| CO5 | To compare and contrast the use of different statistical approaches for different types of NLP applications. | K2 |

#***Based on suggested Revised BTL***

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 2 | 3 | 2 | | 1 | 1 | 1 | 1 | 2 | | | |
| CO2 | 3 | 2 | 3 | | 2 | 3 | 3 | 2 | 3 | | | |
| CO3 | 3 | 2 | 2 | 1 | 3 | 1 | 2 | 1 | 3 | | | |
| CO4 | 1 | 1 | 3 | 2 | 1 | 2 | 2 | 3 | 3 | | | |
| CO5 | 3 | 2 | 3 | 1 | 3 | 2 | 2 | 3 | 3 | | | |

**(Please fill the above with Levels of Correlation, viz., L-1, M-2, H-3)**

**SYLLABUS:**

**UNIT I: Introduction :**Origins and challenges of NLP – Language Modeling: Grammar-based LM, Statistical LM – Regular Expressions, Finite-State Automata – English Morphology, Transducers for lexicon and rules, Tokenization, Detecting and Correcting Spelling Errors, Minimum Edit Distance.

**UNIT II: WORD LEVEL ANALYSIS:** Unsmoothed N-grams, Evaluating N-grams, Smoothing, Interpolation and Backoff – Word Classes, Part-of-Speech Tagging, Rule-based, Stochastic and Transformation-based tagging, Issues in PoS tagging – Hidden Markov and Maximum Entropy models.

**UNIT III: SYNTACTIC ANALYSIS :** Context-Free Grammars, Grammar rules for English, Treebanks, Normal Forms for grammar – Dependency Grammar – Syntactic Parsing, Ambiguity, Dynamic Programming parsing – Shallow parsing – Probabilistic CFG, Probabilistic CYK, Probabilistic Lexicalized CFGs – Feature structures, Unification of feature structures

**UNIT IV: SEMANTICS AND PRAGMATICS:** Requirements for representation, First-Order Logic, Description Logics – Syntax-Driven Semantic analysis, Semantic attachments – Word Senses, Relations between Senses, Thematic Roles, selectional restrictions – Word Sense Disambiguation, WSD using Supervised, Dictionary & Thesaurus, Bootstrapping methods – Word Similarity using Thesaurus and Distributional methods.

**UNIT V: DISCOURSE ANALYSIS AND LEXICAL RESOURCES** : Discourse segmentation, Coherence – Reference Phenomena, Anaphora Resolution using Hobbs and Centering Algorithm – Coreference Resolution – Resources: Porter Stemmer, Lemmatizer, Penn Treebank, Brill's Tagger, WordNet, PropBank, FrameNet, Brown Corpus, British National Corpus (BNC).

**TEXT BOOKS:**

1. Daniel Jurafsky, James H. Martin—Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech, Pearson Publication, 2014.
2. Steven Bird, Ewan Klein and Edward Loper, —Natural Language Processing with Python, First Edition, OReilly Media, 2009.

**REFERENCE BOOKS:**

1. Breck Baldwin, —Language Processing with Java and LingPipe Cookbook, Atlantic Publisher, 2015.
2. Richard M Reese, —Natural Language Processing with Java, OReilly Media, 2015.
3. Nitin Indurkhya and Fred J. Damerau, —Handbook of Natural Language Processing, Second, Chapman and Hall/CRC Press, 2010. Edition
4. Tanveer Siddiqui, U.S. Tiwary, —Natural Language Processing and Information Retrieval, Oxford University Press, 2008.

**Department of Computer Science & Engineering**
**University College of Engineering, JNT University Kakinada**

**III B. Tech II Sem-R20**

**BIG DATA ANALYTICS LAB**
**Code: R20PC3204**

**Software Requirements:**

1. **Hadoop** : https://hadoop.apache.org/release/2.7.6.html
2. **Java** : https://www.oracle.com/java/technologies/javase/javase8u211-later-archive-downloads.html
3. **Eclipse** : https://www.eclipse.org/downloads/

**List of Experiments:**

**Experiment 1: Week 1, 2:**

1. Implement the following Data structures in Java
   a) Linked Lists b) Stacks c) Queues d) Set e) Map

**Experiment 2: Week 3:**

2. (i)Perform setting up and Installing Hadoop in its three operating modes:
   Standalone, Pseudo distributed, Fully distributed
   (ii)Use web based tools to monitor your Hadoop setup.

**Experiment 3: Week 4:**

3. Implement the following file management tasks in Hadoop:
   - Adding files and directories
   - Retrieving files
   - Deleting files

   **Hint:** A typical Hadoop workflow creates data files (such as log files) elsewhere and copies them into HDFS using one of the above command line utilities.

**Experiment 4: Week 5:**

4. Run a basic Word Count MapReduce program to understand MapReduce Paradigm.

**Experiment 5: Week 6:**

5. Write a map reduce program that mines weather data.
   Weather sensors collecting data every hour at many locations across the globe gather a large volume of log data, which is a good candidate for analysis with Map Reduce, since it is semi structured and record-oriented.

**Experiment 6: Week 7:**

6. Use MapReduce to find the shortest path between two people in a social graph.
   **Hint**: Use an adjacency list to model a graph, and for each node store the distance from the original node, as well as a back pointer to the original node. Use the mappers to propagate the distance to the original node, and the reducer to restore the state of the graph. Iterate until the target node has been reached.

**Experiment 7: Week 8:**

7. Implement Friends-of-friends algorithm in MapReduce.

**Hint:** Two MapReduce jobs are required to calculate the FoFs for each user in a social network .The first job calculates the common friends for each user, and the second job sorts the common friends by the number of connections to your friends.

**Experiment 8: Week 9:**

8. Implement an iterative PageRank graph algorithm in MapReduce.

**Hint:** PageRank can be implemented by iterating a MapReduce job until the graph has converged. The mappers are responsible for propagating node PageRank values to their adjacent nodes, and the reducers are responsible for calculating new PageRank values for each node, and for re-creating the original graph with the updated PageRank values.

**Experiment 9: Week 10:**

9. Perform an efficient semi-join in MapReduce.

**Hint:** Perform a semi-join by having the mappers load a Bloom filter from the Distributed Cache, and then filter results from the actual MapReduce data source by performing membership queries against the Bloom filter to determine which data source records should be emitted to the reducers.

**Experiment 10: Week 11:**

10. Install and Run Pig then write Pig Latin scripts to sort, group, join, project, and filter your data.

**Experiment 12: Week 12:**

11. Install and Run Hive then use Hive to create, alter, and drop databases, tables, views, functions, and indexes

**Department of Computer Science & Engineering**
**University College of Engineering, JNT University Kakinada**

**III B. Tech II Sem-R20**

## MACHINE LEARNING USING PYTHON LAB
**Code: R20PC3205**

**COURSE OBJECTIVES:** This course will enable students to

- To learn and understand different Data sets in implementing the machine learning algorithms.

**COURSE OUTCOMES (COS)**: At the end of the course, student will be able to

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| CO1 | Implement procedures for the machine learning algorithms | K4 |
| CO2 | Design and Develop Python programs for various Learning algorithms | K6 |
| CO3 | Apply appropriate data sets to the Machine Learning algorithms | K3 |
| CO4 | Develop Machine Learning algorithms to solve real world problems | K6 |

#***Based on suggested Revised BTL***

**<u>Requirements:</u> Develop the following program using Anaconda/Jupiter/Spider and evaluate ML models.**

**Experiment-1:**
Implement and demonstrate the FIND-S algorithm for finding the most specific hypothesis based on a given set of training data samples. Read the training data from a .CSV file.

**Experiment-2:**
For a given set of training data examples stored in a .CSV file, implement and demonstrate the Candidate-Elimination algorithm to output a description of the set of all hypotheses consistent with the training examples.

**Experiment-3:**
Write a program to demonstrate the working of the decision tree based ID3 algorithm. Use an appropriate data set for building the decision tree and apply this knowledge to classify a new sample.

**Experiment-4:**
Exercises to solve the real-world problems using the following machine learning methods: a) Linear Regression b) Logistic Regression c) Binary Classifier

**Experiment-5:** Develop a program for Bias, Variance, Remove duplicates , Cross Validation

**Experiment-6:** Write a program to implement Categorical Encoding, One-hot Encoding

**Experiment-7:**
Build an Artificial Neural Network by implementing the Back propagation algorithm and test the same using appropriate data sets.

**Experiment-8:**
Write a program to implement k-Nearest Neighbor algorithm to classify the iris data set. Print both correct and wrong predictions.

**Experiment-9:** Implement the non-parametric Locally Weighted Regression algorithm in order to fit data points. Select appropriate data set for your experiment and draw graphs.

**Experiment-10:**
Assuming a set of documents that need to be classified, use the naïve Bayesian Classifier model to perform this task. Built-in Java classes/API can be used to write the program. Calculate the accuracy, precision, and recall for your data set.

**Experiment-11:** Apply EM algorithm to cluster a Heart Disease Data Set. Use the same data set for clustering using k-Means algorithm. Compare the results of these two algorithms and comment on the quality of clustering. You can add Java/Python ML library classes/API in the program.

**Experiment-12:** Exploratory Data Analysis for Classification using Pandas or Matplotlib.

**Experiment-13:**
Write a Python program to construct a Bayesian network considering medical data. Use this model to demonstrate the diagnosis of heart patients using standard Heart Disease Data Set

**Experiment-14:**
Write a program to Implement Support Vector Machines and Principle Component Analysis

**Experiment-15:**
Write a program to Implement Principle Component Analysis

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**III B. Tech II Sem-R20**

**CRYPTOGRAPHY AND NETWORK SECURITY LAB**
**Code: R20PC3206**

**COURSE OBJECTIVES:**

- Understand the basic categories of threats to computers and networks
- Discusses the Mathematics of Cryptography
- Discuss the fundamental ideas of Symmetric and Asymmetric cryptographic Algorithms
- Discusses the Network layer, Transport Layer and Application layer Protocols Enhanced security mechanisms

**COURSE OUTCOMES:** At the end of the course, student will be able to

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| CO1 | Implement the string (char pointer) with a value. | K2 |
| CO2 | Development and perform encryption and decryption using the AND and XOR operations | K3 |
| CO3 | Apply perform encryption and decryption using the following algorithms DES, Blowfish | K3 |
| CO4 | Implement the perform encryption and decryption using the RSA. | K4 |

**List of Experiments:**

**Experiment 1:**

1. Write a C program that contains a string (char pointer) with a value 'Hello world'. The program should XOR each character in this string with 0 and displays the result.

**Experiment 2:**

2. Write a C program that contains a string (char pointer) with a value 'Hello world'. The program should AND or and XOR each character in this string with 127 and display the result.

**Experiment 3:**

3. Write a Java program to perform encryption and decryption using the following algorithms
   a. Ceaser cipher b. Substitution cipher c. Hill Cipher

**Experiment 4:**

4. Write a C/JAVA program to implement the DES algorithm logic.

**Experiment 5:**

5. Write a C/JAVA program to implement the Blowfish algorithm logic.

**Experiment 6:**

6. Write a C/JAVA program to implement the Rijndael algorithm logic.

**Experiment 7:**

7. Write the RC4 logic in java Using Java cryptography; encrypt the text “H ello world” using Blowfish. Create your own key using Java key tool.

**Experiment 8:**

8. Write a Java program to implement RSA algorithm.

**Experiment 9:**

9. Implement the Diffie-Hellman Key Exchange mechanism using HTML and JavaScript.

**Experiment 10:**

10. Calculate the message digest of a text using the SHA-1 algorithm in JAVA.

**Experiment 11:**

11. Calculate the message digest of a text using the MD5 algorithm in JAVA.

**Experiment 12:**

12. Implementation of few programs using Mathematica

**Experiment 13:**

13. Implementation of few programs using MatLab

**Experiment 14:**

14. Implementation of few programs using Maple

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**III B. Tech II Sem-R20**

**MEAN STACK TECHNOLOGIES Module-I**
**(Node.js, Angular JS, Mongo DB)**
**Code: R20SO3201**

**Pre-requisites:** Html, Xml, Java Script

**COURSE OBJECTIVES:**

- To Learn the basics of Web Designing using HTML, DHTML, and CSS
- To learn the basics about Client side scripts and Server side scripts

**Course Outcomes:** After completion of this course

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Apply the File System to perform the following operations Nodejs | **K4** |
| **CO2** | Create our first application Controller working with Angular Js modules | **K2** |
| **CO3** | Create an application for Databases using AngularJS | **K2** |
| CO4 | Develop the programs Invoking data using Mongo DB. | **K4** |

**List of Experiments:**

**Experiment 1:** Download and Install Node.js and NPM

**Experiment 2:** Write a Program to implement the Hello World Server with HTTP Node.js Module

**Experiment 3:** Write a program to create Calculator Node.js Module with functions adds, subtract & multiply and use the Calculator module in another Node.js file.

**Experiment 4:** Write a Node.js for File System to perform the following operations

i. Create a File
ii. Read a File
iii. Write to a File
iv. Delete a File

**Experiment 5:** Create and manage an Employee Database Using Node.js MySQL

**Experiment 6:** Implement the following in Angular JS

i. Angular Js data binding.
ii. Angular JS directives and Events.
iii. Using angular Js fetching data from MySQL.

**Experiment 7:** Write a program to implement AngularJS Scope

**Experiment 8:** Write a program to implement the following using Angular JS

i.Input Validation
ii.Backend Building

**Department of Computer Science & Engineering**
**University College of Engineering, JNT University Kakinada**

**Experiment 9:** Write a program to include the framework's Stylesheet in the Angular CLI's Configuration

**Experiment 10:** Write a program to Create an application for Students Record using AngularJS

**Experiment 11:** Create database, Create collection, insert data, find, find one, sort, limit, skip,distinct, projection

**Experiment 12:** Develop and demonstrate Invoking data using Mongo DB.

**Experiment 13:** Create an Online fee payment form using MangoDB

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**III B. Tech II Sem-R20**

# Employability skills-II

**Code: R20MC3201**

**Course Outcomes:** After completion of this course

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Solve various Basic Mathematics problems by following different methods | **K3** |
| **CO2** | Follow strategies in minimizing time consumption in problem solving Apply shortcut methods to solve problems | **K2** |
| **CO3** | Confidently solve any mathematical problems and utilize these mathematical skills both in their professional as well as personal life. | **K3** |
| **CO4** | Analyze, summarize and present information in quantitative forms including table, graphs and formulas | **K4** |

**SYLLABUS:**

**UNIT I:**

**Numerical ability l:** Number system, HCF & LCM, Average, Simplification, Problems on numbers

**Numerical ability II:** Ratio & Proportion, Partnership, Percentages, Profit & Loss

**UNIT II:**

**Arithmetical ability l:** Problems on ages, Time & Work, Pipes & Cistern, Chain Rule.

**Arithmetical ability ll:** Time & Distance, Problems on boats &Steams, Problems on Trains

**UNIT III:**

**Arithmetical ability lll:** Allegation, Simple interest and compound interest, Races & Games of skills, Calendar and Clock,

**Logical ability:** Permutations and Combination and Probability.

**UNIT IV: Mensuration:** Geometry, Areas, Volumes,

**Data interpretation:** Tabulation, Bar graphs, Pie charts, line graphs

**TEXT BOOKS AND REFERENCE BOOKS:**

1. R. S. Aggarwal “Quantitative Aptitude”, Revised ed., S Chand publication, 2017 ISBN:8121924987

**E- resources:**

1. https://blog.feedspot.com/aptitude_youtube_channels/
2. https://www.tutorialspoint.com/quantitative_aptitude/
3. https://www.careerbless.com/aptitude/qa/home.php

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**IV B. Tech I Sem-R20**

### CLOUD COMPUTING
### Code: R20PE4101

**COURSE OBJECTIVES:**

- To explain the evolving computer model caned cloud computing.
- To introduce the various levels of services that can be achieved by cloud.
- To describe the security aspects in cloud.
- To motivate students to do programming and experiment with the various cloud computing environments.

**COURSE OUTCOMES:** At the end of the course, student will be able to

| CO | Course Outcomes | Knowledge Level(K)# |
|---|---|---|
| **CO1** | Illustrate the key dimensions of the challenge of Cloud Computing | **K2** |
| **CO2** | Classify the Levels of Virtualization and mechanism of tools. | **K2** |
| **CO3** | Analyze Cloud infrastructure including Google Cloud and Amazon Cloud. | **K4** |
| **CO4** | Create Combinatorial Auctions for cloud resource and design scheduling algorithms for computing cloud | **K6** |
| **CO5** | Assess control storage systems and cloud security, the risks involved its impact and develop cloud application | **K5** |

***Mapping of course outcomes with program outcomes***

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 1 | 2 | 3 | | 1 | | | | | | | |
| **CO2** | 1 | 2 | 3 | | | 2 | | | | | | |
| **CO3** | | 2 | 2 | 1 | | 3 | | 1 | | | | |
| **CO4** | 2 | 2 | 3 | | 1 | | 1 | | | | | |
| **CO5** | 1 | 2 | 3 | 1 | 2 | | 2 | | | 1 | | 1 |

**(Levels of Correlation: 1-low,2-medium 3-high)**

**SYLLABUS:**

**UNIT-I: SYSTEMS MODELING, CLUSTERING AND VIRTUALIZATION:** Scalable Computing over the Internet-The Age of Internet Computing, Scalable computing over the internet, Technologies for Network Based Systems, System models for Distributed and Cloud Computing, , Performance, Security and Energy Efficiency

**UNIT-II: VIRTUAL MACHINES AND VIRTUALIZATION OF CLUSTERS AND DATA CENTERS:** Implementation Levels of Virtualization, Virtualization Structures/ Tools and Mechanisms, Virtualization of CPU, Memory and I/O Devices, Virtual Clusters and Resource Management, Virtualization for Data-Center Automation.

**UNIT-III: CLOUD PLATFORM ARCHITECTURE:** Cloud Computing and Service Models, Public Cloud Platforms, Service Oriented Architecture, Programming on Amazon AWS and Microsoft Azure

**UNIT-IV: CLOUD RESOURCE MANAGEMENT AND SCHEDULING:** Policies and Mechanisms for Resource Management, Applications of Control Theory to Task Scheduling on a Cloud, Stability of a Two Level Resource Allocation Architecture, Feedback Control Based on Dynamic Thresholds. Coordination of Specialized Autonomic Performance Managers, Resource Bundling, Scheduling Algorithms for Computing Clouds-Fair Queuing, Start Time Fair Queuing.(

**UNIT-V: STORAGE SYSTEMS:** Evolution of storage technology, storage models, file systems and database, distributed file systems, general parallel file systems. Google file system.

**TEXT BOOKS:**

1. Distributed and Cloud Computing, Kai Hwang, Geoffry C. Fox, Jack J. Dongarra MK Elsevier.
2. Cloud Computing, Theory and Practice, Dan C Marinescu, MK Elsevier.

**REFERENCE BOOKS:**

1. Cloud Computing, A Hands on approach, ArshadeepBahga, Vijay Madisetti, University Press
2. Cloud Computing, A Practical Approach, Anthony T Velte, Toby J Velte, Robert Elsenpeter, TMH
3. Mastering Cloud Computing, Foundations and Application Programming, Raj Kumar Buyya, Christen vecctiola, S Tammaraiselvi, TMH

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**IV B. Tech I Sem-R20**

**NEURAL NETWORKS AND SOFT COMPUTING**
**Code: R20PE4101**

**COURSE OBJECTIVES:** The main objective of the course is
- To introduce the foundations of Artificial Neural Networks.
- To acquire the knowledge on Soft Computing Concepts.
- To learn various types of Genetic algorithms and its applications.
- To gain knowledge to apply optimization strategies.

**COURSE OUTCOMES (COs)**: At the end of the course, student will be able to

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Understand the concepts of Artificial intelligence and soft computing techniques | **K2** |
| **CO2** | Analyze the concepts of Neural Networks and select the Learning Networks in modeling real world systems. | **K4** |
| **CO3** | Implement the concepts of Fuzzy reasoning and concepts of Genetic algorithm and its applications to soft computing. | **K6** |
| **CO4** | Classify Biologically inspired algorithm such as neural networks, genetic algorithms, ant colony optimization, and bee colony optimization. | **K3** |
| **CO5** | Design hybrid system incorporating neural network, genetic algorithms, fuzzy systems. | **K6** |

***#Based on suggested Revised BTL***

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 3 | 2 | | 2 | | | | | | | 2 | |
| CO2 | 3 | 1 | 2 | | 2 | | | | | 2 | 2 | |
| CO3 | 2 | 2 | 2 | | | 2 | | | | | 3 | |
| CO4 | 2 | 2 | | 3 | 2 | | | | | 3 | | |
| CO5 | 3 | 3 | 3 | 3 | 3 | 3 | | | | | 3 | |

**(Please fill the above with Levels of Correlation, viz., L-1, M-2, H-3)**

**SYLLABUS:**

**UNIT I: Soft Computing and Artificial Intelligence:** Introduction of Soft Computing, Soft Computing vs. Hard Computing, Various Types of Soft Computing Techniques, Applications of Soft Computing, AI Search Algorithm, Predicate Calculus, Rules of Interference, Semantic Networks, Frames, Objects, Hybrid Models.

**UNIT II: Artificial Neural Networks and Paradigms:** Introduction to Neuron Model, Neural Network Architecture, Learning Rules, Perceptrons, Single Layer Perceptrons, Multilayer Perceptrons, Back propagation Networks, Kohnen'sself organizing networks, Hopfield network, Applications of NN.

**UNIT – III: Fuzzy Logic:** Introduction, Fuzzy sets and Fuzzy reasoning, Basic functions on fuzzy sets, relations, rule based models and linguistic variables, fuzzy controls, Fuzzy decision making, applications of fuzzy logic.

**UNIT – IV: Genetic Algorithms and Swarm Optimizations:** Introduction, Genetic Algorithm, Fitness Computations, Cross Over, Mutation, Evolutionary Programming, Classifier Systems, Genetic Programming Parse Trees, Variants of GA, Applications, Ant Colony Optimization, Particle Swarm Optimization, Artificial Bee Colony Optimization.

**UNIT –V: Hybrid Systems:** Neuro fuzzy hybrid systems, Adaptive neurofuzzy inference systems, Fuzzy backpropagation network, Genetic neuro hybrid system, Genetic algorithm based backpropagation network, Genetic-fuzzy hybrid systems.

**TEXT BOOKS:**

1. Simon S. Haykin, Neural Networks, Prentice Hall, 2nd edition.
2. S. Rajasekaran & G. A. Vijayalakshmi Pai “Neural Networks, Fuzzy Logic and Genetic Algorithms: Synthesis & Applications”, PHI,2003.

**REFERENCES:**

1. S. N. Sivanandam & S. N. Deepa ”Principles of Soft Computing” Wiley – India, 2nd Edition, 2007.
2. Jang J.S.R., Sun C.T. and Mizutani E, "Neuro-Fuzzy and Soft computing", Prentice Hall, 1998.
3. Jacek M. Zurada, Introduction to Artificial Neural Systems, Jaico Publishing House,1994
4. Zimmermann, “Fuzzy Set Theory and its Application”, 3rd Edition.
5. D.E. Goldberg, "Genetic Algorithms: Search, Optimization and Machine Learning", Addison Wesley, N.Y, 1989.
6. Timothy J. Ross, "Fuzzy Logic with Engineering Applications", McGraw Hill, 3rd edition 2009.

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**IV B. Tech I Sem-R20**

### AD-HOC AND SENSOR NETWORKS
### Code: R20PE4101

**COURSE OUTCOMES:**

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Evaluate the principles and characteristics of mobile ad hoc networks (MANETs) and what distinguishes them from infrastructure-based networks. | **K5** |
| **CO2** | Determine the principles and characteristics of wireless sensor networks. | **K5** |
| **CO3** | Discuss the challenges in designing MAC, routing and transport protocols for wireless ad-hoc sensor networks. | **K6** |
| **CO4** | Illustrate the various sensor network Platforms, tools and applications. | **K4** |
| **CO5** | Demonstrate the issues and challenges in security provisioning and also familiar with the mechanisms for implementing security and trust mechanisms in MANETs and WSNs. | **K2** |

***#Based on suggested Revised BTL***

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 3 | 2 | | 2 | | | | | | | 2 | |
| CO2 | 3 | 1 | 2 | | 2 | | | | | 2 | 2 | |
| CO3 | 2 | 2 | 2 | | | 2 | | | | | 3 | |
| CO4 | 2 | 2 | | 3 | 2 | | | | | 3 | | |
| CO5 | 3 | 3 | 3 | 3 | 3 | 3 | | | | | 3 | |

**(Please fill the above with Levels of Correlation, viz., L-1, M-2, H-3)**

**SYLLABUS:**

**UNIT-I:Introduction to Ad Hoc Wireless Networks-** Cellular and Ad Hoc Wireless Networks, Characteristics of MANETs, Applications of MANETs, Issues and Challenges of MANETs, Ad Hoc Wireless Internet, MAC protocols for Ad hoc Wireless Networks-Issues, Design Goals and Classifications of the MAC Protocols.

**UNIT-II: Routing Protocols for Ad Hoc Wireless Networks-** Issues in Designing a Routing Protocol, Classifications of Routing Protocols, Topology-based versus Position-based Approaches, Issues and design goals of a Transport layer protocol, Classification of Transport layer solutions, TCP over Ad hoc Wireless Networks, Solutions for TCP over Ad Hoc Wireless Networks, Other Transport layer protocols.

**UNIT-III: Security protocols for Ad hoc Wireless Networks-** Security in Ad hoc Wireless Networks, Network Security Requirements, Issues and Challenges in Security Provisioning, Network Security Attacks, Key Management, Secure Routing in Ad hoc Wireless Networks, Cooperation in MANETs, Intrusion Detection Systems.

**UNIT-IV: Basics of Wireless Sensors and Applications-** The Mica Mote, Sensing and Communication Range, Design Issues, Energy Consumption, Clustering of Sensors, Applications, Data Retrieval in Sensor Networks-Classification of WSNs, MAC layer, Routing layer, Transport layer, High-level application layer support, Adapting to the inherent dynamic nature of WSNs.

**UNIT-V: Security in WSNs-** Security in WSNs, Key Management in WSNs, Secure Data Aggregation in WSNs, Sensor Network Hardware-Components of Sensor Mote, Sensor Network Operating Systems–TinyOS, LA-TinyOS, SOS, RETOS, Imperative Language-nesC, **Dataflow Style Language-**TinyGALS, Node-Level Simulators, NS-2 and its sensor network extension, TOSSIM.

**TEXT BOOKS:**

1. Ad Hoc Wireless Networks – Architectures and Protocols, 1st edition, C. Siva Ram Murthy, B. S. Murthy, Pearson Education, 2004
2. Ad Hoc and Sensor Networks – Theory and Applications, 2nd edition *Carlos Corderio Dharma P.Aggarwal,* World Scientific Publications / Cambridge University Press, March 2006

**REFERENCE BOOKS:**

1. Wireless Sensor Networks: An Information Processing Approach, 1stedition , *Feng Zhao, Leonidas Guibas*, Elsevier Science imprint, Morgan Kauffman Publishers, 2005, rp2009
2. Wireless Ad hoc Mobile Wireless Networks – Principles, Protocols and Applications, 1st edition, Subir Kumar Sarkar, et al., Auerbach Publications, Taylor & Francis Group, 2008
3. Ad hoc Networking, 1st edition,*Charles E.Perkins*, Pearson Education, 2001
4. Wireless Ad hoc Networking, 1st edition, *Shih-Lin Wu, Yu-Chee Tseng,* Auerbach Publications, Taylor & Francis Group, 2007
5. Wireless Sensor Networks – Principles and Practice, 1st edition, Fei Hu, Xiaojun Cao, An Auerbach book, CRC Press, Taylor & Francis Group, 2010

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**IV B. Tech I Sem-R20**

**CYBER SECURITY & FORENSICS**
**Code: R20PE4101**

**COURSE OBJECTIVES:**

- The aim of the course is to identify security risks and take preventive steps
- To understand the forensics fundamentals
- To understand the evidence capturing process
- To understand the preservation of digital evidence

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

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Explain the Cybercrime Fundamentals | **K2** |
| **CO2** | Describe the types of attacks on networks | **K3** |
| **CO3** | Analyze various tools available for Cybercrime Investigation | **K4** |
| **CO4** | Explain the Computer Forensics and Investigation Fundamentals and tools | **K2** |
| **CO5** | Analyze the legal perspectives of Cybercrime | **K4** |

***#Based on suggested Revised BTL***

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 1 | 2 | 3 | 2 | | | | | | | 1 | |
| **CO2** | 2 | 3 | 2 | | 2 | | | | | 2 | 2 | |
| **CO3** | 1 | 2 | 3 | | | 2 | | | | | 1 | |
| **CO4** | 3 | 2 | 2 | 3 | 2 | | | | | 3 | | |
| **CO5** | 3 | 2 | 3 | | 1 | 3 | | | | | 3 | |

**(Please fill the above with Levels of Correlation, viz., L-1, M-2, H-3)**

**SYLLABUS:**

**UNIT-I: Introduction to Cybercrime:** Introduction, Cybercrime: Definition and Origins of the Word, Cybercrime and Information Security, Cybercriminals, Classifications of Cybercrime, Cyberstalking, Cybercafe and Cybercrimes, Botnets. Attack Vector, Proliferation of Mobile and Wireless Devices, Security Challenges Posed by Mobile Devices, Attacks on Mobile/Cell Phones, Network and Computer Attacks.

**UNIT II: Tools and Methods :** Proxy Servers and Anonymizers, Phishing, Password Cracking, Keyloggersand Spywares, Virus and Worms, Trojan Horses and Backdoors, Steganography, Sniffers, Spoofing, Session Hijacking Buffer over flow, DoS and DDoS Attacks, SQL Injection, Buffer Overflow, Attacks on Wireless Networks, Identity Theft (ID Theft), Foot Printing and Social Engineering, Port Scanning, Enumeration.

Department of Computer Science & Engineering
University College of Engineering, JNT University Kakinada

**UNIT III: Cyber Crime Investigation:** Introduction, Investigation Tools, eDiscovery, Digital Evidence Collection, Evidence Preservation, E-Mail Investigation, E-Mail Tracking, IP Tracking, E-Mail Recovery, Hands on Case Studies. Encryption and Decryption Methods, Search and Seizure of Computers, Recovering Deleted Evidences, Password Cracking.

**UNIT IV: Computer Forensics and Investigations:** Understanding Computer Forensics, Preparing for Computer Investigations. Current Computer Forensics Tools: Evaluating Computer Forensics Tools, Computer Forensics Software Tools, Computer Forensics Hardware Tools, Validating and Testing Forensics Software, Face, Iris and Fingerprint Recognition, Audio Video Analysis, Windows System Forensics, Linux System Forensics, Graphics and Network Forensics, E-mail Investigations, Cell Phone and Mobile Device Forensics.

**UNIT V: Cyber Crime Legal Perspectives:** Introduction, Cybercrime and the Legal Landscape around the World, The Indian IT Act, Challenges to Indian Law and Cybercrime Scenario in India, Consequences of Not Addressing the Weakness in Information Technology Act, Digital Signatures and the Indian IT Act, Amendments to the Indian IT Act, Cybercrime and Punishment, Cyberlaw, Technology and Students: Indian Scenario.

**TEXT BOOKS:**

1. Sunit Belapure Nina Godbole “Cyber Security: Understanding Cyber Crimes, Computer Forensics and Legal Perspectives”, WILEY, 2011.
2. Nelson Phillips and Enfinger Steuart, “Computer Forensics and Investigations”, Cengage Learning, New Delhi, 2009.

**REFERENCE BOOKS:**

1. Michael T. Simpson, Kent Backman and James E. Corley, “Hands on Ethical Hacking and Network Defence”, Cengage, 2019.
2. Computer Forensics, Computer Crime Investigation by John R. Vacca, Firewall Media, New Delhi.
3. Alfred Basta, Nadine Basta,Mary Brown and Ravinder Kumar “Cyber Security and Cyber Laws” , Cengage,2018.

**E-Resources:**

1. CERT-In Guidelines- http://www.cert-in.org.in/
2. https://www.coursera.org/learn/introduction-cybersecurity-cyber-attacks [ Online Course]
3. https://computersecurity.stanford.edu/free-online-videos [ Free Online Videos]
4. Nickolai Zeldovich. 6.858 Computer Systems Security. Fall 2014. Massachusetts Institute of Technology: MIT OpenCourseWare, https://ocw.mit.edu License: Creative Commons BY-NC-SA.

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**IV B. Tech I Sem-R20**

**DEEP LEARNING TECHNIQUES**
**Code: R20PE4102**

**COURSE OBJECTIVES:** At the end of the course, the students will be expected to:

- Learn deep learning methods for working with sequential data,
- Learn deep recurrent and memory networks,
- Learn deep Turing machines,
- Apply such deep learning mechanisms to various learning problems.
- Know the open issues in deep learning, and have a grasp of the current research directions.

**COURSE OUTCOMES:** After the completion of the course, student will be able to

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Demonstrate the fundamental concepts learning techniques of Artificial Intelligence, Machine Learning and Deep Learning. | **K2** |
| **CO2** | Discuss the Neural Network training, various random models. | **K6** |
| **CO3** | Explain the Techniques of Keras, TensorFlow, Theano and CNTK | **K5** |
| **CO4** | Classify the Concepts of CNN and RNN | **K2** |
| **CO5** | Implement Interactive Applications of Deep Learning. | **K3** |

**#based on suggested Revised BTL**

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 1 | 3 | 2 | | 2 | | | | | | | 2 |
| **CO2** | | 3 | 1 | 2 | | 2 | | | | | 2 | 2 |
| **CO3** | 1 | 2 | 2 | 2 | | | 2 | | | | | 3 |
| **CO4** | | 2 | 2 | | 3 | 2 | | | | | 3 | |
| **CO5** | 2 | 3 | 3 | 3 | 3 | 3 | 3 | | | | | 3 |

**(Please fill the above with Levels of Correlation, viz., L, M, H)**

**SYLLABUS:**

**UNIT- 1: Fundamentals of Deep Learning:** Artificial Intelligence, History of Machine learning: Probabilistic Modeling, Early Neural Networks, Kernel Methods, Decision Trees, Random forests and Gradient Boosting Machines, **Fundamentals of Machine Learning:** Four Branches of Machine Learning, Evaluating Machine learning Models, Overfitting and Underfitting. **[Text Book 2]**

**UNIT- 2: Introducing Deep Learning:** Biological and Machine Vision, Human and Machine Language, Artificial Neural Networks, Training Deep Networks, Improving Deep Networks. **[Text Book3]**

**UNIT- 3: Neural Networks:** Anatomy of Neural Network, Introduction to Keras: Keras, TensorFlow, Theano and CNTK, Setting up Deep Learning Workstation, Classifying Movie Reviews: Binary Classification, Classifying newswires: Multiclass Classification. **[Text Book 2]**

**UNIT- 4: Convolutional Neural Networks:** Nerual Network and Representation Learing, Convolutional Layers, Multichannel Convolution Operation, **Recurrent Neural Networks:** Introduction to RNN, RNN Code, PyTorch Tensors: Deep Learning with PyTorch, CNN in PyTorch. **[Text Book 3]**

**UNIT-5: Interactive Applications of Deep Learning:** Machine Vision, Natural Language processing, Generative Adversial Networks, Deep Reinforcement Learning. **[Text Book 1]**

**Deep Learning Research:** Autoencoders**,** Deep Generative Models: Boltzmann Machines Restricted Boltzmann Machines, Deep Belief Networks. **[Text Book 1]**

**TEXT BOOKS:**

1. Deep Learning- Ian Goodfellow, Yoshua Bengio and Aaron Courvile, MIT Press, 2016
2. Deep Learning with Python - Francois Chollet, Released December 2017, Publisher(s): Manning Publications, ISBN: 9781617294433
3. Deep Learning Illustrated: A Visual, Interactive Guide to Artificial Intelligence - Jon Krohn, Grant Beyleveld, Aglaé Bassens, Released September 2019, Publisher(s): Addison-Wesley Professional, ISBN: 9780135116821
4. Deep Learning from Scratch - Seth Weidman, Released September 2019, Publisher(s): O'Reilly Media, Inc., ISBN: 9781492041412

**REFERENCE BOOKS:**

1. Artificial Neural Networks, Yegnanarayana, B., PHI Learning Pvt. Ltd, 2009.
2. Matrix Computations, Golub, G.,H., and Van Loan,C.,F, JHU Press,2013.
3. Neural Networks: A Classroom Approach, Satish Kumar, Tata McGraw-Hill Education, 2004.

**Web Link:**

1. Swayam NPTEL: Deep Learning: https://onlinecourses.nptel.ac.in/noc22_cs22/preview

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**IV B. Tech I Sem-R20**

**SOCIAL NETWORKS & SEMANTIC WEB**
**Code: R20PE4102**

**COURSE OBJECTIVES:**

- To learn Web Intelligence
- To learn Knowledge Representation for the Semantic Web
- To learn Ontology Engineering
- To learn Semantic Web Applications, Services and Technology
- To learn Social Network Analysis and semantic web

**COURSE OUTCOMES:** After the completion of the course, student will be able to

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| CO1 | Demonstrate social network analysis and measures. | K2 |
| CO2 | Analyze random graph models and navigate social networks data | K4 |
| CO3 | Apply the network topology and Visualization tools. | K3 |
| CO4 | Analyze the experiment with small world models and clustering models. | K4 |
| CO5 | Compare the application driven virtual communities from social network Structure. | K5 |

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | | 1 | | | 3 | 2 | | | | | | |
| CO2 | | | 1 | 2 | | 3 | | | | | | |
| CO3 | 1 | 2 | | | 3 | | | | | | | |
| CO4 | | 1 | 2 | | | | 3 | | | | | |
| CO5 | | 3 | 1 | | 2 | | | | | | | |

**(Please fill the above with Levels of Correlation, viz., L-1, M-2, H-3)**

**SYLLABUS:**

**UNIT I: Web Intelligence:** Thinking and Intelligent Web Applications, The Information Age ,The World Wide Web, Limitations of Today's Web, The Next Generation Web, Machine Intelligence, Artificial Intelligence, Ontology, Inference engines, Software Agents, Berners-Lee www, Semantic Road Map, Logic on the semantic Web.

**UNIT -II: Knowledge Representation for the Semantic Web:** Ontologies and their role in the semantic web, Ontologies Languages for the Semantic Web – Resource Description Framework(RDF) / RDF Schema, Ontology Web Language(OWL), UML, XML/XML Schema.

**UNIT-III: Ontology Engineering:** Ontology Engineering, Constructing Ontology, Ontology Development Tools, Ontology Methods, Ontology Sharing and Merging, Ontology Libraries and Ontology Mapping, Logic, Rule and Inference Engines.

**UNIT-IV: Semantic Web Applications, Services and Technology:** Semantic Web applications and services, Semantic Search, e-learning, Semantic Bioinformatics, Knowledge Base ,XML Based Web Services, Creating an OWL-S Ontology for Web Services, Semantic Search Technology, Web Search Agents and Semantic Methods,

**UNIT-V: Social Network Analysis and semantic web:** What is social Networks analysis, development of the social networks analysis, Electronic Sources for Network Analysis – Electronic Discussion networks, Blogs and Online Communities, Web Based Networks, Building Semantic Web Applications with social network features.

**TEXT BOOKS**:

1. Thinking on the Web – Berners Lee, Godel and Turing, Wiley inter science, 2008.
2. Social Networks and the Semantic Web, Peter Mika, Springer, 2007.

**REFERENCE BOOKS**:

1. Semantic Web Technologies, Trends and Research in Ontology Based Systems, J. Davies, R. Studer, P. Warren, John Wiley & Sons.
2. Semantic Web and Semantic Web Services -Liyang Lu Chapman and Hall/CRC Publishers,(Taylor & Francis Group)
3. Information sharing on the semantic Web – Heiner Stuckenschmidt; Frank Van Harmelen, Springer Publications.
4. Programming the Semantic Web, T. Segaran, C. Evans, J. Taylor, O'Reilly, SPD.

**Department of Computer Science & Engineering**
**University College of Engineering, JNT University Kakinada**

**IV B. Tech I Sem-R20**

## DEVOPS
**Code: R20PE4102**

**COURSE OBJECTIVES:**

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

**COURSE OUTCOMES:** At the end of the course, student will be able to

| CO | Course Outcome | Knowledge Level(#) |
|---|---|---|
| **CO1** | Enumerate the principles of continuous development and deployment, automation of configuration management, inter-team collaboration, and IT service agility. | **K2** |
| **CO2** | Describe DevOps & DevSecOps methodologies and their key concepts | **K5** |
| **CO3** | Illustrate the types of version control systems, continuous integration tools, continuous monitoring tools, and cloud models | **K2** |
| **CO4** | Set up complete private infrastructure using version control systems and CI/CD tools | **K3** |
| **CO5** | Acquire the knowledge of maturity model, Maturity Assessment | **K5** |

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | | 1 | | | 3 | 2 | | | | 1 | | |
| CO2 | | | 1 | 2 | | 3 | | | | | 1 | |
| CO3 | 1 | 2 | | | 3 | | | | | | | 1 |
| CO4 | | 1 | 2 | | | | 3 | | | | | |
| CO5 | | 3 | 1 | | 2 | | | | | | | 1 |

**(Please fill the above with Levels of Correlation, viz., L-1, M-2, H-3)**

**SYLLABUS:**

**UNIT I:** Phases of Software Development Life Cycle, Values and principles of agile software development.

**UNIT II**: Fundamentals of DevOps: Architecture, Deployments, Orchestration, Need, Instance of applications, DevOps delivery pipeline, DevOps eco system.

**UNIT III:** DevOps adoption in projects: Technology aspects, Agiling capabilities, Tool stack implementation, People aspect, processes

**UNIT IV:** CI/CD: Introduction to Continuous Integration, Continuous Delivery and Deployment, Benefits of CI/CD, Metrics to track CICD practices

**UNIT V:** Devops Maturity Model: Key factors of DevOps maturity model, stages of Devops maturity model, DevOps maturity Assessment

**TEXT BOOKS:**

1. The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations, Gene Kim , John Willis , Patrick Debois , Jez Humb,1st Edition, O'Reilly publications, 2016.
2. What is Devops? Infrastructure as code, 1st Edition, Mike Loukides ,O'Reilly publications, 2012.

**REFERENCE BOOKS:**

1. Building a DevOps Culture, 1st Edition, Mandi Walls, O'Reilly publications, 2013.

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**IV B. Tech I Sem-R20**

**BLOCKCHAIN TECHNOLOGIES**
**Code: R20PE4103**

**COURSE OBJECTIVES:**

- To understand block chain technology and Cryptocurrency works

**COURSE OUTCOMES:** After the completion of the course, student will be able to

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Demonstrate the block chain basics, Crypto currency | **K2** |
| **CO2** | To compare and contrast the use of different private vs. public block chain and use cases | **K6** |
| **CO3** | Design an innovative Bit coin Block chain and scripts, Block chain Science on varies coins | **K5** |
| **CO4** | Classify Permission Block chain and use cases – Hyper ledger, Corda | **K2** |
| **CO5** | Make Use of Block-chain in E-Governance, Land Registration, Medical-Information Systems, and others | **K3** |

**#based on suggested Revised BTL**

**Mapping of course outcomes with program outcomes**

| CO | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 1 | 3 | 2 | | 2 | | | | | | | 2 |
| CO2 | | 3 | 1 | 2 | | 2 | | | | | 2 | 2 |
| CO3 | 1 | 2 | 2 | 2 | | | 2 | | | | | 3 |
| CO4 | | 2 | 2 | | 3 | 2 | | | | | 3 | |
| CO5 | 2 | 3 | 3 | 3 | 3 | 3 | 3 | | | | | 3 |

**(Please fill the above with Levels of Correlation, viz., L-1, M-2, H-3)**

**SYLLABUS:**

**UNIT- I Introduction**: Introduction – basic ideas behind blockchain, how it is changing the landscape of digitalization, introduction to cryptographic concepts required, Block chain or distributed trust, Protocol, Currency, Cryptocurrency, How a Cryptocurrency works, Crowdfunding, contracts-Financial services, Bitcoin prediction markets, smart property, smart contracts.

**UNIT- II:** Hashing, public key cryptosystems, private vs public blockchain and use cases, Hash Puzzles, Extensibility of Block chain concepts, Digital Identity verification, Block chain Neutrality, Digital art, Block chain Environment

**UNIT- III Introduction to Bitcoin :**Bitcoin Block chain and scripts, Use cases of Bitcoin Blockchain scripting language in micropayment, escrow etc Downside of Bit coin – mining, Block chain Science: Grid coin, Folding coin, Block chain Genomics, Bit coin MOOCs, Alternative coins, Smart contracts,

**UNIT – IV :** Ethereum continued, IOTA, The real need for mining – consensus – Byzantine Generals Problem, and Consensus as adistributed coordination problem – Coming to private or permissioned blockchains –Introduction to Hyper ledger, Currency, Token, Tokenizing, Campus coin, Coin drop as a strategy for Public adoption, Currency Multiplicity, Demurrage currency

**UNIT – V:** Technical challenges, Business model challenges, Scandals and Public perception, Government Regulations, Uses of Block chain in E-Governance, Land Registration, Medical Information Systems.

**TEXT BOOK:**

1. Blockchain Blue print for Economy by Melanie Swan

**REFERENCES:**

1. Blockchain Basics: A Non-Technical Introduction in 25 Steps 1st Edition, by Daniel Drescher

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**IV B. Tech I Sem-R20**

**WIRELESS NETWORK SECURITY**
**Code: R20PE4103**

**COURSE OBJECTIVES:**

- The objective of this course is to understand the importance of Wireless networks security and its application

**COURSE OUTCOMES:** After the completion of the course, student will be able to

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| CO1 | Explain the Threats in networks and provide Authentication to real time problems. | K2 |
| CO2 | Identify and investigate in-depth both early and contemporary threats to wireless networks security | K3 |
| CO3 | Ability to analyze and determine for any organization the database security requirements and appropriate solutions | K4 |
| CO4 | Determined IP Security Issues and solve real time problems. | K5 |
| CO5 | Build wireless Development Strategies in real time issues | K6 |

**#based on suggested Revised BTL**

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 1 | | 2 | 3 | | | | | | | | |
| CO2 | | 1 | | 2 | | 3 | | | | | | |
| CO3 | 1 | | | 3 | 2 | | | | | | | |
| CO4 | | 2 | 1 | | | 3 | | | | | | |
| CO5 | 1 | | 2 | | 3 | | | | | | | |

**(Please fill the above with Levels of Correlation, viz., L-1, M-2, H-3)**

**SYLLABUS:**

**UNIT-I: Introduction to Wireless**: History of Wireless Technologies, History of Wireless Security, State of the Wireless Security Industry, 2001 **Wireless Threats:** Uncontrolled Terrain, Communications Jamming, DoS Jamming, Injections and Modifications of Data, Man-in-the-Middle (MITM) Attack, Rogue Client, Rogue Network Access Points, Attacker Equipment,

**UNIT-II: Introduction to Wireless Security Protocols and Cryptography:** Recovery the FUD, OSI Model, OSI Simplified, Internet Model, Wireless LAN Security Protocols, Cryptography, SSL/TLS, Secure Shell Protocols, Terminal

Access and File Transfer, Port Forwarding a Word of Caution, Man-in-the-Middle of SSL/TLS and SSH, WTLS, WEP,802.1x, IP Security

**UNIT-III: Security Considerations to Wireless Devices:** Wireless Device Security Issues, Physical Security, Information Leakage, Device Security Features, Application Security, Detailed Device Analysis, Laptops, Personal Digital Assistants (PDAS), Wireless Infrastructure **Wireless Technologies and Applications:** Introduction to Cellular Networks- FDMA, TDMA, CDMA, Spread Spectrum Primer, Analogy, TDMA Vs CDMA, PDC, Security Threats

**UNIT-IV: Introduction to Wireless Data Networks:** Cellular Digital Packet Data (CDPD), CDPD Architecture, CDPD Security, Mobitex- Mobitex Architecture, Mobitex Security Architecture, Security Issues, Gateway, Security Model **Wireless Standards and Technologies:** Current and Future Technologies- Infrared, Radio, Spread Spectrum, OFDM, Current and Future Standards- IEEE 802 Standards, ETSI, Home RF, Ultra-wide band Radio (UWB)

**UNIT-V: Wireless Deployment Strategies:** Implementing Wireless LAN's- Security Considerations Common Wireless Network Applications, Enterprise Campus Designs, Wireless IST Design, Retail and Manufacturing Design, Small Office/Home Office Design (SOHO)

**TEXT BOOKS:**

1. Wireless Security, Merritt Maxim and David Pollino, Osborne/McGraw Hill, New Delhi, 2005
2. Wireless Security Models: Threats and Solutions, Nichols and Lekka, Tata McGraw Hill, New Delhi 2006

**REFERENCE BOOKS:**

1. Behrouz A.Forouzan, ―Cryptography & Network Security‖, Tata McGraw Hill, India, New Delhi, 2009
2. William Stallings, ―Cryptography and Network Security, Prentice Hall, New Delhi, 2006
3. Bruce Schneier, "Applied Cryptography", John Wiley & Sons, New York, 2004

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**IV B. Tech I Sem-R20**

**ETHICAL HACKING**
**Code: R20PE4103**

**COURSE OBJECTIVES:**

- The aim of the course is to introduce the methodologies and framework of ethical hacking for enhancing the security.
- The course includes-Impacts of Hacking; Types of Hackers; Information Security Models, Information Security Program, Business Perspective, Planning a Controlled Attack
- Framework of Steps (Reconnaissance, Enumeration, Vulnerability Analysis, Exploitation, Deliverable and Integration)

**COURSE OUTCOMES**: At the end of the course, student will be able to

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Explain the concepts related to hacking, ports and protocols, pen testing and virtualization | **K2** |
| **CO2** | Determine the applicable footprinting techniques and scanning methods | **K4** |
| **CO3** | Explain the process of system hacking and Explain the concepts Trojans, backdoors, worms and virus and it's countermeasures | **K2** |
| **CO4** | Demonstrate systematic understanding of the concepts of Sniffing and Social Engineering and it's attacks | **K2** |
| **CO5** | Determine the applicable methods of cryptography, stegnography and Vulnerability Assessment | **K4** |

***#Based on suggested Revised BTL***

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 2 | 3 | 2 | | 2 | | | | | | | 2 |
| **CO2** | | 3 | 1 | 2 | | 2 | | | | | 2 | 2 |
| **CO3** | 1 | 2 | 2 | 2 | | | 2 | | | | | 3 |
| **CO4** | | 2 | 2 | | 3 | 2 | | | | | 3 | |
| **CO5** | 2 | 3 | 3 | 3 | 3 | 3 | 3 | | | | | 3 |

**(Please fill the above with Levels of Correlation, viz., L-1, M-2, H-3)**

**SYLLABUS:**

**UNIT-I: Introduction to Hacking:** Hacking, Types and phases of hacking, **Introduction to Ports & Protocols**: Ports, Protocols, Primary Network Types, **Virtualization & Introduction to Kali Linux:** Virtualization, Virtualization software, supported platforms, **Introduction to Penetration Testing:** Penetration test, Categories and Types of Penetration tests, Structure of Penetration Test Report.

**UNIT-II: Footprinting:** Footprinting, Types, Using ping and ns Lookup commands in Windows command line, **Scanning:** Scanning, Basics of Scanning, Basic Techniques of Scanning, Enumerating DNS using dns enum, Performing flag scan using hping3.

**UNIT-III: Hacking into System:** System Hacking, Password Cracking, Default password databases, Manual and Automated Password Cracking, Process of System Hacking, Using Keyloggers, **Trojans & Backdoors:** Trojans, Working of Trojan, Infection Techniques, Attack, Lifecycle and Classification of Virus, Worms, Virus Construction Kit.

**UNIT-IV: Sniffing, Packet Analysis & Session Hijacking:** Sniffing, Packet Analysis, Types of Sniffing, Active and Passive Sniffing Techniques, Session Hijacking, **Social Engineering:** Social Engineering, Process, Identity Theft, Human and Computer Based Social Engineering Techniques, Phishing Process, Types of Phishing Attacks, Social Engineering Toolkit (SET)

**UNIT-V: Cryptography:** Cryptography, Digital Signature, Hash Functions, **Steganography:** Steganography Process, watermarking, Steganography Methods and Attacks, Steganography tools, **Vulnerability Assessment:** Vulnerability, The Open Web Application Security Project (OWASP), Prevention, Damn Vulnerable Web Application (DVWA), installation and testing of DVWA

**TEXT BOOKS:**

1. Hacking: Be a Hacker with Ethics, Harsh Bothra, Khanna Publications, 2019
2. Ethical Hacking and Penetration Testing Guide, Rafay Baloch, 2014

**REFERENCE BOOKS:**

1. Kali Linux Wireless Penetration Testing Beginner's Guide, Vivek Ramachandran, Cameron Buchanan, Packt Publishing, 2015
2. SQL Injection Attacks and Defense, 1st Edition, Justin Clarke-Salt, Syngress Publication
3. Mastering Modern Web Penetration Testing, Prakhar Prasad, Packt Publishing, October 2016

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**IV B. Tech I Sem-R20**

**SECURE CODING TECHNIQUES**
**Code: R20OE4101**

**COURSE OBJECTIVE:** This course aims to provide an understanding of the various security attacks and knowledge to recognize and remove common coding errors that lead to vulnerabilities. It gives an outline of the techniques for developing a secure application.

**COURSE OUTCOMES**: At the end of the course, student will be able to

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Explain why input validation and data sanitization is necessary in the face of Software security problem. | **K2** |
| **CO2** | Classify common input validation errors, and write correct input validation code. | **K4** |
| **CO3** | Demonstrate using a high-level programming language how to prevent a race condition from occurring and how to handle an exception. | **K2** |
| **CO4** | Demonstrate the identification and handling of Web Applications error conditions | **K2** |
| **CO5** | Conduct a security verification and assessment (static and dynamic) of a software application. | **K4** |

***#Based on suggested Revised BTL***

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 2 | 3 | 2 | | 2 | | | | | | | 2 |
| **CO2** | | 3 | 1 | 2 | | 2 | | | | | 2 | 2 |
| **CO3** | 1 | 2 | 2 | 2 | | | 2 | | | | | 3 |
| **CO4** | | 2 | 2 | | 3 | 2 | | | | | 3 | |
| **CO5** | 2 | 3 | 3 | 3 | 3 | 3 | 3 | | | | | 3 |

**(Please fill the above with Levels of Correlation, viz., L-1, M-2, H-3)**

**SYLLABUS:**

**UNIT I: Software Security Problem:** Defensive Problem, Security Features, Quality Fallacy, Classifying Vulnerabilities, Introduction to static Analysis: Capabilities and Limitations of Static Analysis, Solving Problems with Static Analysis, Little Theory, a Little Reality.

**UNIT II: Code Review Process:** Performing Code Review, Adding Security Review to Existing Development Process, Static Analysis Metric, Static Analysis Internals: Building Model, Analysis Algorithms, Rules, Reporting Results.

**UNIT III: Handling Inputs:** Validates, Preventing Metacharacter Vulnerabilities, Buffer Overflow: Introduction, Strings, Bride of Buffer Overflow: Integers, Runtime Protection, Errors and Exceptions: Handling Errors with Return Codes, Managing Exceptions, Preventing Resource Leaks, Logging and Debugging,

**UNIT IV: Web Applications:** Input and Output Validation, HTTP Considerations, Maintaining Session State, Struts Validator, XML and Web Services: Working with XML, Using Web Services

**UNIT V: Privacy and Security:** Privacy and Regulation, Outbound Password, Random Numbers, Cryptography, Secrets in Memory, Privileged Programs: Implications of Privilege, Managing privilege, Privilege Escalation Attacks.

**Textbook:**

1. Chess, B and West, J, Secure Programming with Static Analysis, Addison-Wesley, 2007.

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**IV B. Tech I Sem-R20**

**COMPUTATIONAL THINIKING**
**Code: R20OE4102**

**COURSE OBJECTIVES:**

- describe the skills that are involved in computational thinking
- define and use the concepts of abstraction as modelling and abstraction as encapsulation
- understand the distinctive nature of computational thinking, when compared with engineering and mathematical thinking
- To be aware of a range of applications of computational thinking in different disciplines.

**COURSE OUTCOMES**: At the end of the course, student will be able to

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Describe a systematic process for solving problems and making decisions. | **K5** |
| **CO2** | Evaluate alternative solutions to computational problems | **K3** |
| **CO3** | Explain the operation of a variety of algorithms | **K2** |
| **CO4** | Develop algorithms to implement chosen solutions for Data and Algorithemic thinking | **K4** |
| **CO5** | Evaluate the costs and benefits of the use of computing technology in automating processes | **K3** |

***#Based on suggested Revised BTL***

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 2 | 3 | 2 | | 2 | | | | | | | 2 |
| **CO2** | | 3 | 1 | 2 | | 2 | | | | | 2 | 2 |
| **CO3** | 1 | 2 | 2 | 2 | | | 2 | | | | | 3 |
| **CO4** | | 2 | 2 | | 3 | 2 | | | | | 3 | |
| **CO5** | 2 | 3 | 3 | 3 | 3 | 3 | 3 | | | | | 3 |

**(Please fill the above with Levels of Correlation, viz., L-1, M-2, H-3)**

**SYLLABUS:**

**UNIT I:** Introduction, Computers, Computational Thinking, Abacus to Machine, The First Software, The First Modern Computer, Moore's Law, **Real-World Information Becomes Computable Data,** Information and Data, Converting Information into Data, Data Capacity, Data Compression.

**UNIT II: Logic:** Boolean Logic, Applications of Propositional Logic, **Solving Problems,** Logical Reasoning, Decomposition: Software Design, **Decomposition:** Other Uses, Abstraction: Class Diagrams, Abstraction: Use Case Diagrams.

**UNIT III: Algorithmic Thinking:** Algorithms, Software and Programming, Languages, Actions, **Modeling Solutions:** Activity Diagrams, Selection in Activity Diagrams, Repetition in Activity Diagrams, Control Abstraction in Activity Diagrams, States and State Diagrams, Behavior in State Diagrams.

**UNIT IV: Data Organization:** Names, Lists, Graphs, Hierarchies, **Algorithmic Thinking:** Von Neumann Architecture, Spreadsheets, Text Processing, Patterns, Computer Errors, Software Correctness, Verification, Software Testing, White Box Testing, Black Box Testing with Equivalence Partitioning, Boundary Value Analysis.

**UNIT V: Concurrent Activity:** Parallelism or Concurrency, Scheduling, Sorting Networks, **Information Security:** Security, Foundations, Common Forms of Cybercrime, Secure- Step 1: Authenticate, Secure- Step 2: Authorization, All a Matter of Risk.

**TEXT BOOK:**

1. David Riley, Kenny A. Hunt, Computational Thinking for the Modern Problem Solver, 2014.

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**IV B. Tech I Sem-R20**

**Universal Human Values 2: Understanding Harmony**
**R20HSS4101**

**Human Values Courses**

This course also discusses their role in their family. It, very briefly, touches issues related to their role in the society and the nature, which needs to be discussed at length in one more semester forwhich the foundation course named as “H-102 Universal Human Values 2: Understanding Harmony”is designed which may be covered in their III or IV semester. During the Induction Program, students would get an initial exposure to human values through Universal Human Values – I. This exposure is to be augmented by this compulsory full semester foundation course.

**Universal Human Values 2: Understanding Harmony**

Course code: HSMC (H-102)
Credits: L-T-P-C 2-1-0-3 or 2L:1T:0P 3 credits
Pre-requisites: None. Universal Human Values 1 (desirable)

**1. OBJECTIVE:**

**The objective of the course is four fold:**

1. Development of a holistic perspective based on self-exploration about themselves (human being),family, society and nature/existence.
2. Understanding (or developing clarity) of the harmony in the human being, family, society and nature/existence
3. Strengthening of self-reflection.
4. Development of commitment and courage to act.

**2. COURSE TOPICS:**

**The course has 28 lectures and 14 practice sessions in 5 modules:**

**Module 1: Course Introduction - Need, Basic Guidelines, Content and Process for Value Education**

1. Purpose and motivation for the course, recapitulation from Universal Human Values-I
2. Self-Exploration–what is it? - Its content and process; ‘Natural Acceptance’ and Experiential Validation- as the process for self-exploration
3. Continuous Happiness and Prosperity- A look at basic Human Aspirations
4. Right understanding, Relationship and Physical Facility- the basic requirements for fulfilment of aspirations of every human being with their correct priority
5. Understanding Happiness and Prosperity correctly- A critical appraisal of the current scenario
6. Method to fulfil the above human aspirations: understanding and living in harmony at various levels.

Include practice sessions to discuss natural acceptance in human being as the innate acceptance for living with responsibility (living in relationship, harmony and co-existence) rather than asarbitrariness in choice based on liking-disliking

**Module 2: Understanding Harmony in the Human Being - Harmony in Myself!**

7. Understanding human being as a co-existence of the sentient 'I' and the material 'Body'
8. Understanding the needs of Self ('I') and 'Body' - happiness and physical facility
9. Understanding the Body as an instrument of 'I' (I being the doer, seer and enjoyer)
10. Understanding the characteristics and activities of 'I' and harmony in 'I'
11. Understanding the harmony of I with the Body: Sanyam and Health; correct appraisal of Physical needs, meaning of Prosperity in detail
12. Programs to ensureSanyam and Health.

Include practice sessions to discuss the role others have played in making material goods available to me. Identifying from one's own life. Differentiate between prosperity and accumulation. Discuss program for ensuring health vs dealing with disease

**Module 3: Understanding Harmony in the Family and Society- Harmony in Human-Human Relationship**

13. Understanding values in human-human relationship; meaning of Justice (nine universal values in relationships) and program for its fulfilment to ensure mutual happiness; Trust and Respect as the foundational values of relationship
14. Understanding the meaning of Trust; Difference between intention and competence
15. Understanding the meaning of Respect, Difference between respect and differentiation; the other salient values in relationship
16. Understanding the harmony in the society (society being an extension of family): Resolution, Prosperity, fearlessness (trust) and co-existence as comprehensive Human Goals
17. Visualizing a universal harmonious order in society- Undivided Society, Universal Order- from family to world family.

Include practice sessions to reflect on relationships in family, hostel and institute as extended family, real life examples, teacher-student relationship, goal of education etc. Gratitude as a universal value in relationships. Discuss with scenarios. Elicit examples from students' lives

**Module 4: Understanding Harmony in the Nature and Existence - Whole existence as Coexistence**

18. Understanding the harmony in the Nature
19. Interconnectedness and mutual fulfilment among the four orders of nature-recyclability and self-regulation in nature
20. Understanding Existence as Co-existence of mutually interacting units in all-pervasive space

21. Holistic perception of harmony at all levels of existence.
Include practice sessions to discuss human being as cause of imbalance in nature (film "Home" can be used), pollution, depletion of resources and role of technology etc.

**Module 5: Implications of the above Holistic Understanding of Harmony on Professional Ethics**

22. Natural acceptance of human values
23. Definitiveness of Ethical Human Conduct
24. Basis for Humanistic Education, Humanistic Constitution and Humanistic Universal Order
25. Competence in professional ethics: a. Ability to utilize the professional competence for augmenting universal human order b. Ability to identify the scope and characteristics of people- friendly and eco-friendly production systems, c. Ability to identify and develop appropriate technologies and management patterns for above production systems.
26. Case studies of typical holistic technologies, management models and production systems
27. Strategy for transition from the present state to Universal Human Order: a. At the level of individual: as socially and ecologically responsible engineers, technologists and managers b. At the level of society: as mutually enriching institutions and organizations
28. Sum up.

Include practice Exercises and Case Studies will be taken up in Practice (tutorial) Sessions eg. To discuss the conduct as an engineer or scientist etc.

**3. READINGS:**

3.1 Text Book

1. Human Values and Professional Ethics by R R Gaur, R Sangal, G P Bagaria, Excel Books,New Delhi, 2010

**3.2 Reference Books**

1. Jeevan Vidya: Ek Parichaya, A Nagaraj, Jeevan Vidya Prakashan, Amarkantak, 1999.
2. Human Values, A.N. Tripathi, New Age Intl. Publishers, New Delhi, 2004.
3. The Story of Stuff (Book).
4. The Story of My Experiments with Truth - by Mohandas Karamchand Gandhi
5. Small is Beautiful - E. F Schumacher.
6. Slow is Beautiful - Cecile Andrews
7. Economy of Permanence - J C Kumarappa
8. Bharat Mein Angreji Raj - PanditSunderlal
9. Rediscovering India - by Dharampal
10. Hind Swaraj or Indian Home Rule - by Mohandas K. Gandhi
11. India Wins Freedom - Maulana Abdul Kalam Azad
12. Vivekananda - Romain Rolland (English)
13. Gandhi - Romain Rolland (English)

**4. MODE OF CONDUCT (L-T-P-C 2-1-0-3 or 2L:1T:0P 3 credits)**

Lectures hours are to be used for interactive discussion, placing the proposals about the topics at hand and motivating students to reflect, explore and verify them.

Tutorial hours are to be used for practice sessions.

While analysing and discussing the topic, the faculty mentor's role is in pointing to essentialelements to help in sorting them out from the surface elements. In other words, help the studentsexplore the important or critical elements.

In the discussions, particularly during practice sessions (tutorials), the mentor encourages the student to connect with one's own self and do self-observation, self-reflection and self-exploration. Scenarios may be used to initiate discussion. The student is encouraged to take up" ordinary" situations rather than" extra-ordinary" situations. Such observations and their analyses are shared and discussed with other students and faculty mentor, in a group sitting.

Tutorials (experiments or practical) are important for the course. The difference is that the laboratory is everyday life, and practical are how you behave and work in real life. Depending on the nature of topics, worksheets, home assignment and/or activity are included. The practice sessions (tutorials)
would also provide support to a student in performing actions commensurate to his/her beliefs. It is intended that this would lead to development of commitment, namely behaving and working basedon basic humanvalues.

It is recommended that this content be placed before the student as it is, in the form of a basic foundation course, without including anything else or excluding any part of this content. Additional content may be offered in separate, higher courses.

This course is to be taught by faculty from every teaching department, including HSS faculty. Teacher preparation with a minimum exposure to at least one 8-day FDP on Universal Human Values is deemed essential.

**5. ASSESSMENT:**

This is a compulsory credit course. The assessment is to provide a fair state of development of thestudent, so participation in classroom discussions, self-assessment, peer assessment etc. will be used in evaluation.

Example:

Assessment by faculty mentor: 10 marks
Self-assessment: 10 marks
Assessment by peers: 10 marks
Socially relevant project/Group Activities/Assignments: 20 marks
Semester End Examination: 50 marks

The overall pass percentage is 40%. In case the student fails, he/she must repeat the course.

**6. OUTCOME OF THE COURSE:**

By the end of the course, students are expected to become more aware of themselves, and their surroundings (family, society, nature); they would become more responsible in life, and in handling problems with sustainable solutions, while keeping human relationships and human nature in mind.

They would have better critical ability. They would also become sensitive to their commitment towards what they have understood (human values, human relationship and human society). It is hoped that they would be able to apply what they have learnt to their own self in different day-to-day settings in real life, at least a beginning would be made in this direction.

This is only an introductory foundational input. It would be desirable to follow it up by

a) faculty-student or mentor-mentee programs throughout their time with the institution

b) Higher level courses on human values in every aspect of living. E.g. as a professional

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**IV B. Tech I Sem-R20**

**DEEP LEARNING**
**Code: R20SO4101**

**COURSE OBJECTIVES:** The main objective of the course is

- To gain knowledge on the concepts of deep learning using python.
- To acquire the knowledge on various application areas of computer vision, natural language processing, structured data, time series, audio data and video data.

**COURSE OUTCOMES :** At the end of the course, student will be able to

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Develop various applications using computer vision. | **K6** |
| **CO2** | Design various applications using natural language processing. | **K6** |
| **CO3** | Develop various applications using structured data. | **K6** |
| **CO4** | Develop different applications using time series analysis. | **K6** |
| **CO5** | Design various applications using audio and video data. | **K6** |

***#Based on suggested Revised BTL***

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 1 | 2 | 3 | | 3 | 2 | | | | 1 | | |
| CO2 | 1 | 2 | 1 | 2 | 1 | 3 | | | | | 1 | |
| CO3 | 1 | 2 | 3 | | 3 | | | | | | | 1 |
| CO4 | 2 | 1 | 2 | | 1 | | 3 | | | | | |
| CO5 | 2 | 3 | 1 | | 2 | | | | | | | 1 |

**(Please fill the above with Levels of Correlation, viz., L-1, M-2, H-3)**

**Implement the following programs using python programming**

**<u>List of Experiments:</u>**

**Experiment 1:** Write a program to implement hand written digit recognition with MNIST dataset.

**Experiment 2:** Write a program to implement image captioning using Flickr 8K dataset.

**Experiment 3:** Write a program to implement bi-directional LSTM with IMDB dataset.

**Experiment 4:** Write a program to implement sentiment analysis using RMN.

**Experiment 5:**Write a program to implement time series analysis using LSTM for weather prediction.

**Experiment 6:** Write a program to implement time series analysis for house sales prediction.

**Experiment 7:** Write a program to implement speech emotion detection.

**Experiment 8:** Write a program to implement speaker recognition.

**Experiment 9:** Write a program to implement movie review analysis with IMDB dataset.

**Experiment 10:** Write a program to implement short text emotion detection using multiclass SVM.

**Experiment 11:** Write a program to implement cyber threat detection using deep learning.

**Experiment 12:** Write a program to implement image recognition with CNN using cifar10 dataset.

**Web References:**

1. https://keras.io/examples/
2. https://stackabuse.com/image-recognition-in-python-with-tensorflow-and-keras/

Department of Computer Science & Engineering
University College of Engineering, JNT University Kakinada

# OPEN ELECTIVES TO BE OFFERED BY CSE FOR OTHER BRANCHES

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

***Open Electives to be offered by CSE for Other Branches:***

| **Open Elective-I:** | **Open Elective-II:** |
|---|---|
| 1. Data Structures<br>2. Java Programming<br>3. Data Base Management Systems<br>4. Computer Graphics<br>5. C++ Programming<br>6. Advanced Unix programming | 1. Operating Systems<br>2. Python Programming<br>3. Web Technologies<br>4. Soft Computing<br>5. Distributed Computing<br>6. AI and ML for Robotics |
| **Open Elective-III:**<br>1. Big Data Analytics<br>2. AI Tools & Techniques<br>3. Image Processing<br>4. Information Security<br>5. Mobile Application Development<br>6. Cloud computing | **Open Elective-IV:**<br>1. Cyber Security<br>2. Deep Learning<br>3. Data Science<br>4. Block Chain Technologies<br>5. Game Theory<br>6. Internet of Things |

**Department of Computer Science & Engineering**
**University College of Engineering, JNT University Kakinada**

**Open Elective-I:**

## DATA STRUCTURES

**Course Objectives:**

- Solve problems using data structures such as linear lists, stacks, queues, hash tables
- Be familiar with advanced data structures such as balanced search trees, AVL Trees, and B Trees.

**Course Outcomes:** By the end of the course student will be able to understand

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| CO1 | Select appropriate data structures as applied to specified problem definition | K5 |
| CO2 | Summarize and understand the practical applications of several advanced techniques like Hashing and Analyzing and Implement appropriate sorting/searching technique for given problems | K2 |
| CO3 | Demonstrate the operations such as Insertion, Deletion and Search on Data structures like Binary Search Tree and solve the problems | K3 |
| CO4 | Demonstrate the operations such as Insertion, Deletion and Search on Advanced Data structures like Heaps, AVL trees and B Trees. | K3 |
| CO5 | Comparisons of trees like Red Black trees and B-Trees etc. and priority queue operations. | K4 |

**#based on suggested Revised BTL**

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 |
|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 3 | 2 | 2 | | | | | | | |
| CO2 | 2 | | 3 | | | | | 3 | | |
| CO3 | 1 | 2 | 3 | | 3 | | | | | |
| CO4 | | | | 2 | 3 | 1 | | 3 | | |
| CO5 | 2 | | | | | | | 2 | | |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**SYLLABUS:**

**UNIT- I: Introduction to Data Structures:** Abstract Data Types (ADTs), The List ADT: Simple Array Implementation of Lists, Simple Linked Lists, Doubly Linked Lists, Circularly Linked Lists. The Stack ADT: The Stack Model, Implementation of Stacks, Applications of Stack. The Queue ADT: Queue Model, Array Implementation of Queues, Application of Queues.Stacks and Queue implementation using linked list.

**UNIT-II: Searching**: List Searches, Linear and Binary Search Methods.
**Sorting:** Selection Sort, Insertion Sort, Quick Sort, Merge Sort, Heap sort.
**Hashing:** Hash Function, Separate Chaining, Collision Resolution-Separate Chaining.

**UNIT- III: Trees:** Binary Trees- Implementation, Expression Trees. Binary Search Trees- find, findMin and findMax, insert, delete operations.

**UNIT- IV: Trees:** AVL Trees- Single and Double Rotation, Operations.
**B-Tree:** searching, insertion, deletion

**UNIT –V:**
**Trees:** Introduction to Red-Black, splay trees and Comparison of Search Trees
**Priority Queues:** Priority Queue Models, Simple Implementations.

**TEXT BOOKS:**

1. Data Structures and Algorithm Analysis, 4th Edition, Mark Allen Weiss,Pearson.
2. Data Structures: A PseudoCode Approach with C, 2nd Edition, Richard F.Gilberg, &Behrouz A. Forouzon, Cengage.

**REFERENCES BOOKS**:

1. Data Structures, Algorithms and Applications in java, 2/e, Sartaj Sahni,University Press.
2. Data Structures using C, 2/e, Reema Thareja

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### JAVA PROGRAMMING

**Course Objective:** Implementing programs for user interface and application development using core java principles

**Course Outcomes:** By the end of the course student will be able to understand

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Discuss and understand java programming constructs, Control structures | **K2** |
| **CO2** | Illustrate and experiment Object Oriented Concepts like classes, objects | **K3** |
| **CO3** | Apply Object Oriented Constructs such as Inheritance, interfaces, and exception handling | **K3** |
| **CO4** | Construct applications using multithreading and I/O | **K3** |
| **CO5** | Develop Dynamic User Interfaces using applets and Event Handling in java | **K3** |

**#based on suggested Revised BTL**

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | | 1 | 2 | 1 | 3 | | | | | 1 | 1 | |
| **CO2** | 1 | 2 | 3 | 1 | 2 | | | | 1 | 2 | 1 | 3 |
| **CO3** | 1 | 2 | 2 | 3 | 1 | 2 | | | 1 | 2 | 1 | 1 |
| **CO4** | 1 | | 3 | | 1 | 1 | | | 2 | 2 | 1 | |
| **CO5** | | 2 | 2 | 3 | 1 | | | | 1 | | | |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**SYLLABUS:**

**UNIT I: Objective:** Focus on object oriented concepts and java program structure and its installation, Introduction to OOP Introduction, Need of Object Oriented Programming, Principles of Object Oriented Languages, Procedural languages Vs OOP, Applications of OOP, History of JAVA, Java Virtual Machine, Java Features.

**UNIT II: Objective:** Comprehension of java programming constructs, control structures in Java Programming Constructs Variables , Primitive Datatypes, Identifiers- Naming Conventions, Keywords, Literals, Operators-Binary, Unary and ternary, Expressions, Precedence rules and Associativity, Primitive Type Conversion and Casting, Flow of control Branching, Conditional, loops.,

**UNIT III:** Classes and Objects- classes, Objects, Creating Objects, Methods, constructors, Constructor overloading, cleaning up unused objects-Garbage collector, Class variable and Methods-Static keyword, this keyword, Arrays, Command line arguments **Interfaces and exception handling Inheritance:** Types of Inheritance, Deriving classes using extends keyword, Method overloading, super keyword, final keyword, Abstract class Interfaces,

**UNIT IV: Objective**: Understanding of Thread concepts and I/O in Java MultiThreading: java.lang.Thread, The main Thread, Creation of new threads, Thread priority, Multithreading- Using isAlive() and join(), Syncronization, suspending and Resuming threads, Communication between Threads.

**UNIT V: Objective**: Being able to build dynamic user interfaces using applets and Event handling in java Swing: Introduction , javax.swing package , JFrame, JApplet, JPanel, Components in swings, Layout Managers, JList and JScroll Pane, Split Pane, JTabbedPane, Dialog Box.

**TEXT BOOKS:**

1. The Complete Refernce Java, 8ed, Herbert Schildt, TMH
2. Programming in JAVA, Sachin Malhotra, Saurabh choudhary, Oxford.

**REFERENCE BOOKS:**

1. JAVA Programming, K.Rajkumar.Pearson
2. Core JAVA, Black Book, Nageswara Rao, Wiley, Dream Tech
3. JAVA for Beginners, 4e, Joyce Farrell, Ankit R. Bhavsar, Cengage Learning.
4. Object oriented programming with JAVA, Essentials and Applications, Raj Kumar Bhuyya, Selvi, Chu TMH
5. Introduction to Java programming, 7th ed, Y Daniel Liang, Pearson
   Core JAVA for Beginners, Rashmi Kanta Das, Vikas.
6. Object Oriented Programming through JAVA , P Radha Krishna , University Press

## DATA BASE MANAGEMENT SYSTEMS

**Pre-requisite:** Basic Knowledge on Data Structures

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

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | **Illustrate** the concept of databases, database management systems, database languages, database structures and their work | **K2** |
| **CO2** | **Apply** ER modeling and Relational modeling for designing simple databases. | **K3** |
| **CO3** | **Summarize** the concepts related to relational model and SQL and **Write** database queries using relational algebra and structured query language. | **K2** |
| **CO4** | **Design and develop** databases from the real world by applying the concepts of Normalization. | **K6** |
| **CO5** | **Outline** the issues associated with Transaction Management and Recovery, Tree Structured Indexing | **K2** |

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 1 | 1 | 2 | 3 | 1 | | | | 2 | 1 | 1 | |
| **CO2** | 1 | 2 | 2 | | 2 | | | | 3 | 2 | 2 | 1 |
| **CO3** | 1 | 2 | 2 | 3 | 2 | 1 | | | | 2 | 1 | 1 |
| **CO4** | | | 3 | 2 | 1 | | | | | 1 | 2 | 1 |
| **CO5** | 1 | 3 | 2 | 1 | 1 | 1 | | | 2 | 3 | 1 | 1 |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**SYLLABUS:**

**UNIT – I: Overview of Database System**: Database System Applications, Purpose of Database Systems, View of Data, Database Languages, Relational Databases, Database Design, Data Storage and Querying, Transaction Management, Database Architecture, Data Mining and Informational Retrieval, Specialty Databases, Database Users and Administrators, History of Database Systems. **[Text Book -2]**

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**UNIT –II: Introduction to Database Design:** Database Design and ER Diagrams, Entities, Attributes and Entity Sets, Relationships and Relationship Sets, Additional Features of the ER Model, Conceptual Design with the ER Model, Extended ER features **[Text Book -1]**

**UNIT –III: Relational Model:** Introduction to the Relational Model, Integrity Constraints over Relations, Enforcing Integrity Constraints, Querying Relational Data, Logical Database Design: ER to Relational, Introduction to Views, Destroying/Altering Tables and Views **[Text Book -1]]**

**UNIT –IV: SQL: Queries, Constraints, Triggers:** The Form of a Basic SQL Query, UNION, INTERSECT and EXCEPT, Nested Queries, Aggregate Operators, Null Values, Complex Integrity Constraints in SQL, Triggers, Exceptions, Procedures, Functions **[Text Book -1]**

**UNIT –V: Normal Forms:** Introduction to Schema Refinement, Functional Dependencies, Reasoning about FDs, Normal Forms, Properties of Decompositions, Normalization. **[Text Book -1]**

**TEXT BOOKS**:
1. Data base Management Systems, 3/e, Raghurama Krishnan, Johannes Gehrke, Mc Graw-Hill
2. Data base System Concepts, 6/e, Abraham Silberschatz, Henry F. Korth, S. Sudarshan, Mc Graw-Hill

**REFERENCE BOOKS**:
1. Database Systems, 6/e Ramez Elmasri, Shamkant B. Navathe, Pearson
2. Introduction to Database Systems, 8/e, C J Date, Pearson
3. Database Systems, 9/e, Carlos Coronel, Steven Morris, Peter Rob, Cengage

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### COMPUTER GRAPHICS

**Course Objectives:**

- To develop, design and implement two and three dimensional graphical structures
- To enable students to acquire knowledge Multimedia compression and animations
- To learn Creation, Management and Transmission of Multimedia objects.

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

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Acquire the basics of computer graphics, different graphics systems and applications of computer graphics with various algorithms for line, circle and ellipse drawing objects for 2D transformations | **K3** |
| **CO2** | Explain projections and visible surface detection techniques for display of 3D scene on 2D screen | **K5** |
| **CO3** | Develop scene with basic graphic primitive algorithms using OPENGL programming. | **K3** |
| **CO4** | Know and be able to Explain selected among models for lighting/shading: Color, ambient light; distant and light with sources; Phong reflection model; and shading (flat, smooth, Gourand, Phong).. | **K5** |
| **CO5** | Illustrate able to create the general software architecture of programs that use 3D object sets with computer graphics | **K3** |

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 3 | 2 | 1 | | | | | | | | 1 | |
| **CO2** | | 3 | | | 1 | 2 | | | | | 1 | |
| **CO3** | | | 3 | 3 | | | | 1 | 1 | | | |
| **CO4** | 3 | | 2 | 2 | | 1 | | | | | | 1 |
| **CO5** | | | | 3 | 1 | 2 | | | | | 2 | 1 |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**SYLLABUS:**

**UNIT-I: 2D Primitives :** Output primitives – Line, Circle and Ellipse drawing algorithms - Attributes of output primitives – Two dimensional Geometric transformations - Two dimensional viewing – Line, Polygon, Curve clipping algorithms

**UNIT-II: 3D Concepts:** Parallel and Perspective projections - Three dimensional object representation – Polygons, Curved lines, Splines, Quadric Surfaces, -

Visualization of data sets - 3D transformations – Viewing -Visible surface identification.

**UNIT-III: Graphics Programming :** Color Models – RGB, YIQ, CMY, HSV – Animations – General Computer Animation, Raster, Keyframe - Graphics programming using OPENGL – Basic graphics primitives – Drawing three dimensional objects.

**UNIT- IV: Rendering :** Introduction to Shading models – Flat and Smooth shading – Adding texture to faces – Adding shadows of objects – Building a camera in a program – Creating shaded objects– Rendering texture – Drawing Shadows.

**UNIT- V: Fractals:** Fractals and Self similarity – Peano curves – Creating image by iterated functions – Mandelbrot sets – Julia Sets – Random Fractals

**TEXT BOOKS:**

1. Donald Hearn, Pauline Baker, Computer Graphics – C Version, second edition, Pearson Education,2004.
2. F.S. Hill, Computer Graphics using OPENGL, Second edition, Pearson Education, 2003.

**REFERENCE BOOKS:**

1. James D. Foley, Andries Van Dam, Steven K. Feiner, John F. Hughes, Computer Graphics- Principles and practice, Second Edition in C, Pearson Education, 2007.

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### C++ PROGRAMMING

**Course Objectives:**

- This course is designed to provide a comprehensive study of the C programming language. It stresses the strengths of C, which provide students with the means of writing efficient, maintainable and portable code. The nature of C language is emphasized in the wide variety of examples and applications.
- To learn and acquire art of computer programming.
- Programming language for solving a problem.

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

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Demonstrate basics of object oriented mode, differences between conventional and oops programming, the top-down and bottom-up approach I/O streams in C++ | **K2** |
| **CO2** | Explain Write, compile and debug programs and Use different data types, classes, objects and member functions in C++ language. | **K5** |
| **CO3** | Make use of Basic concept in C++ programming, Operators, control structures, functions, overloading, and recursion. | **K5** |
| **CO4** | Build dynamic memory management techniques using pointers, constructors, destructors, virtual functions. | **K5** |
| **CO5** | Classify inheritance with the understanding of early and late binding, usage of exception handling, generic programming. | **K3** |

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 2 | 1 | 2 | 3 | 1 | | | | 2 | 1 | 1 | |
| **CO2** | 1 | | 3 | | 2 | | | | 3 | 2 | 2 | 1 |
| **CO3** | 1 | 2 | 2 | 3 | 2 | 1 | | | | 2 | 1 | |
| **CO4** | 2 | | 3 | 2 | | | | | | 1 | | 1 |
| **CO5** | 1 | 3 | 2 | | | 1 | | | 2 | 3 | 2 | 1 |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**Syllabus:**

**UNIT-I:** Introduction to C++ Difference between C and C++- Evolution of C++- The Object Oriented Technology Disadvantage of Conventional Programming- Key Concepts of Object Oriented Programming- Advantage of OOP- Object Oriented Language.

**UNIT-II:** Classes and Objects &Constructors and Destructor Classes in C++- Declaring Objects-Defining Member Function-Overloading Member Function-

Nested class, Constructors and Destructors, Introduction- Constructors and Destructor- Characteristics of Constructor and Destructor-Application with Constructor- Constructor with Arguments.

**UNIT-III:** Operator Overloading and Type Conversion & Inheritance The Keyword Operator- Overloading Unary Operator- Operator Return Type Overloading Assignment Operator (=)- Rules for Overloading Operators, Inheritance, Reusability- Types of Inheritance- Advantages of Inheritance-Disadvantages of Inheritance,

**UNIT-IV:** Pointers & Binding Polymorphisms and Virtual Functions Pointer, Features of Pointers- Pointer Declaration- Pointer to Class- Pointer Object The this Pointer- Pointer to Derived Classes and Base Class, Binding Polymorphisms and Virtual Functions.

**UNIT-V:** Generic Programming with Templates & Exception, Need for Templates- Definition of class Templates- Normal Function Templates- Over Loading of Template Function-Bubble Sort Using Function Templates- Difference Between Templates and Macros-, Exception Handling- Principles of Exception Handling- The Keywords try throw and catch- Multiple Catch Statements.

**TEXT BOOKS:**
1. A First Book of C++, Gary Bronson, Cengage Learning.
2. The Complete Reference C++, Herbert Schildt, TMH. 3. Programming in C++, Ashok N Kamthane, Pearson 2nd Edition

**REFERENCE BOOKS:**
1. Object Oriented Programming C++, Joyce Farrell, Cengage.
2. C++ Programming: from problem analysis to program design, DS Malik, Cengage Learning

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### ADVANCED UNIX PROGRAMMING

**Course Objectives:**
Understating the shell commands, shell programming, system calls of files and processes, signals, inter-process communication concepts and programming, TCP and UDP.

**Course Outcomes:**
After finishing this course student will be able to:

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | **Gain** good knowledge on Unix commands AND Awareness of shell programming | **K2** |
| **CO2** | **Know** about different system calls for files and directories | **K6** |
| **CO3** | **Ability** to know the working of processes and signals | **K3** |
| **CO4** | **Application** of client server program for IPC | **K4** |
| **CO5** | **Knowledge** about socket programming | **K3,K5** |

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | | 1 | 2 | 3 | 3 | | | | 3 | | 1 | |
| **CO2** | 1 | | 2 | | 3 | 2 | | | | 2 | 3 | 1 |
| **CO3** | | 2 | | 3 | | 1 | | | 3 | 2 | 1 | |
| **CO4** | 2 | | 3 | 2 | | | | | | 1 | | 1 |
| **CO5** | 2 | 3 | 2 | 1 | | 1 | | | 2 | 3 | 2 | 1 |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**SYLLABUS:**
**UNIT-1:** Introduction, Architecture of unix, Responsibilities of shell, unix file system, vi editor. **Unix commands**: Some Basic Commands, file utilities, process utilities, text processing utilities, network utilities, disk utilities, Security by file permissions.

**UNIT-2 Shell Programming**: shell variables, The Export command, The Profile File a Script Run During starting, The First Shell Script, The read command, Positional Parameters, The $? Variable , The Exit command, Branching Control Structures, Loop Control Structures, The Continue and Break Statement- Real Arithmetic in Shell Programs

**UNIT-3 Files** - Introduction, file descriptors, 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, link, unlink.

**UNIT-4 Directories-**Creating, removing and changing Directories-mkdir, rmdir, chdir, obtaining current working directory-getcwd, Directory contents, Scanning Directories-opendir, readdir, closedir, rewinddir functions.

**UNIT-5**
**Process Control**: process identifiers, fork function, vfork function, exit function, wait and waitpid functions, exec functions, user identification. **Signals**: signal handlin g using signal function, kill and raise, alarm, pause, abort and sleep functions. **Shared memory**-system calls of shared memory, semaphore structure in kernel, client server example.

**TEXT BOOKS:**

1. Unix the ultimate guide, 3rd edition, sumitabha Das, TMH.
2. Advanced programming in the unix environment by W. Richard Stevens.
3. Unix network programming by W. Richard Stevens.

**REFERENCE BOOKS:**

1. Introduction to Unix and shell programming, Venkateshmurhty
2. Unix and shell programming by B.M. Harwani, OXFORD university press.

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**Open Elective-II:**

### OPERATING SYSTEMS

**Course Objectives:**

- Study the basic concepts and functions of operating systems.
- Understand the structure and functions of OS.
- Learn about Processes, Threads and Scheduling algorithms.
- Understand the principles of concurrency and Deadlocks.
- Learn various memory management schemes.
- Study I/O management and File systems.
- Learn the basics of Linux system and perform administrative tasks on Linux Servers.

**Course Outcomes:**

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| CO1 | Describe Computer Operating System Functions, Structures and System Calls. | K1 |
| CO2 | Demonstrate various Process Management Concepts and CPU Scheduling Algorithms and Process Synchronization Techniques. | K3 |
| CO3 | Illustrate Memory Management Techniques and Page Replacement Algorithms. | K2 |
| CO4 | Apply Deadlock Prevention and Avoidance Techniques | K3 |
| CO5 | Demonstrate File System Concepts and Mass Storage Structures | K3 |

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 2 | 1 | 3 | 2 | 2 | | | | 2 | | 1 | |
| CO2 | | 1 | 3 | | 1 | 2 | | | 1 | 1 | 2 | |
| CO3 | 1 | 2 | | 1 | 2 | 1 | | | 3 | 2 | 1 | |
| CO4 | | | 3 | 2 | 3 | | | | 2 | | 2 | 1 |
| CO5 | 2 | 3 | 2 | 2 | 1 | 1 | | | 2 | 3 | 3 | 1 |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**SYLLABUS:**

**UNIT-I: Introduction to Operating System Concept:** Types of operating systems, operating systems concepts, Evaluation of operating systems, operating systems services, structure of OS, Introduction to System call, System call types.

**UNIT-II: Process Management** – Process concept, The process, Process State Diagram , Process control block, Process Scheduling- Scheduling Queues, Schedulers, Operations on Processes, Interprocess Communication, Threading Issues, Scheduling-Basic Concepts, Scheduling Criteria, Scheduling Algorithms.

**UNIT-III: Memory Management:** Swapping, Contiguous Memory Allocation, Paging, structure of the Page Table, **Virtual Memory Management:** Virtual Memory, Demand Paging, Page-Replacement Algorithms.

**UNIT-IV: Concurrency:** Process Synchronization, The Critical- Section Problem, Synchronization Hardware, Semaphores, Classic Problems of Synchronization, Monitors, Synchronization examples **Principles of deadlock**: System Model, Deadlock Characterization, Deadlock Prevention, Detection and Avoidance, Recovery form Deadlock

**UNIT-V: File System Interface:** Concept of a file, Access Methods, Directory structure, File system mounting, file sharing, protection. **File System Implementation:** File system structure, allocation methods, Disk scheduling, **Case studies: Android, UNIX, Windows**

**TEXT BOOK:**

1. Operating System Concepts, Abraham Silberschatz, Peter Baer Galvin and Greg Gagne 9th Edition, John Wiley and Sons Inc., 2012.
2. Operating Systems – Internals and Design Principles, William Stallings, 7th Edition, Prentice Hall, 2011.

**REFERENCES:**

1. Modern Operating Systems, Andrew S. Tanenbaum, Second Edition, Addison Wesley, 2001.
2. Operating Systems: A Design-Oriented Approach, Charles Crowley, Tata McGraw Hill Education, 1996.
3. Operating Systems: A Concept-Based Approach, D M Dhamdhere, Second Edition, Tata McGraw-Hill Education, 2007.
4. Operating Systems-S Halder, Alex A Aravind Pearson Education Second Edition 2016

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### PYTHON PROGRAMMING

**Course Objectives:**

- Introduction to Scripting Language
- Exposure to various problems solving approaches of computer science Syllabus

**Course Outcomes:**

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Describe comprehend the basics of python programming.. | **K1** |
| **CO2** | Demonstrate the principles of structured programming and be able to describe, design, implement, and test structured programs using currently accepted methodology.. | **K3** |
| **CO3** | Explain the use of the built-in data structures list, sets, tuples and dictionary. | **K2** |
| **CO4** | Make use of functions and its applications | **K3** |
| **CO5** | Identify real-world applications using oops, files and exception handling provided by python. | **K3** |

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 1 | 2 | 2 | 2 | 3 | 3 | | | 2 | | 1 | |
| **CO2** | | 1 | 2 | 2 | 1 | 1 | | | 1 | | 2 | |
| **CO3** | 1 | 2 | | 1 | 2 | | | | 3 | 2 | | |
| **CO4** | | | 3 | 2 | 3 | 2 | | | 2 | | | 2 |
| **CO5** | 2 | 3 | 1 | 2 | | 1 | | | 3 | | 2 | 1 |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**SYLLABUS:**

**UNIT – I: Introduction:** History of Python, Python Language, Features of Python, Applications of Python, Using the REPL(Shell), Running Python Scripts, Variables, Assignment, Keywords, Input-Output, Indentation.

**UNIT – II: Types, Operators and Expressions:** Types - Integers, Strings, Booleans; Operators Arithmetic Operators, Comparison (Relational) Operators, Assignment Operators, Logical Operators, Bitwise Operators, Membership Operators, Identity Operators, Expressions and order of evaluations, Control Flow- if, if-elif-else, for, while, break, continue.

**UNIT – III:** Data Structures Lists - Operations, Slicing, Methods; Tuples, Sets, Dictionaries, Sequences. Comprehensions.

**UNIT – IV: Functions** - Defining Functions, Calling Functions, Passing Arguments, Keyword Arguments, Variable-length arguments, Anonymous Functions, Fruitful Functions(Function Returning Values), Scope of the Variables in a Function - Global and Local Variables.

**UNIT – V: Modules:** Creating modules, import statement, from.import statement, name spacing, Python packages, Introduction to PIP, Installing Packages via PIP, Using Python Packages Error and Exceptions: Difference between an error and Exception, Handling Exception, try except block, Raising Exceptions, User Defined Exceptions

**TEXT BOOKS:**

1. Python Programming: A Modern Approach, Vamsi Kurama, Pearson
2. Learning Python, Mark Lutz, Orielly

**REFERENCE BOOKS**:

1. Think Python, Allen Downey, Green Tea Press
2. Core Python Programming, W.Chun, Pearson.
3. Introduction to Python, Kenneth A. Lambert, Cengage

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### WEB TECHNOLOGIES

**Course Objectives:**
- To Learn the basics of Web Designing using HTML, DHTML, and CSS
- To learn the basics about Client side scripts and Server side scripts

**Course Outcomes (COs):**
At the end of the course, student will be able to

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| CO1 | Describe basics of Web Designing using HTML, DHTML, and CSS | K2 |
| CO2 | Build real world applications using client side and server side scripting languages | K3 |
| CO3 | Design and develop applications using web servers | K5 |
| CO4 | Analyze the basics of PHP programming | K4 |
| CO5 | Apply Database connectivity with case study for student Information System and Health Management system | K3 |

#***Based on suggested Revised BTL***

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 1 | 3 | 2 | 2 | 1 | 3 | | | 3 | | 1 | |
| CO2 | 2 | 2 | 3 | | 2 | 1 | | | | | | |
| CO3 | 1 | | 2 | 1 | 2 | | | | 3 | 2 | 1 | |
| CO4 | 1 | 1 | | | 3 | 2 | 1 | | 1 | | 2 | 2 |
| CO5 | 2 | 2 | | 3 | 3 | 1 | | | 2 | | 2 | 1 |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**SYLLABUS:**

**UNIT I HTML :-** Introduction - HTML Formatting - Hyper-Links - Lists - Tables - Images - Forms - Frames - Cascading Style sheets - Types - XML - Document type definition - XML Schemas - Document Object model.

**UNIT II Introduction to Client Side scripting** - JavaScript - Control statements - Functions - Arrays - Objects - Events - Dynamic HTML with Java Script - AJAX: Ajax Client Server Architecture - XML Http Request Object - Call Back Methods.

**UNIT III Web Application-** Web servers – IIS (XAMPP) and Tomcat Servers - Server Side Scripting - Java Servlets - Java Server Pages - JSF Components - Cookies.

**UNIT IV PHP Programming** - Basic Syntax - Defining variable and constant - PHP Data types - Operator and Expression - Operator Precedence - Decisions and Loop - Functions & Recursion - String Processing and Regular Expressions

**UNIT V JDBC:** Database Connectivity with MySQL - Servlets - JSP - PHP – Case Studies - Student information system - Health Management System

**TEXT BOOKS**:

1. Paul J. Deitel, Harvey M. Deitel, Abbey Deitel, "Internet & World Wide Web How to Program", Fifth Edition, Deitel Series, 2012.
2. Jason Gilmore, "Beginning PHP and MySQL from Novice to Professional", Fourth Edition, Apress Publications, 2010.
3. Brown, Ethan, "Web Development with Node and Express: Leveraging the JavaScript Stack", O'Reilly Media, 2019. CSE Dept. Flexible Curriculum NITTUGCSE19 95
4. Anthony, Accomazzo, Murray Nathaniel, Lerner Ari, "Fullstack React: The Complete Guide to React JS and Friends", Fullstack.io, 2017.
5. Kozlowski, Pawel, "Mastering Web Application Development with Angular JS", Packt Publishing Ltd., 2013.

**REFERENCE BOOKS:**

1. Robert W. Sebesta, "Programming with World Wide Web", Fourth Edition, Pearson, 2008.
2. David William Barron, "The World of Scripting Languages", Wiley Publications, 2000.
3. Dayley B., "Node.js, MongoDB, and AngularJS Web Development", Addison-Wesley Professional, 2014.
4. Vainikka J., "Full-Stack Web Development using Django REST Framework and React", 2018

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### SOFT COMPUTING

**Course Objectives:** In the course the student will Learn soft computing concepts and techniques and foster their abilities in designing and implementing soft computing based solutions for real-world problems.

**Course Outcomes (COs)**: At the end of the course, student will be able to

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Able to apply fuzzy logic and reasoning to handle uncertainty in engineering problems. | **K2** |
| **CO2** | Make use of genetic algorithms to combinatorial optimization problems | **K3** |
| **CO3** | Apply artificial intelligence techniques, including search heuristics, knowledge representation, planning and reasoning. | **K5** |
| **CO4** | Learn and apply the principles of self adopting and self organizing neuro fuzzy inference systems | **K4** |
| **CO5** | Evaluate and compare solutions by various soft computing approaches for a given problem | **K3** |

#***Based on suggested Revised BTL***

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | | 3 | | 2 | 1 | 3 | | | 2 | 1 | 2 | |
| **CO2** | 1 | 2 | 3 | 1 | 2 | 1 | | | | 2 | 1 | |
| **CO3** | 1 | 2 | | 3 | | | 1 | | 3 | | 1 | |
| **CO4** | 1 | 1 | 2 | | | 1 | 1 | | 1 | | | 2 |
| **CO5** | 1 | 2 | | 3 | | 2 | | | | | | 1 |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**SYLLABUS:**

**UNIT I :FUZZY SET THEORY:** Introduction to Neuro – Fuzzy and Soft Computing, Fuzzy Sets, Basic Definition and Terminology, Set-theoretic Operations, Member Function Formulation and Parameterization, Fuzzy Rules and Fuzzy Reasoning, Extension Principle and Fuzzy Relations. Fuzzy Inference Systems, Mamdani Fuzzy Models, Sugeno Fuzzy Models.

**UNIT II: OPTIMIZATION:** Derivative based Optimization, Descent Methods, The Method of Steepest Descent, Classical Newton's Method, Step Size Determination, Derivative-free Optimization, Genetic Algorithms.

**UNIT III: ARTIFICIAL INTELLIGENCE :** Introduction, Knowledge Representation, Reasoning, Issues and Acquisition: Prepositional and Predicate Calculus Rule Based knowledge Representation Symbolic Reasoning, Heuristic Search: Techniques for Heuristic search Heuristic Classification.

**UNIT IV: NEURO FUZZY MODELING:** Adaptive Neuro-Fuzzy Inference Systems, Architecture – Hybrid Learning Algorithm, Learning Methods that Cross-fertilize ANFIS and RBFN –Framework Neuron Functions for Adaptive Networks – Neuro Fuzzy Spectrum.

**UNIT V: APPLICATIONS OF COMPUTATIONAL INTELLIGENCE :** Printed Character Recognition, Inverse Kinematics Problems, Automobile Fuel Efficiency Prediction, Soft Computing for Color Recipe Prediction.

**TEXT BOOKS:**

1. J.S.R.Jang, C.T.Sun and E.Mizutani, “Neuro-Fuzzy and Soft Computing”, PHI, 2004, Pearson Education 2004
2. N.P.Padhy, “Artificial Intelligence and Intelligent Systems”, Oxford University Press, 2006.

**REFERENCES:**

1. Elaine Rich & Kevin Knight, Artificial Intelligence, Second Edition, Tata Mcgraw Hill Publishing Comp., 2006, New Delhi.
2. Timothy J.Ross, “Fuzzy Logic with Engineering Applications”, McGraw-Hill, 1997.
3. Davis E.Goldberg, “Genetic Algorithms: Search, Optimization and Machine Learning”, Addison Wesley, N.Y., 1989.
4. S. Rajasekaran and G.A.V.Pai, “Neural Networks, Fuzzy Logic and Genetic Algorithms”, PHI,
5. R.Eberhart, P.Simpson and R.Dobbins, “Computational Intelligence - PC Tools”, AP Professional, Boston, 1996.
6. Amit Konar, “Artificial Intelligence and Soft Computing Behaviour and Cognitive model of the human brain”, CRC Press, 2008

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### DISTRIBUTED COMPUTING

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

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Explain the fundamentals of Distributed Computing | **K2** |
| **CO2** | Identify an Message communication process | **K3** |
| **CO3** | Briefly explain uses of RPC Model in a system | **K3** |
| **CO4** | Design and Implementation issues of DSM | **K6** |
| **CO5** | Compare the relationship between Clock Synchronization, Algorithms | **K2** |

***#Based on suggested Revised BTL***

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 1 | 3 | | 2 | 1 | 3 | | | 2 | 2 | 3 | 1 |
| **CO2** | 2 | 2 | 3 | 2 | 2 | 1 | | | | | 1 | |
| **CO3** | 1 | | 3 | | 1 | 2 | | | 3 | | 1 | 2 |
| **CO4** | 2 | | | 2 | | 2 | 1 | | 1 | | | |
| **CO5** | 1 | 3 | 2 | 3 | 1 | | 1 | | | 2 | | 2 |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**SYLLABUS:**

**UNIT 1:** Fundamentals Evolution of Distributed Computing Systems, System models, issues in design of Distributed Systems, Distributed computing environment, web based distributed model, computer networks related to distributed systems and web based protocols.

**UNIT 2:** Message Passing Inter process Communication, Desirable Features of Good Message-Passing Systems, Issues in IPC by Message, Synchronization, Buffering, Multidatagram Messages, Encoding and Decoding of Message Data.

**UNIT 3:** Remote Procedure Calls The RPC Model, Transparency of RPC, Implementing RPC Mechanism, Stub Generation, RPC Messages, Server Management, Communication Protocols for RPCs, Complicated RPCs, Client-Server Binding, Exception Handling, Security, Some Special Types of RPCs.

**UNIT 4:** Distributed Shared Memory Design and Implementation issues of DSM, Granularity, Structure of Shared memory Space, Consistency Models,Thrashing, Other Approaches to DSM, Advantages of DSM.

**UNIT 5:** Synchronization Clock Synchronization, Event Ordering, Mutual Exclusion, Election Algorithms.

**TEXT BOOKS:**

1. Distributed OS by Pradeep K. Sinha (PHI)
2. Tanenbaum S.: Distributed Operating Systems, Pearson Education

**REFERENCES:**

1. Tanenbaum S. Maarten V.S.: Distributed Systems Principles and Paradigms, (Pearson Education)
2. George Coulouris, Jean Dollimore. Tim Kindberg: Distributed Systems concepts and design.

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### AI and ML for Robotics

**Course Objectives** :

To learn the concepts of searching for AI problems
- To learn about agents and knowledge representation
- To understand the various factors involved in inferences
- To get introduced to fundamentals of machine learning
- To learn about the possibilities of Supervised and Unsupervised learning

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

| | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Explain the History of AI - Agents - Structure of Intelligent agents | **K5** |
| **CO2** | Design agents for any given problem | **K3** |
| **CO3** | Describe Represent real world knowledge using first order or propositional logic | **K6** |
| **CO4** | To make use of Solve problems by appropriated using the supervised or unsupervised machine learning algorithms | **K5** |
| **CO5** | Develop appropriate clustering algorithm for solving real-world problems | **K4** |

***#Based on suggested Revised BTL***

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 2 | 2 | 1 | 3 | 2 | 3 | | | 2 | 3 | 3 | |
| **CO2** | 2 | 1 | 3 | 2 | 2 | 1 | | | 1 | | 1 | 2 |
| **CO3** | 1 | | 3 | | 1 | 2 | | | 3 | 2 | 1 | 2 |
| **CO4** | 2 | 2 | | 2 | 3 | 2 | 2 | | 1 | 1 | | |
| **CO5** | 1 | 2 | | 3 | 1 | | 1 | | | 2 | 2 | 2 |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**SYLLABUS:**

**UNIT I** : AI - History of AI - Agents - Structure of Intelligent agents - Environments - Problem solving methods - Problem solving agents - Formulating problems - search strategies - Breadth-first - Uniform cost - Depth-first – Depth limited - Bidirectional - Informed Search - Best-first Heuristic Functions - Memory bounded search - A* - SMA* - Iterative Improvement algorithms - Hill Climbing - Simulated annealing - Measure of performance and analysis of search algorithms.

**UNIT II:** Game playing - Perfect Decisions - Imperfect Decisions - Alpha-beta pruning - Knowledge based agent - Wumpus World Environment - Propositional logic - agent for wumpus world - First order logic - syntax - semantics - extensions - Using First order logic - Representation change in the world - Goal based agents.

**UNIT III:** Knowledge Base - Knowledge representation - Production based system - Frame based system - Inference - Backward chaining - Forward chaining.

**UNIT IV:** Learning from agents - inductive learning - Types of Machine learning - Supervised learning - learning decision trees - support vector machines - Neural and Belief networks - Perceptron - Multi-layer feed forward networks - Bayesian belief networks.

**UNIT V:** Unsupervised learning - K-means clustering - hierarchical clustering - Agglomerative and Divisive clustering - Fuzzy clustering.

**TEXT BOOKS:**

1. Stuart Russel, Peter Norvig, “AI – A Modern Approach”, Second Edition, Pearson Education, 2007.
2. Kevin Night, Elaine Rich, Nair B., “Artificial Intelligence (SIE)”, McGraw Hill, 2008.

**REFERENCE BOOKS**:

1. Vinod Chandra SS, Anand Hareendran S, “Artificial and Machine Learning”, First Edition, PHI Learning, 2014.
2. Dan W. Patterson, “Introduction to AI and ES”, Pearson Education, 2007
3. G. Luger, W. A. Sttubblefield, “Artificial Intelligence”, Third Edition, Addison-Wesley Longman, 1998.
4. N. J. Nilson, “Principles of Artificial Intelligence”, Narosa Publishing House, 1980.
5. Tom Mitchell, “Machine Learning”, First Edition, Tata McGraw Hill India, 2017.

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**Open Elective-III:**

### BIG DATA ANALYTICS

**Course Objectives:**

- To understand the competitive advantages of big data analytics
- To understand the big data frameworks
- To learn data analysis methods
- To learn stream computing
- To gain knowledge on Hadoop related tools such as HBase, Cassandra, Pig, and Hive for big data analytics

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

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Understand how to leverage the insights from big data analytics | **K2** |
| **CO2** | Analyze data by utilizing various statistical and data mining approaches | **K3** |
| **CO3** | Perform analytics on real-time streaming data | **K3** |
| **CO4** | Understand the various NoSql alternative database models | **K4** |

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 2 | 2 | 3 | 2 | 2 | 2 | | | 2 | 3 | 3 | |
| **CO2** | | 1 | 3 | 2 | | 1 | | | 1 | | 1 | 1 |
| **CO3** | 1 | | 3 | | 1 | 2 | | | 3 | 2 | | 2 |
| **CO4** | 2 | 1 | | 2 | | 3 | 2 | | 1 | 1 | | 1 |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**SYLLABUS:**

**UNIT I- INTRODUCTION TO BIG DATA:** Big Data – Definition, Characteristic Features – Big Data Applications - Big Data vs Traditional Data - Risks of Big Data - Structure of Big Data - Challenges of Conventional Systems - Web Data – Evolution of Analytic Scalability.

**UNIT II- HADOOP FRAMEWORK**: Distributed File Systems - Large-Scale FileSystem Organization – HDFS concepts - MapReduce Execution, Algorithms using MapReduce – Hadoop YARN.

**UNIT III-DATA ANALYSIS: Statistical Methods:** Regression modelling, Multivariate Analysis - Classification: SVM & Kernel Methods - Rule Mining - Cluster Analysis, Types of Data in Cluster Analysis, Partitioning Methods, Predictive Analytics – Data analysis using R.

**UNIT IV- MINING DATA STREAMS: Streams:** Concepts – Stream Data Model and Architecture - Sampling data in a stream - Mining Data Streams and Mining Time-series data - Real Time Analytics Platform (RTAP) Applications - Case Studies – Real Time Sentiment Analysis.

**UNIT V- BIG DATA FRAMEWORKS:** Introduction to NoSQL – Aggregate Data Models – Hbase: Data Model and Implementations – Hbase Clients – Examples – .Cassandra: Data Model – Examples – Cassandra Clients – Hadoop Integration.

**SOFTWARE LINKS:**

1. Hadoop: http://hadoop.apache.org/
2. Hive: https://cwiki.apache.org/confluence/display/Hive/Home
3. Pig latin: http://pig.apache.org/docs/r0.7.0/tutorial.html

**TEXT BOOKS:**

1. Big Java 4th Edition, Cay Horstmann, Wiley John Wiley & Sons, INC
2. Hadoop: The Definitive Guide by Tom White, 3rd Edition, O'reilly
3. Hadoop in Action by Chuck Lam, MANNING Publ.

**REFERENCE BOOKS:**

1. Hadoop in Practice by Alex Holmes, MANNING Publ.
2. Hadoop MapReduce Cookbook, SrinathPerera, ThilinaGunarathne
3. Michael Berthold, David J. Hand, ―Intelligent Data Analysis‖, Springer, Second Edition, 2007.
4. Michael Minelli, Michelle Chambers, and Ambiga Dhiraj, "Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends for Today's Businesses", Wiley, 201

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### AI TOOLS & TECHNIQUES

**Course Objectives:**

- To learn the basic concepts and techniques of AI and machine learning
- To explore the various mechanism of Knowledge and Reasoning used for building expert system.
- To become familiar with supervised and unsupervised learning models
- To design and develop AI and machine learning solution using modern tools.

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

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Explain the fundamentals of AI and machine learning | **K2** |
| **CO2** | Identify an appropriate AI problem solving method and knowledge representation technique | **K3** |
| **CO3** | Identify appropriate machine learning models for problem solving | **K3** |
| **CO4** | Design and develop the AI applications in real world scenario | **K6** |
| **CO5** | Compare the relationship between AI, ML, and Deep Learning | **K2** |

***#Based on suggested Revised BTL***

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 1 | 2 | 3 | 3 | 1 | 3 | 1 | | 2 | 3 | 3 | |
| **CO2** | 2 | 1 | | 2 | 3 | 1 | | | 2 | | 1 | |
| **CO3** | | 2 | 3 | | 1 | 2 | | | 3 | 2 | 1 | 2 |
| **CO4** | 2 | 2 | | 2 | | 2 | | | 1 | 2 | 1 | |
| **CO5** | 1 | | | 3 | 1 | | 1 | | 1 | 2 | 2 | |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**SYLLABUS:**

**UNIT– I: Introduction to AI-** Definition, Problem, State space representation. Intelligent Systems: Categorization of Intelligent System, Components of AI Program, Foundations of AI, Applications of AI, Current trends in AI, Intelligent Agents: Anatomy, structure, Types.

**UNIT– II: Problem solving-**Solving problem by Searching: Problem Solving Agent, Formulating Problems. Uninformed Search Methods: Breadth First Search (BFS), Depth First Search (DFS), Depth Limited Search, Depth First Iterative Deepening (DFID), Informed Search Methods- Greedy best first Search, A* Search, Memory bounded heuristic Search. Local Search Algorithms and Optimization Problems- Hill climbing search Simulated annealing and local beam search.

**UNIT – III: Knowledge and Reasoning-**Knowledge based Agents, The Wumpus World, and Propositional logic**. First Order Logic**- Syntax and Semantic, Inference in FOL, Forward chaining, backward Chaining, Knowledge Engineering in First-Order Logic, Unification and Resolution.

**UNIT – IV: Concepts of Machine learning** -Supervised, unsupervised, semi-supervised, Rote learning, Reinforcement learning, Issues, steps and applications, Designing a learning System. Case study- hand written digit recognition, stock price prediction. Learning Models- Decision tree learning. Probabilistic Models, Deterministic Models, Hidden Markov Model, Reinforcement Learning-Model based learning, Temporal Difference Learning, Generalization, Partially Observable States.

**UNIT – V: Artificial Neural Network:** Introduction, neural network representation, Problems for neural network learning, perception, multilayer network & Back propagation Algorithm**.** Deep learning- Definition, relationship between AI, ML, and Deep Learning, Trends in Deep Learning.

**TEXT BOOKS:**

1. Artificial Intelligence and Machine Learning, 1st Edition, Vinod Chandra S.S., Anand Hareendran S, 2014
2. Artificial Intelligence: A Modern Approach, 2nd Edition, Pearson Education, Stuart J. Russell, Peter Norvig, 2002

**REFERENCE BOOKS:**

1. PROLOG Programming for Artificial Intelligence", 3rd Edition, Pearson Education, Ivan Bratko, 2002
2. Artificial Intelligence, Third Edition, McGraw Hill Education, Elaine Rich and Kevin Knight, 2017
3. Data Mining Concepts and Techniques, Morgann Kaufmann Publishers, Han Kamber, 2011
4. Machine learning with R, 2nd Edition, Brett Lantz, 2015
5. Genetic Algorithms: Search, Optimization and Machine Learning, 1st ed, Davis E. Goldberg, Addison Wesley, N.Y., 1989

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### IMAGE PROCESSING

**Course Objectives:**

- Describe and explain basic principles of digital image processing.
- Design and implement algorithms that perform basic image processing (e.g. noise removal and image enhancement).
- Design and implement algorithms for advanced image analysis (e.g. image compression, image segmentation).
- Assess the performance of image processing algorithms and systems.

**Course Outcomes:**

After the completion of the course, student will be able to

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Demonstrate the components of image processing | **K2** |
| **CO2** | Explain various filtration techniques. | **K5** |
| **CO3** | Apply image compression techniques. | **K3** |
| **CO4** | Discuss the concepts of wavelet transforms. | **K6** |
| **CO5** | Analyze the concept of morphological image processing. | **K4** |

**#based on suggested Revised BTL**

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 1 | 2 | | 3 | | 2 | 1 | | 2 | 3 | | |
| **CO2** | 2 | 1 | | 2 | 3 | 1 | | | | | 1 | |
| **CO3** | | | 3 | | 1 | 2 | | | 3 | 2 | 2 | 1 |
| **CO4** | 1 | 2 | | 1 | | 2 | | | 2 | 3 | 1 | |
| **CO5** | 1 | | | 3 | 1 | | 1 | | 1 | 2 | | |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**SYLLABUS:**

**UNIT I: Introduction:** Fundamental steps in Image Processing System, Components of Image Processing System, Elements of Visual Perception, Image Sensing and acquisition, Image sampling & Quantization, Basic Relationship between pixels.

**UNIT II: Image Enhancement Techniques**: Spatial Domain Methods: Basic grey level transformation, Histogram equalization, Image subtraction, image averaging.
**Spatial filtering**: Smoothing, sharpening filters, Laplacian filters.

**UNIT III: Image Compression**: Redundancies- Coding, Inter pixel, Psycho visual; Fidelity, Source and Channel Encoding, **Elements of Information Theory;** Loss

Less and Lossy Compression; Run length coding, Differential encoding, DCT, Image Compression Standards-JPEG, JPEG 2000, MPEG; Video compression.

**UNIT IV: Wavelet Based Image Compression**: Expansion of functions, Multi-resolution analysis, Scaling functions, MRA refinement equation, Wavelet series expansion, Discrete Wavelet Transform (DWT), Continuous, Wavelet Transform.

**UNIT V: Image Segmentation**: Discontinuities, Edge Linking and boundary detection, Thresholding, Region Based Segmentation, basic gray-scale morphology operations; Feature extraction; Classification; Object recognition. **Digital Image Watermarking:** Introduction, need of Digital Image Watermarking, applications of watermarking.

**TEXT BOOKS:**

1. Digital Image Processing. 2nd ed. Gonzalez, R.C. and Woods, R.E. India: Person Education,2009

**REFERENCE BOOKS:**

1. Digital Image Processing. John Wiley, Pratt, W. K, Fourth Edition-2001
2. Digital Image Processing, Jayaraman, S., Veerakumar, T. and Esakkiranjan, S.,Tata McGraw-Hill,Edition-3,2009

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### INFORMATION SECURITY

**Course Course Objectives:**

- To provide an understanding of principal concepts, major issues, technologies, and basic approaches in information security
- Master the key concepts of information security and how they "work."
- Develop a "security mindset:" learn how to critically analyze situations of computer and network usage from a security perspective, identifying the salient issues, viewpoints, and trade-offs.

**Course Outcomes:**

After finishing this course student will be able to:

| CO | Course Outcome | Knowledge Level# |
|---|---|---|
| **CO1** | Evaluate vulnerability of an information system and establish a plan for risk management. | **K2** |
| **CO2** | Demonstrate basic principles of Web application security | **K5** |
| **CO3** | Evaluate the authentication and encryption needs of an information system. | **K2** |
| **CO4** | Demonstrate how to secure a IP & TLS | **K3** |
| **CO5** | Evaluate System security policies and procedures | **K5** |

**#based on suggested Revised BTL**

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 1 | 2 | 1 | 1 | 2 | | | | 2 | | | |
| **CO2** | 2 | | 1 | 3 | 3 | 1 | | | | 2 | 1 | |
| **CO3** | 1 | 2 | 3 | | 1 | 3 | | | 3 | 2 | | |
| **CO4** | | 2 | 1 | 2 | | | | | | | 1 | |
| **CO5** | 1 | 2 | | 3 | 1 | | | | 1 | 3 | | |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**SYLLABUS:**

**UNIT – I: Computer Security Concepts:** OSI Security Architecture, Security Attacks, Security Services, Security Mechanisms, Model for Network Security, Classical Encryption Techniques: Transposition Techniques, Substitution Techniques, Steganography.

**UNIT – II: Cryptographic Ciphers:** Block Cipher Principles, DES, AES.
**Asymmetric Cipher:** Principles of Public-Key Cryptography, RSA, Diffie– Hellman Key Exchange.

**UNIT – III: Cryptographic Hash Functions:** Applications of Cryptographic Hash Functions, SHA.

**Message Authentication Codes**: Message Authentication Requirements, Message Authentication Functions, Requirements for MAC, HMAC,

**Digital Signatures**: Digital Signatures Properties, Attacks & Forgeries, Digital Signatures Requirements, Direct Digital Signature, Elgamal DSS, Digital Signature Scheme.

**UNIT – IV: IP & Transport-Level Security: IP Security:** IP Security Overview, Applications of IPsec, Benefits of IPsec, Routing Applications, IPsec Documents, IPsec Services, Transport & Tunnel Models, Authentication Header, ESP.

**Transport-Level Security:** Web Security Considerations: Web Security Threats, Web Traffic Security Approaches. **Secure Socket Layer:** SSL Architecture, Record Protocols, Change Cipher Spec protocol, Handshake Protocol. **Transport Layer Security:** Version Number, Pseudorandom Function, Cipher Suites, Cryptographic Computations, Padding.

**UNIT – V: System Security:** Intruders, Detection, Password Management, Malicious Software Types, Viruses, Worms, Firewalls.

**TEXT BOOKS:**

1. William Stallings, Cryptography and Network Security: Principles and Practice, Sixth Edition, Pearson, 2011.

**REFERENCES:**

1. Security in Computing, Fourth Edition, by Charles P. Pfleeger, Pearson Education
2. Modern Cryptography: Theory and Practice, by Wenbo Mao, Prentice Hall.
   Network Security Essentials: Applications and Standards, by William Stallings, Prentice Hall.

**Department of Computer Science & Engineering**
**University College of Engineering, JNT University Kakinada**

## MOBILE APPLICATION DEVELOPMENT

**Course Outcomes:** After finishing this course student will be able to:

| CO | Course Outcomes | Knowledge Level# |
|---|---|---|
| **CO1** | Install and configure Android application development tools. | **K2** |
| **CO2** | Design and develop user Interfaces for the Android platform. | **K5** |
| **CO3** | Save state information across important operating system events. | **K2** |
| **CO4** | Apply Java programming concepts to Android application development. | **K3** |
| **CO5** | Design and Implement Packaging and Deploying, Performance Best Practices | **K5** |

**#based on suggested Revised BTL**

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 1 | 2 | 1 | 1 | 2 | | | | 2 | | | |
| CO2 | 2 | | 1 | 3 | 3 | 1 | | | | 2 | 1 | |
| CO3 | 1 | 2 | 3 | | 1 | 3 | | | 3 | 2 | | |
| CO4 | | 2 | 1 | 2 | | | | | | | 1 | |
| CO5 | 1 | 2 | | 3 | 1 | | | | 1 | 3 | | |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**SYLLABUS:**

**UNIT I: Introduction to Mobile Computing-** Android Development Environment, Factors in Developing Mobile Applications, Mobile Software Engineering, Frameworks and Tools, Generic UI Development, Android User.

**UNIT II: Intents on UIs VUIs and Mobile Apps:** Android Intents and Services, Characteristics of Mobile Applications, Successful Mobile Development- Storing and Retrieving Data-Synchronization and Replication of Mobile Data, Getting the Model Right, Android Storing and Retrieving Data, Working with a Content Provider

**UNIT III: Communications Via Network and the Web-** State Machine, Correct Communications Model, Android Networking and Web, Telephony- Deciding Scope of an App, **Notifications and Alarms-** Performance, Performance and Memory Management, Android Notifications and Alarms

**UNIT IV: Graphics-**Performance and Multithreading, Graphics and UI Performance, Android Graphics and Multimedia, Mobile Agents and Peer-to-Peer Architecture, Android Multimedia Location, Mobility and Location Based Services, Android,

**UNIT V:**Packaging and Deploying, Performance Best Practices, Android Field Service App, Security and Hacking - Active Transactions, More on Security, Hacking Android, **Platforms and Additional Issues** - Development Process, Architecture, Design, Technology Selection, Mobile App Development Hurdles, Testing

**TEXT BOOKS:**

1. Professional Mobile Application Development 11 October 2012 by Jeff Mcherter and Scott Gowell

**REFERENCE BOOK:**

1. Android Programming: The Big Nerd Ranch Guide (3rd Edition)
2. iOS Programming: The Big Nerd Ranch Guide (6th Edition)
3. Mastering Xamarin UI Development 1st
4. Xamarin Mobile Application Development: Cross-Platform C# and Xamarin.Forms Fundamentals 1st
5. Professional Android 4 Application Development
6. Android Programming for Beginners

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### CLOUD COMPUTING

**Course Objectives:**

- Ability to understand various service delivery models of a cloud computing architecture.
- Ability to understand the ways in which the cloud can be programmed and deployed.
- Understanding cloud service providers.
- Understand and learn the Characteristics of IaaS and Characteristics of PaaS
- Learn and gain knowledge about the Cloud Service providers.

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

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Explain different types of Computing | **K3** |
| **CO2** | Illustrate Four types of Cloud Deployment Models | **K4** |
| **CO3** | Demonstrate different Phases of Cloud Migration Approaches for Cloud Migration | **K5** |
| **CO4** | Analyze and Develop Cloud Service Models | **K3** |
| **CO5** | Design applications for an organization which use cloud environment. | **K4** |

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 1 | 2 | 1 | | 2 | | | | 2 | | | |
| **CO2** | 2 | | 2 | 3 | | 2 | | | | 2 | 1 | |
| **CO3** | 1 | | 3 | | 1 | 3 | | | 3 | 2 | | |
| **CO4** | | 2 | 1 | 2 | | | | | | 2 | 1 | |
| **CO5** | 1 | 2 | | | 1 | | | | 1 | 3 | | |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**SYLLABUS:**

**UNIT-I: Computing Paradigms:** High-Performance Computing, Parallel Computing, Distributed Computing, Cluster Computing, Grid Computing, Cloud Computing, Quantum Computing, Optical Computing, Nano computing.

**UNIT-2: Cloud Computing Fundamentals:** Motivation for Cloud Computing, The Need for Cloud Computing, Definition of Cloud computing, Cloud Computing Is a Service, Principles of Cloud computing, Five Essential Characteristics, Four Cloud Deployment Models.

**UNIT-3: Cloud Computing Architecture and Management:** Cloud architecture, Layer, Anatomy of the Cloud, Network Connectivity in Cloud Computing, Applications, Managing the Cloud Infrastructure Managing the Cloud application, Migrating Application to Cloud, Phases of Cloud Migration Approaches for Cloud Migration.

**UNIT-4: Cloud Service Models:** Infrastructure as a Service, Characteristics of IaaS. Suitability of IaaS, Pros and Cons of IaaS, Summary of IaaS Providers, Platform as a Service, Characteristics of PaaS, Suitability of PaaS, Pros and Cons of PaaS, Summary of PaaS Providers, Software as a Service, Characteristics of SaaS, Suitability of SaaS, Pros and Cons of SaaS, Summary of SaaS Providers.

**UNIT-5**: Python for Amazon Web Services, Python for Google Cloud Platform, Python for Windows Azure, Python for MapReduce, Python Packages of Interest, Python Web Application Framework – Django, Designing a RESTful Web API.

**TEXT BOOKS:**

1. Essentials of cloud Computing: K. Chandrasekhran, CRC press, 2014
2. ArshadeepBhaga, Vijay Madisetti, “Cloud Computing AHandson Approach”, Universities Press, 2018.

**REFERENCE BOOKS:**

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

**<u>Open Elective-IV:</u>**

## CYBER SECURITY

**Course Objectives:**

- To learn threats and risks within context of the cyber security architecture.
- Student should learn and Identify security tools and hardening techniques.
- To learn types of incidents including categories, responses and timelines for response.

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

| Course Outcomes | | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Apply cyber security architecture principles. | **K3** |
| **CO2** | Demonstrate the risk management processes and practices. | **K2** |
| **CO3** | Appraise cyber security incidents to apply appropriate response | **K5** |
| **CO4** | Distinguish system and application security threats and vulnerabilities. | **K4** |
| **CO5** | Identify security tools and hardening techniques | **K3** |

***#Based on suggested Revised BTL***

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 2 | 2 | 1 | 1 | 2 | | | | 2 | 1 | | |
| CO2 | 1 | 3 | | 3 | 3 | 2 | | | | 2 | 1 | |
| CO3 | 1 | 2 | 3 | | 1 | 3 | | | 2 | 1 | | 1 |
| CO4 | 1 | 3 | 1 | | 3 | | | | | 2 | 1 | |
| CO5 | 1 | 2 | | 3 | 1 | | | | 1 | 3 | | |

**(Please fill the above with Levels of Correlation, viz., L, M, H)**

**SYLLABUS:**

**UNIT–I: Introduction to Cyber Security-**Cyber security objectives, roles, differences between information security and cyber security, Cyber security principles-confidentiality, integrity, availability, authentication and non repudiation

**UNIT–II: Information Security within Lifecycle Management-**Lifecycle management landscape, Security architecture processes, Security architecture tools, **Risks & Vulnerabilities-**Basics of risk management, Operational threat environments.

**UNIT–III: Incident Response-**Incident categories, Incident response, Incident recovery, **Operational security protection**-Digital and data assets, ports and protocols, Protection technologies, Identity and access Management.

**UNIT–IV: Threat Detection and Evaluation Monitoring-**Vulnerability management, Security logs and alerts, Monitoring tools and appliances, **Analysis-** Network traffic analysis, packet capture and analysis

**UNIT–V: Introduction to backdoor System and security-**Introduction to metasploit, backdoor, demilitarized zone (DMZ), Digital signature, Brief study on Harding of operating system.

**TEXT BOOKS**:

1. NASSCOM: Security Analyst Student Hand Book, Dec 2015
2. Information Security Management Principles, Updated Edition, David Alexander, Amanda Finch, David Sutton, BCS publishers, June 2013

**REFERENCE BOOKS:**

1. Cyber Security Fundamentals-Cyber Security, Network Security and Data Governance Security, 2nd Edition, ISACA Publishers, 2019

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### DEEP LEARNING

**Course Objectives:**
At the end of the course, the students will be expected to:

- Learn deep learning methods for working with sequential data,
- Learn deep recurrent and memory networks,
- Learn deep Turing machines,
- Apply such deep learning mechanisms to various learning problems.
- Know the open issues in deep learning, and have a grasp of the current research directions.

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

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Demonstrate the basic concepts fundamental learning techniques and layers. | **K2** |
| **CO2** | Discuss the Neural Network training, various random models. | **K6** |
| **CO3** | Explain different types of deep learning network models. | **K5** |
| **CO4** | Classify the Probabilistic Neural Networks. | **K2** |
| **CO5** | Implement tools on Deep Learning techniques. | **K3** |

**#based on suggested Revised BTL**

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 1 | 2 | 1 | 1 | 2 | | | | 2 | | |
| **CO2** | 2 | 1 | 3 | 2 | 3 | 1 | | | | | 1 |
| **CO3** | 1 | | 3 | | 1 | 3 | | | 3 | 2 | |
| **CO4** | | 2 | | 2 | 3 | 1 | | | | | 1 |
| **CO5** | 1 | 2 | | 3 | 1 | | | | 2 | 1 | |

**(Please fill the above with Levels of Correlation, viz., L, M, H)**

**Syllabus:**

**UNIT-1: Foundations of Neural Networks Introduction**- Definition, paradigms, perceptive and issues of neural networks, **neural networks –** Biological neuron, Perceptron, Multi Layer Perceptron.

**UNIT-2:Fundamentals of deep learning**- Definition of Deep Learning, Common Architecture Principles of Deep Networks, Building Blocks of Deep Learning.
**Architectures of Deep Networks**- Feed Forward Neural Networks, Convolution Neural Networks, Recurrent Neural Networks, Recursive Neural Networks, LSTM

**UNIT -3: Deep Learning Research Linear factor models** - Probabilistic PCA And Factor Analysis, Independent Component Analysis, Sparse Coding, Manifold Interpretation of PCA, Auto Encoders.

**Department of Computer Science & Engineering**
**University College of Engineering, JNT University Kakinada**

**UNIT -4: Deep Generating Models :** Boltzmann Machines, Deep Belief Networks, Deep Boltzmann Machines, Convolution Boltzmann Machines, Backpropagation through Random Operations, Directed Generative Nets, Generating Static Networks.

**UNIT -5: Applications:** Large Scale Deep Learning, Image Recognition, Speech Recognition, Natural Language Processing, Other Applications.
**Deep Learning Frameworks**: Tensor Flow, Opencv, Pytorch, Keras, DL4J.

**TEXT BOOKS:**

1. Deep learning A practitioner's approach- josh Patterson and Adam Gibson, OREILLY.
2. Goodfellow, I., Bengio,Y., and Courville, A., Deep Learning, MIT Press, 2016..

**REFERENCE BOOKS:**

1. Artificial Neural Networks, Yegnanarayana, B., PHI Learning Pvt. Ltd, 2009.
2. Matrix Computations, Golub, G.,H., and Van Loan,C.,F, JHU Press,2013.
3. Neural Networks: A Classroom Approach, Satish Kumar, Tata McGraw-Hill Education, 2004.
4. Bishop, C.M., Pattern Recognition and Machine Learning, Springer, 2006.

**Web Link:**

1. Swayam NPTEL: Deep Learning: https://onlinecourses.nptel.ac.in/noc22_cs22/preview

**Department of Computer Science & Engineering**
**University College of Engineering, JNT University Kakinada**

## DATA SCIENCE

**Course Objectives:**

From the course the student will learn

- Provide you with the knowledge and expertise to become a proficient data scientist.
- Demonstrate an understanding of statistics and machine learning concepts that are vital for data science;
- Produce Python code to statistically analyse a dataset;
- Critically evaluate data visualisations based on their design and use for communicating stories from data

**Course Outcomes:**

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| CO1 | Acquire the knowledge and expertise to become a proficient data scientist | K3 |
| CO2 | Demonstrate an understanding of statistics and machine learning concepts that are vital for data science | K3 |
| CO3 | Explain how data is collected, managed and stored for data science | K2 |
| CO4 | Interpret the key concepts in data science, including their real-world applications and the toolkit used by data scientists | K2 |
| CO5 | Illustrate data collection and management scripts using MongoDB | K3 |

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 1 | | 1 | 1 | 2 | | | | 2 | 1 | |
| CO2 | 1 | 1 | | 3 | | 3 | | | | | 1 |
| CO3 | 2 | 2 | 3 | 1 | 1 | | | | 3 | 2 | |
| CO4 | 1 | | | 2 | 3 | 1 | | | | | 1 |
| CO5 | 1 | 2 | | 2 | 3 | | | | 2 | 1 | |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**SYLLABUS**

**UNIT-I: Introduction to Core Concepts and Technologies-** Introduction, Terminology, data science process, data science toolkit, Types of data, Example applications.

**UNIT-II: Data Collection and Management-** Introduction, Sources of data, Data collection and APIs, Exploring and fixing data, Data storage and management, using multiple data sources.

**Department of Computer Science & Engineering**
**University College of Engineering, JNT University Kakinada**

**UNIT -III: Data Analysis-** Introduction, Terminology and concepts, Introduction to statistics, Central tendencies and distributions, Variance, Distribution properties and arithmetic, Basic machine learning algorithms, Linear regression, SVM, Naive Bayes.

**UNIT-IV: Data Visualisation-** Introduction, Types of data visualisation, **Data for visualisation-** Data types, Data encodings, Retinal variables, Mapping variables to encodings, Visual encodings.

**UNIT-V: Applications of Data Science-** Technologies for visualisation, Bokeh (Python), recent trends in various data collection and analysis techniques, various visualization techniques, application development methods of used in data science.

**TEXT BOOKS:**

1. "The Art of Data Science", 1st edition, Roger D. Peng and Elizabeth matsui, Lean Publications, 2015
2. "Algorithms for Data Science", 1st edition, **Steele**, Brian, **Chandler**, John, **Reddy**, Swarna, springers Publications, 2016

**REFERENCE BOOKS:**

1. Doing Data Science: Straight Talk From The Frontline, 1st edition, Cathy O'Neil and Rachel Schutt, O'Reilly, 2013
2. Mining of Massive Datasets, 2nd edition, Jure Leskovek, Anand Rajaraman and Jeffrey Ullman, v2.1, Cambridge University Press, 2014

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### BLOCK CHAIN TECHNOLOGIES

**Course Objectives:**
To understand block chain technology and Cryptocurrency works

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

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Demonstrate the block chain basics, Crypto currency | **K2** |
| **CO2** | To compare and contrast the use of different private vs. public block chain and use cases | **K6** |
| **CO3** | Design an innovative Bit coin Block chain and scripts, Block chain Science on varies coins | **K5** |
| **CO4** | Classify Permission Block chain and use cases – Hyper ledger, Corda | **K2** |
| **CO5** | Make Use of Block-chain in E-Governance, Land Registration, Medical Information Systems, and others | **K3** |

**#based on suggested Revised BTL**

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | | 1 | 1 | 1 | 2 | 2 | | | 2 | 1 | | 1 |
| **CO2** | 1 | 2 | 1 | 2 | 1 | 3 | | | | 2 | 1 | |
| **CO3** | 2 | 2 | 2 | 2 | 2 | 1 | | | 2 | | | |
| **CO4** | 2 | 1 | 3 | 3 | 3 | 2 | | | 3 | | 1 | |
| **CO5** | 1 | 2 | | 2 | 3 | | | | 2 | 1 | | |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**SYLLABUS:**

**UNIT- I Introduction**: Introduction – basic ideas behind block chain, how it is changing the landscape of digitalization, introduction to cryptographic concepts required, Block chain or distributed trust, Currency, Cryptocurrency, How a Cryptocurrency works, Financial services, Bitcoin prediction markets.

**UNIT- II:** Hashing, public key cryptosystems, private vs public block chain and use cases, Hash Puzzles, Extensibility of Block chain concepts, Digital Identity verification, Block chain Neutrality, Digital art, Block chain Environment

**UNIT- III Introduction to Bitcoin :** Bitcoin Block chain and scripts, Use cases of Bitcoin Blockchain scripting language in micropayment, escrow etc Downside of Bit coin – mining, Block chain Science: Grid coin, Folding coin, Block chain Genomics, Bit coin MOOCs.

**UNIT – IV :** Ethereum continued, IOTA, The real need for mining – consensus – Byzantine Generals Problem, and Consensus as a distributed coordination problem – Coming to private or permissioned block chains –Introduction to Hyper ledger, Currency, Token,Campus coin, Coin drop as a strategy for Public adoption, Currency Multiplicity, Demurrage currency

**UNIT – V :** Technical challenges, Business model challenges, Scandals and Public perception, Government Regulations, Uses of Block chain in E-Governance, Land Registration, Medical Information Systems.

**TEXT BOOK:**
1. Blockchain Blue print for Economy by Melanie Swan

**REFERENCES:**
1. Blockchain Basics: A Non-Technical Introduction in 25 Steps 1st Edition, by Daniel Drescher

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### GAME THEORY

**Course Objectives:**
To understand Games and Solutions Game Theory, Electronic Mail Game.

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

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Explain the game theory terminology and solutions of games. | **K4** |
| **CO2** | Compare with Mixed, Correlated, and Evolutionary Equilibrium. | **K5** |
| **CO3** | To elaborate the uses of Knowledge and Equilibrium | **K3** |
| **CO4** | Discuss the Extensive Games with Perfect Information. | **K5** |
| **CO5** | Demonstrate The Basic Idea Infinitely Repeated Games | **K3** |

**#based on suggested Revised BTL**
**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 1 | | 2 | 1 | 3 | 1 | | | 2 | 1 | 2 | 1 |
| **CO2** | 1 | 2 | 1 | | 3 | 2 | | | | 2 | 1 | |
| **CO3** | 2 | | 2 | 2 | | 1 | | | 2 | | 2 | |
| **CO4** | 1 | 1 | 3 | | 3 | 2 | | | 3 | 2 | 1 | 1 |
| **CO5** | 2 | 2 | | 2 | 3 | | | | 2 | 1 | 2 | |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**SYLLABUS:**
**UNIT – I: Introduction:** Game Theory, Games and Solutions Game Theory and the Theory of Competitive Equilibrium, Rational Behavior, The Steady State and Deductive Interpretations, Bounded Rationality Terminology and Notation Nash Equilibrium- Strategic Games, Strictly Competitive Games.

**UNIT – II:** Mixed, Correlated, and Evolutionary Equilibrium-Mixed Strategy Nash Equilibrium Interpretations of Mixed Strategy Nash Equilibrium Correlated Equilibrium Evolutionary Equilibrium Rationalizability and Iterated Elimination of Dominated Actions.

**UNIT – III:** Knowledge and Equilibrium –A Model of Knowledge Common Knowledge, Can People Agree to Disagree? , Knowledge and Solution Concepts, The Electronic Mail Game.

**UNIT – IV:** Extensive Games with Perfect Information –Extensive Games with Perfect Information Subgame Perfect Equilibrium Two Extensions of the Definition of a Game The Interpretation of a Strategy , Two Notable Finite Horizon Games ,

**UNIT – V:** Repeated Games –The Basic Idea Infinitely Repeated Games vs.\ Finitely Repeated Games Infinitely Repeated Games: Definitions Strategies as Machines

Trigger Strategies: Nash Folk Theorems Punishing for a Limited Length of Time: A Perfect Folk Theorem for the Limit of Means Criterion Punishing the Punisher:

**TEXT BOOKS:**

1. M. J. Osborne and A. Rubinstein, A course in Game Theory, MIT Press
2. Roger Myerson, Game Theory, Harvard University Press
3. D. Fudenberg and J. Tirole, Game Theory, MIT Press

**REFERENCES:**

1. J. von Neumann and O. Morgenstern, Theory of Games and Economic Behavior, New York:n John Wiley and Sons.
2. R.D. Luce and H. Raiffa, Games and Decisions, New York: John Wiley and Sons.,
3. G. Owen, Game Theory, (Second Edition), New York: Academic Press,

## INTERNET OF THINGS

**Course Objectives:**
From the course the student will learn

- Able to understand the application areas of IOT
- Able to realize the revolution of Internet in Mobile Devices, Cloud & Sensor Networks
- Able to understand building blocks of Internet of Things and characteristics

**Course Outcomes:**

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Review Internet of Things (IoT). | **K2** |
| **CO2** | Demonstrate various business models relevant to IoT. | **K3** |
| **CO3** | Construct designs for web connectivity | **K6** |
| **CO4** | Organize sources of data acquisition related to IoT, integrate to enterprise systems. | **K4** |
| **CO5** | Describe IoT with Cloud technologies. | **K2** |

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 1 | | 3 | | 2 | 1 | | | 2 | 1 | 2 | 1 |
| **CO2** | 2 | 2 | 3 | | 1 | 2 | | | | 2 | 1 | |
| **CO3** | | 1 | 2 | 2 | | 1 | | | 2 | | 2 | |
| **CO4** | 1 | 1 | 3 | | 2 | 1 | | | 3 | 2 | 1 | 1 |
| **CO5** | 2 | 2 | | 2 | | 1 | | | 3 | 1 | 2 | |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**SYLLABUS:**

**UNIT–I: The Internet of Things-** An Overview of Internet of things, Internet of Things Technology, behind IoTs Sources of the IoTs, Examples OF IoTs, Design Principles For Connected Devices, Internet connectivity, **Application Layer Protocols-** HTTP, HTTPS, FTP

**UNIT–II:** Business Models for Business Processes in the Internet of Things, IoT/M2M systems LAYERS AND designs standardizations, Modified OSI Stack for the IoT/M2M Systems ,ETSI M2M domains and High-level capabilities, Communication Technologies, Data Enrichment and Consolidation and Device Management Gateway Ease of designing and affordability.

**UNIT-III:** Design Principles for the Web Connectivity for connected-Devices, Web Communication protocols for Connected Devices, Message Communication protocols for Connected Devices, Web Connectivity for connected-Devices.

**UNIT-IV:** Data Acquiring, Organizing and Analytics in IoT/M2M, Applications/Services/Business Processes, IOT/M2M Data Acquiring and Storage, Business Models for Business Processes in the Internet Of Things, Organizing Data, Transactions, Business Processes, Integration and Enterprise Systems.

**UNIT-V:** Data Collection, Storage and Computing Using a Cloud Platform for IoT/M2M Applications/Services, Data Collection, Storage and Computing Using cloud platform Everything as a service and Cloud Service Models, IOT cloud-based services using the Xively (Pachube/COSM), Nimbits and other platforms Sensor, Participatory Sensing, Actuator, Radio Frequency Identification, and Wireless, Sensor Network Technology, Sensors Technology, Sensing the World.

**TEXT BOOKS:**

1. Internet of Things: Architecture, Design Principles And Applications, Rajkamal, McGraw Hill Higher Education
2. Internet of Things, A.Bahgya and V.Madisetti, Univesity Press, 2015

**REFERENCE BOOKS:**

1. Designing the Internet of Things, Adrian McEwen and Hakim Cassimally, Wiley
2. Getting Started with the Internet of Things, CunoPfister , Oreilly

# HONORS COURSES

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### HONORS COURSES

**(Tentative list be appended)**

**Note**

1. **The subjects opted for HONORS should be Advanced type which are not covered in regular curriculum**
2. **Students have to acquire 16 credits with minimum one subject from each pool.**
3. **Concerned BoS can add or delete the subjects as per the decision of the board.**
4. **Pre requisites to be defined by the board for each course.**
5. **Compulsory MOOC/NPTEL Courses for 04 credits (02 courses@ 2 credits each)**

| | |
|---|---|
| **POOL1**<br>1. Network Architecture and Design<br>2. Operating Systems Administration and Security<br>3. E-Commerce<br>4. Software Architecture and Design Patterns | **POOL 2**<br>1. Neural Networks<br>2. Security Governance Risk and compliance<br>3. Randomized algorithms<br>4. Data Visualization |
| **POOL3**<br>1. Cloud and IoT Security<br>2. No SQL Databases<br>3. Soft computing<br>4. Social Networks & Semantic Web | **POOL4**<br>1. Digital Marketing<br>2. Wireless Networks Security<br>3. High performance Computing<br>4. Multi Agent Systems |

Department of Computer Science & Engineering
University College of Engineering, JNT University Kakinada

**POOL1:**

## NETWORK ARCHITECTURE AND DESIGN

**Course Objectives:**

- The goal of this course is to give a background in computer networks mechanisms and basic concepts used in the design of protocols and network architectures.
- These mechanisms include any of the protocol stack layers and a wide range of technologies either for wired networks or for wireless networks Outline.

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

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | **Evolution** of the Network Architecture | **K5** |
| **CO2** | **Analyze** Routing and Inter-Networking Routing Algorithms | **K4** |
| **CO3** | **Analyze** Network and Transport Protocols IPv6. IPv4-IPv6 coexistence | **K4** |
| **CO4** | **Discuss** Resource Management Quality of Service principles. Quality of Service and Quality of Experience (QoS and QoE). | **K6** |
| **CO5** | **Determine** Software Defined Networks | **K5** |

***#Based on suggested Revised BTL***

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO10 | PO10 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | | 3 | 2 | | 1 | | | | | | | |
| CO2 | 2 | 3 | 2 | 1 | | | | | | | | |
| CO3 | | 3 | 2 | | 1 | | | | | | | |
| CO4 | 3 | 3 | 2 | | 2 | | | | | | | |
| CO5 | 2 | 3 | 2 | | 2 | | | | | | | |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**SYLLABUS:**

**UNIT-I: Evolution of the Network Architecture:** Internet Design Principles, Key protocols and their evolution, Internet structure.

**Trends in the Evolution of the Network Architecture**: Naming and Addressing, Addressing and Routing, Mobility, New Network Architectures.

**UNIT-II:** Routing and Inter-Networking Routing Algorithms, Classless Inter-domain Routing, Inter-domain Routing, IDR, BGP, IBGP, BGP attributes, Scalability of BGP, Transport Network (Backbone) Optical Transport Network, IP over SDH, IP over WDM/ASON, IP over WDM/Gigabit Ethernet.

**UNIT-III:** New Network and Transport Protocols IPv6, IPv4-IPv6 coexistence, Mobile-IP, IP-Multicast, Other IP protocols (HIP), Multipath-TCP, Other Transport protocols (QUIC).

**UNIT-IV:** Resource Management Quality of Service principles, Quality of Service and Quality of Experience (QoS and QoE), Integrated Services Architecture, Differentiated Services Architecture.

**UNIT-V:** 5G Network Architecture Framework for 5G networks, Core, edge and access networks, Virtualization technologies (NFV), Software Defined Networks (SDN), Orchestration and management, Slicing.

**TEXT BOOK:**
1. Architecture of Network Systems "The Morgan Kaufmann Series in Computer Architecture and Design", 1st Edition

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### OPERATING SYSTEMS ADMINISTRATION & SECURITY

**Course Objectives:**

- To know some basic security measures to take in system administration.
- To prepare for possible disasters, including an understanding of backup and Restoration of file systems.

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

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| CO1 | **Describe** the important computer system resources and the role of operating system in their management policies and algorithms. | K1 |
| CO2 | **Explain** the concepts of Access control Fundamentals, Multics. | K2 |
| CO3 | **Identify** and assess current and anticipated security risks and vulnerabilities. | K3 |
| CO4 | **Identify** the security Techniques and apply the real time applications. | K3 |
| CO5 | **Know** the role and responsibilities of a system administrator and Create and administer user accounts on both a Linux and Windows platform. | K3 |

***# Based on suggested Revised BTL***

**Mapping of course outcomes with program outcomes**

| CO | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | | 3 | 2 | | 1 | | | | | | | |
| CO2 | | 3 | 2 | 1 | | | | | | | | |
| CO3 | | 3 | 2 | | 1 | | | | | | | |
| CO4 | 3 | 2 | | 2 | | | | | | 1 | | 1 |
| CO5 | 3 | 2 | | 2 | | | 1 | | | | | |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**SYLLABUS:**

**UNIT–I: Overview of Operating Systems-** Introduction, Computer system Organization and Architecture, Operating System structure and operations, Principles and design of process, memory, and file systems management.

**Introduction to real-time systems**: The history of real-time programming, What is a real-time system?, Case study: anti-lock braking system, Components of real-time systems.

**UNIT–II: Access Control Fundaments-**Access Control Fundaments, Protection systems, Lampson’s access matrix, mandatory protection systems, Reference monitor, Secure operating system definition, Assessment criteria’ MULTICS, security fundamentals, protection systems, access control lists reference monitor, security, vulnerability analysis

**UNIT–III: Operating System Security-**Security in Windows and UNIX: Protection system, Authorization, Security analysis and vulnerabilities, Verifiable
**Security goals**: Information flow – secrecy models – integrity models- convert channels, **Security kernel:** Secure Communication processor architecture hardware – SCOMP trusted OS-Gemini Secure OS, VM systems.

**UNIT–IV: Computer Security Techniques-**Authentication, Access control, intrusion detection, malware defences, buffer overflow attacks, UNIX and Windows security, **OS patch management**- Automates Windows patching, mitigates security risks and fixes vulnerabilities in minutes.

**UNIT–V: Linux Administration with Shell Scripts-** Introduction, Linux Kernel architecture, system administration tasks in linux managing user accounts-.rc files, device management disk space allocation automation system administration with shell scripts

**Text Books**

1. Abraham Silberschatz,Peter Baer Galvin, Greg Gagne, "Operating System Concepts",9th Edition,Wiley Publication,2008.
2. William Stalling, Operating System: Internals and Design Principles, Prentice Hall, 7th Edition, 2012.
3. A_practical_introduction_to_realtime_systems_for_undergraduate_engineerin g.pdf

**Reference Books**:

1. Mukesh Singhal and N. G. Shivaratri, Advanced Concepts in Operating Systems, McGraw- Hill, 2000
2. Promod Chandra P Bhat, An Introduction to Operating Systems: Concepts and practice, Prentice hall of India, 4th Edition, 2014
3. Tom Adelstein and Bill Lubanovic, Linux System Administration, O'Reilly Media, Inc.,

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### E-COMMERCE

**Course Objective:**

- This course provides an introduction to information systems for business and management.
- It is designed to familiarize students with organizational and managerial foundations of systems, the technical foundation for understanding information systems

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

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | **Design and implement** an e-commerce application with a shopping cart. | **K6** |
| **CO2** | **Explain** the effectiveness of network computing and cloud computing policies in a multi- location organization. | **K3** |
| **CO3** | **Analyze** real business cases regarding their e-business strategies and transformation processes and choices. | **K4** |
| **CO4** | **Identify** the e-businesses Consumer oriented e-Commerce process. | **K4** |
| **CO5** | **Apply** Web advertising and Web publishing in Internet | **K5** |

***#Based on suggested Revised BTL***

**Mapping of course outcomes with program outcomes**

| CO | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 1 | | 3 | | | | | 2 | | 1 | | |
| CO2 | 1 | 2 | 3 | | | 1 | | | | 2 | | |
| CO3 | 2 | 3 | | | 1 | | | | 2 | | | |
| CO4 | | 2 | 3 | | | 1 | | | | 2 | | |
| CO5 | 2 | 3 | | | 1 | | | | 2 | | | |

**(Please fill the above with Levels of Correlation, viz., L, M, H)**

**SYLLABUS:**

**UNIT-I: Electronic Commerce Framework –** Electronic Commerce and Media convergence. The Network Infrastructure for Electronic Commerce: Components of the I-Way-Network Access Equipment-Global Information Distribution Networks.

**UNIT-II: Electronic Payment Systems**: Overview of Electronic Payment Systems – Smart card and e-Payment system – Credit Cards based e-Payment system, risk and e-Payment system, Designing e-Payment system. Corporate Digital Library: Dimension of Internal Electronic Commerce Types of Digital Documents- Issues behind Document Infrastructure. Corporate data Warehouses

**UNIT-III: Internet Payment System:** Characteristics of Payment system – 4C Payment methods – SET Protocol for credit card payment – e-Cash – e-Check – Micropayment system – Overview of smart card and Mondex. e-Services: Categories of e-Services – Web enabled services – Match making services – Information selling on the web – e-Entertainment – Auctions & other specialized services. E-Governance.

**UNIT-IV: Consumer oriented e-Commerce: Introduction –** Traditional retailing and e-Retailing – Benefits of e-Retailing – Key success factors – Models of e-Retailing – Features – Developing a consumer-oriented e-Commerce system – The PASS Model. Business oriented e-Commerce: Features – Business Models – Integration. Examples of the types of e-Commerce: Intel - Amazon – e-bay – Priceline.

**UNIT-V:Web advertising and Web publishing :** Traditional Vs. Internet advertising – Internet advertising techniques and strategies – Business models for advertising and their revenue streams – Pricing models – Web publishing – Website development methodologies – Logical design of the user interface I & II – Usability testing and quality assurance –Step by step exercise for building the Virtual Book Store (VBS).

**TEXT BOOKS**:

1. E- Commerce Fundamentals and Applications, Henry Chan, Raymond Lee, Tharam Dillon and Elizabeth Chang, Wiley India Pvt. Ltd., Third reprint 2007.
2. Frontiers of Electronic commerce, Ravi Kalakota and Andrew B. Whinston, Pearson Publication Ltd, 2013, V Edition.

**REFERENCE BOOKS**:

1. E-Commerce – Concepts, Models, Strategies, CSV Murthy , Himalaya Publishing House, 1st Ed 2011
2. E-Commerce Business Technology. Society, Schneider, Thomson Publication
3. E-Commerce the cutting edge of business, Kamlesh K Bajaj and Debjani Nag, Tata McGraw Hill , Sixth reprint 2008

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### SOFTWARE ARCHITECTURE AND DESIGN PATTERNS

**Course Objectives:** The main objective is to introduce the student to architecture of software and design Patterns. The other objectives include

- Understand the creational and structural patterns.
- Be capable of applying his knowledge to create an architecture for given application.

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

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| CO1 | **Design and Create** the architecture, creating it and moving from one to any, different structural patterns | K1,K5 |
| CO2 | **Analyze** the architecture and build the system from the components | K4 |
| CO3 | **Design** creational and structural patterns. | K5 |
| CO4 | **Learn** about the behavioral patterns. | K1 |
| CO5 | **Discuss** about case study in utilizing architectural structures. | K6 |

***#Based on suggested Revised BTL***

**Mapping of course outcomes with program outcomes**

| CO | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 2 | 1 | 3 | | 2 | | | | | | | |
| CO2 | 1 | 3 | 2 | | 2 | 1 | | | | | | |
| CO3 | | | 3 | | 3 | | | 2 | | | 1 | |
| CO4 | 2 | 3 | | 2 | | | | 1 | | | 1 | |
| CO5 | 1 | 1 | 2 | | | 2 | | 2 | | | 3 | |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**SYLLABUS:**

**UNIT- I: Envisioning Architecture:** The Architecture Business Cycle, What is Software Architecture, Architectural patterns, reference models, reference architectures, architectural structures and views.

**Creating architecture:** Quality Attributes, Achieving qualities, Architectural styles and patterns, designing the Architecture, Documenting software architectures, Reconstructing Software Architecture

**UNIT –II: Analyzing Architectures:** Architecture Evaluation, Architecture design decision making, ATAM, and CBAM. **Moving from one system to many:** Software Product Lines, Building systems from off the shelf components, Software architecture in future.

**UNIT-III: Patterns:** Pattern Description, Organizing catalogs, role in solving design problems, Selection and usage. **Creational and Structural patterns:** Abstract factory, builder, factory method, prototype, singleton, adapter, bridge, composite, façade.
**UNIT- IV: Behavioral patterns:** Chain of responsibility, command, Interpreter, iterator, mediator, memento, observer, state, strategy, template method, visitor.

**UNIT –V: Case Studies:** A-7E – A case study in utilizing architectural structures, The World Wide Web - a case study in interoperability, Air Traffic Control – a case study in designing for high availability, Celsius Tech – a case study in product line development

**TEXT BOOKS:**

1. Len Bass,Paul Clements&Rick Kazman, *Software Architecture in Practice*, 2nd Edition, Pearson Education, 2003.
2. Erich Gamma, *Design Patterns*, 1st Edition, Pearson Education, 1995.

**REFERENCE BOOKS:**

1. Luke Hohmann , *Beyond Software architecture*, Addison wesley, 2003.
2. David M. Dikel, David Kane and James R. Wilson, *Software architecture*, 1st Edition, Prentice Hall,2001
3. F.Buschmann , *Pattern Oriented Software Architecture*, Wiley&Sons,1st Edition,2001

**Web References:**

1. http://en.wikibooks.org/wiki/Introduction_to_Software_Engineering/Architecture/Design_Patterns

**Department of Computer Science & Engineering**
**University College of Engineering, JNT University Kakinada**

**POOL2:**

## NEURAL NETWORKS

**Course Objectives:**

- The main objective of Neural Network Techniques to Improve Data Analysis Solutions is to strengthen the dialogue between the statistics and soft computing research communities in order to cross-pollinate both fields and generate mutual improvement activities.
- Also introduce the neural networks for classification, regression and to give design methodologies for artificial neural networks.

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

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| CO1 | **Explain** Fundamentals of Neural Networks | K2,K5 |
| CO2 | **Explain** Architecture of Back-propagation(BP) Networks | K5 |
| CO3 | **Design** Associative Memory for Real coded pattern pairs | K6 |
| CO4 | **Design** ART1 and ART2 Architecture and algorithms | K6 |
| CO5 | **Compare** Hybrid system and Neural Networks | K2 |

***#Based on suggested Revised BTL***

**Mapping of course outcomes with program outcomes**

| CO | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 3 | 1 | | 2 | | | | | | | | |
| CO2 | 1 | 3 | 2 | | | | | | | 1 | | |
| CO3 | 1 | 2 | 3 | | 1 | | | | | | | |
| CO4 | | 2 | 3 | | 1 | | | 1 | | | | |
| CO5 | 1 | | | 1 | | | | | | | 1 | |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**SYLLABUS:**

**UNIT-I: Introduction:** Neural Network, Fuzzy logic, Genetic Algorithm. **Fundamentals of Neural Networks**: What is Neural Network, Model of Artificial Neuron, Learning rules and various activation functions.

**UNIT-II: Neural Network Architecture:** Single layer Feed-forward networks. Multilayer Feed-forward networks. Recurrent Network , **Back propagation Networks:** Back Propagation networks, Architecture of Back-propagation(BP) Networks, Back-propagation Learning, Variation of Standard Back propagation algorithms.

**UNIT-III: Associative Memory:** Auto correlators, Heterocorrelators, Wang et al's Multiple Training Encoding Strategy, Exponential BAM, Associative Memory for Real coded pattern pairs, Applications.

**UNIT-IV: Adaptive Resonance Theory:** Cluster Structure, Vector Quantization, Classical ART Network, Simplified ART Architecture,ART1 and ART2 Architecture and algorithms, Applications, Sensitivities of ordering of data.

**UNIT-V: Introduction about Fuzzy set theory:** Fuzzy versus Crisp, Crisp and fuzzy sets, Crisp and Fuzzy relations, **Integration of Neural Network, Fuzzy logic and Genetic Algorithm:** Hybrid system. Neural Networks, Fuzzy logic, and Genetic Algorithm Hybrids.

**TEXT BOOKS:**

1. Bishop, C. M. Neural Networks for Pattern Recognition. Oxford University Press. 1995.
2. Neural Networks, Fuzzy Logic and Genetic Algorithms, by S.Rajasekaran and G.A. Vijayalakshmi Pai.
3. Neuro-Fuzzy Systems, Chin Teng Lin, C. S. George Lee, PHI.
   Build_ Neural_ Network_ With_ MS_ Excel_ sample by Joe choong

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### SECURITY GOVERNANCE RISK AND COMPILANCE

**Course Objectives:**
- Knowledge and understanding of the different theories on cyber-governance, the implications of cyberspace
- Understanding the internet for traditional notions such as sovereignty, power, war and conflict, terrorism and crime.
- Understanding the historical developments in cyber governance and how key events have led to the current state of affairs.

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

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | **Demonstrate** the fundamental concepts and principles of the cyber Security Governance and theories of governance. | **K2** |
| **CO2** | **Illustrate** the metrics of Cyber Security Governance and Intrusion detection and prevention. | **K2** |
| **CO3** | **Measure** Measurement of Governance and Vulnerability metrics. | **K5** |
| **CO4** | **Model** the Cyber security governance risk management, which involves mitigating risks and reducing or preventing potential impact on information resources and Risk Assessment Policy**.** | **K3** |
| **CO5** | **Choose** the Governance Mechanism and Compliance Decision Making Process. | **K6** |

***#Based on suggested Revised BTL***

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 1 | 3 | 2 | | | | 1 | | | | | |
| **CO2** | | 2 | 3 | | 1 | | | 1 | | 2 | | |
| **CO3** | | | | | | | | | | | | |
| **CO4** | 1 | 2 | | | 3 | 1 | | | 1 | | | |
| **CO5** | | | | 2 | | 1 | | | | 1 | | |

**(Please fill the above with Levels of Correlation, viz., L-1, M-2, H-3)**

**SYLLABUS:**

**UNIT–I: Cyber security Governance**-Principles of cyber security governance, Assessment of cyber security maturity. Theories of governance: introduction, Governance – definitions and typologies, Tools, methods and processes.
**Organizational Strategic Governance Framework:** Inter-Organizational development, An Organizational Strategic Governance Framework

**UNIT–II: Network Device metrics**-Vulnerability management, Threat management, Endpoint management, Intrusion detection and prevention (IDPS), Security incident management, Security operations centre (SOC) and related concepts.

**UNIT–III: Measurement of Governance**-Metrics – concepts, Application security metrics, Network security metrics, Security incident metrics, Vulnerability metrics, Service level objectives / agreement (SLO / SLA), NIST metrics.

**UNIT–IV: Communication Risk Management Strategy:** Introduction, culture and communication, Business continuity management planning revisited, Communication Risk Management strategy.
**Risk Assessment Policy and its Strategic Context:** Introduction to Vulnerability, Risk Assessment Policy, Strategic Management Framework, Cyber Security Strategy.

**UNIT–V: Integrated Governance Mechanism:** Introduction, Placing Corporate Governance in Context, Organizational Commitment to Corporate Governance, Corporate Governance Revisited.
**Governance and Compliance Decision Making Process**: Introduction, Linking Organizational Resilience with Corporate Governance, Organizational Decision Making Approach, New Business Models, Information Security Governance

**TEXT BOOKS:**

1. Cyber Security Management: A Governance, Risk and Compliance Framework, Peter Trim, Yang-Im Lee
2. IT Security Metrics: A Practical Framework for Measuring Security & Protecting Data, Hayden, lance, McGraw-hill education group, 2010

**REFERENCE BOOKS:**

1. Cyber Security, Critical Infrastructure, Framework for Improving Critical Infrastructure Cyber Security, Framework, 1st Edition 2014
2. Data-Driven Security: Analysis,1st Edition, Visualization and Dashboards, Jacobs, Jay, and Bob Rudis, John wiley & sons, 2014

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### RANDAMIZED ALGORITHEM

**Pre-requisite: Basics of DAA, C**

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

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | **Design and analyze** efficient randomized algorithms | **K2** |
| **CO2** | **Apply** the application tail inequalities to bound error-probability | **K2** |
| **CO3** | **Analyze** randomized algorithms with respect to probability of error and expected running time | **K3** |
| **CO4** | **Analyze** approximation algorithms and determine approximation factor. | **K3** |
| **CO5** | **Apply** the graph algorithms in real time applications | **K6** |

***#Based on suggested Revised BTL***

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | | 3 | 2 | 2 | | 1 | | | | | | |
| **CO2** | 3 | | 2 | | | 1 | | | | | | |
| **CO3** | 3 | 2 | 2 | | 1 | | | | | | | |
| **CO4** | | 2 | 3 | 2 | 2 | | 1 | | | | | |
| **CO5** | 3 | 2 | 2 | | 1 | | 1 | | | | | |

**(Please fill the above with Levels of Correlation, viz., L-1, M-2, H-3)**

**SYLLABUS:**

**Unit-1:Randomized Algorithms Introduction**– Sample Space, Events, Conditional probability, Independent events, Random Variables, Linearity of Expectation , Los vegas and Moute Carlo-Chernoff Bound – Probabilistic Recurrence– Typical randomised algorithms (e.g. Min cut, Randomised Quick Sort, Randomised Selection, Primdity testing), Computational Model and Complexity Classes

**Unit-2: Game-Theoretic Techniques:** Game Tree Evaluation, The Minimax Principle Moments and Deviations : Markov and Chebyshev Inequalities, Randomized Selection, Two-Point Sampling, The Coupon Collector's Problem

**Unit-3: The Probabilistic Method:** Overview, probabilistic analysis, use of indicator random variables,analysis using indicator random variables, The Method of Conditional Probabilities, Markov Chains and Random Walks- A 2-SAT Example, Markov Chains, Random Walks on Graphs, Graph Connectivity.

**Unit-4: Randomized data structures and algorithms :** Skip lists, Hashing, Randomized min-cut, Verifying matrix multiplication, Randomized quicksort, Randomized selection, Coupon Collector's algorithm, Randomized pattern matching. **Number theoretic algorithms:** Primality testing -- Miller Rabin test.

**Unit-5: Graph Algorithms** - All-pairs Shortest Paths ,Online Algorithms- The Online Paging Problem, Adversary Models, Relating the Adversaries, The Adaptive Online Adversary

**TEXT BOOKS:**

1. Rajeev Motwani and Prabhakar Raghavan, Randomized Algorithms, Cambridge University Press,1995.
2. J. Hromkovic, Design and Analysis of Randomized Algorithms, Springer,2005.

**REFERENCES:**

1. T. H. Cormen, C. E. Leiserson, and R. L. Rivest, "Introduction to Algorithms", The MIT press, Cambridge, Massachusetts and McGraw Hill, 1990.
2. H. S. Wilf, Algorithms and complexity, Prentice hall.

**Resources:** https://nptel.ac.in/courses/106/103/106103187/

## DATA VISUALIZATION

**Pre-Requisites: Computer Graphics, Image Processing**
**Course Objective:**

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

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

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | **Explain** Visualization and representation of data | **K6** |
| **CO2** | **Creating** visual representations and visualization reference model of applications | **K3** |
| **CO3** | **Classify** the visualization systems in a data representation | **K4** |
| **CO4** | **Identify** Visualization of groups and trees | **K3** |
| **CO5** | **Determine** the visualization of volumetric different data sets in applications | **K6** |

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | | 3 | 2 | | 1 | | 1 | | | 1 | | |
| CO2 | 2 | 1 | 2 | | 2 | 1 | | | | | | |
| CO3 | | 3 | 2 | | | | | | | 1 | | |
| CO4 | 1 | 2 | 3 | 1 | 2 | | | | | | | |
| CO5 | 1 | 1 | 2 | | | 2 | | 2 | | 1 | | |

**(Please fill the above with Levels of Correlation, viz., L-1, M-2, H-3)**

**SYLLABUS:**

**UNIT-1: Introduction**: What Is Visualization?, History of Visualization, Relationship between Visualization and Other Fields The Visualization Process, Introduction of visual perception, visual representation of data, Gestalt principles, information overloads.

**UNIT-2:** Creating visual representations, visualization reference model, visual mapping, visual analytics, Design of visualization applications

**UNIT-3:** Classification of visualization systems, Interaction and visualization techniques misleading, Visualization of one, two and multi-dimensional data, text and text documents.

**UNIT-4:** Visualization of groups, trees, graphs, clusters, networks, software, Metaphorical visualization

**UNIT-5:** Visualization of volumetric data, vector fields, processes and simulations, Visualization of maps, geographic information, GIS systems, collaborative visualizations, Evaluating visualizations
**Recent trends** in various perception techniques, various visualization techniques, data structures used in data visualization.

**TEXTBOOK:**

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

**Resources:**

1. https://kdd.cs.ksu.edu/Courses/CIS536/Lectures/Slides/Lecture-34-Main_6up.pdf

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**POOL 3:**

### CLOUD & IoT SECURITY

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

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | **Discuss** about Security Requirements in IoT Architecture | **K6** |
| **CO2** | **Explain** Random number generation | **K3** |
| **CO3** | **Demonstrate** Authorization with Publish / Subscribe schemes | **K6** |
| **CO4** | **Identify** Lightweight and robust schemes for Privacy protection | **K3** |
| **CO5** | **Explain** about IoT cloud security architecture | **K6** |

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 1 | 2 | 3 | 2 | | 1 | | | | | | |
| **CO2** | | 3 | 2 | 2 | | 1 | | | | | | |
| **CO3** | 3 | | 2 | | | 1 | | | | | | |
| **CO4** | 3 | 2 | 2 | | 1 | | | | | | | |
| **CO5** | 1 | 2 | 3 | 2 | | 1 | | | | | | |

**(Please fill the above with Levels of Correlation, viz., L-1, M-2, H-3)**

**SYLLABUS:**

**UNIT I – INTRODUCTION: SECURING THE INTERNET OF THINGS:** Security Requirements in IoT Architecture - Security in Enabling Technologies – Security Concerns in IoT Applications. Security Architecture in the Internet of Things – Security Requirements in IoT - Insufficient Authentication /Authorization – Insecure Access Control - Threats to Access Control, Privacy, and Availability - Attacks Specific to IoT. Vulnerabilities – Secrecy and Secret-Key Capacity - Authentication/Authorization for Smart Devices - Transport Encryption – Attack, Fault trees

**UNIT II- CRYPTOGRAPHIC FUNDAMENTALS FOR IOT:** Cryptographic primitives and its role in IoT – Encryption and Decryption – Hashes –Digital Signatures – Random number generation – Cipher suites – key management fundamentals – cryptographic controls built into IoT messaging and communication protocols.

**Department of Computer Science & Engineering**
**University College of Engineering, JNT University Kakinada**

**UNIT III- IDENTITY & ACCESS MANAGEMENT SOLUTIONS FOR IOT**: Identity lifecycle – authentication credentials – IoT IAM infrastructure – Authorization with Publish / Subscribe schemes and access control

**UNIT IV-PRIVACY PRESERVATION AND TRUST MODELS FOR IOT:** Concerns in data dissemination – Lightweight and robust schemes for Privacy protection – Trust and Trust models for IoT – self-organizing Things – Preventing unauthorized access.

**UNIT V - CLOUD SECURITY FOR IOT:** Cloud services and IoT – offerings related to IoT from cloud service providers – Cloud IoT security controls – enterprise IoT cloud security architecture – New directions in cloud enabled IoT computing

**TEXT BOOKS:**

1. Practical Internet of Things Security (Kindle Edition) by Bria Russell, Drew VanDuren

**REFERENCES BOOKS:**

1. Securing the Internet of Things Elsevier
2. Security and Privacy in Internet of Things (IoTs): Models, Algorithms, and Implementations

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### NO SQL DATABASES

**Course Objective:** The student will be able to

- Define, compare and use the four types of NoSQL Databases (Document-oriented, Key-Value Pairs, Column-oriented and Graph).
- Demonstrate an understanding of the detailed architecture, define objects, load data, query data and performance tune Column-oriented NoSQL databases.
- Explain the detailed architecture, define objects, load data, query data and performance tune Document-oriented NoSQL databases.

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

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| CO1 | **Discuss** about Aggregate Data Models | K6 |
| CO2 | **Explain** about Master-Slave Replication, Peer-to-Peer Replication | K3 |
| CO3 | **Describe** the Structure of Data, Scaling, Suitable Use Cases | K2 |
| CO4 | **Make use of** Complex Transactions Spanning Diferent Operations | K2 |
| CO5 | **Identify** Routing, Dispatch and Location-Based Services | K3 |

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 1 | 2 | 3 | 2 | | 1 | | | | | | |
| CO2 | 2 | 3 | 2 | | 1 | | | | | | | |
| CO3 | | 3 | 1 | 2 | | 1 | | | | | | |
| CO4 | | 3 | 2 | 2 | | 1 | | | | | | |
| CO5 | 1 | 2 | 3 | 2 | | 1 | | | | | | |

**(Please fill the above with Levels of Correlation, viz., L, M, H)**

**SYLLABUS:**

**UNIT-I:** Why NoSQL? The Value of Relational Databases, Getting at Persistent Data, Concurrency, Integration, A (Mostly) Standard Model, Impedance Mismatch, Application and Integration Databases, Attack of the Clusters, The Emergence of NoSQL, Aggregate Data Models; Aggregates, Example of Relations and Aggregates, Consequences of Aggregate Orientation, Key-Value and Document Data Models, Column-Family Stores, Summarizing Aggregate-Oriented Databases. More Details on Data Models; Relationships, Graph Databases, Schemaless Databases, Materialized Views, Modelling for Data Access,

**Department of Computer Science & Engineering**
**University College of Engineering, JNT University Kakinada**

**UNIT-II: Distribution Models:** Single Server, Shading, Master-Slave Replication, Peer-to-Peer Replication, Combining Shading and Replication. Consistency, Update Consistency, Read Consistency, Relaxing Consistency, The CAP Theorem, Relaxing Durability, Quorums. Version Stamps, Business and System Transactions, Version Stamps on Multiple Nodes

**UNIT-III:** What Is a Key-Value Store, Key-Value Store Features, Consistency, Transactions, Query Features, Structure of Data, Scaling, Suitable Use Cases, Storing Session Information, User Profiles, Preference, Shopping Cart Data, When Not to Use, Relationships among Data, Multi operation Transactions, Query by Data, Operations by Sets.

**UNIT-IV:** Document Databases, What Is a Document Database?, Features, Consistency, Transactions, Availability, Query Features, Scaling, Suitable Use Cases, Event Logging, Content Management Systems, Blogging Platforms, Web Analytics or Real-Time Analytics, ECommerce Applications, When Not to Use, Complex Transactions Spanning Diferent Operations, Queries against Varying Aggregate Structure

**UNIT-V:** Graph Databases, What Is a Graph Database?, Features, Consistency, Transactions, Availability, Query Features, Scaling, Suitable Use Cases, Connected Data, Routing, Dispatch and Location-Based Services, Recommendation Engines, When Not to Use

**TEXTBOOKS:**

1. Sadalage, P. & Fowler, No SQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence, Pearson Addision Wesley, 2012

**REFERENCE BOOKS:**

1. Dan Sullivan, "NoSQL For Mere Mortals", 1st Edition, Pearson Education India, 2015. (ISBN13: 978-9332557338)
2. Dan McCreary and Ann Kelly, "Making Sense of NoSQL: A guide for Managers and the Rest of us", 1st Edition, Manning Publication/Dreamtech Press, 2013. (ISBN-13: 978-9351192022)
3. Kristina Chodorow, "Mongodb: The Definitive Guide- Powerful and Scalable Data Storage", 2nd Edition, O'Reilly Publications, 2013. (ISBN-13: 978-9351102694)

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### SOFT COMPUTING

**Course Objectives:**

- To introduce soft computing concepts and techniques and foster their abilities in designing appropriate technique for a given scenario.
- To implement soft computing based solutions for real-world problems.
- To give students knowledge of non-traditional technologies and fundamentals of artificial neural networks, fuzzy sets, fuzzy logic, genetic algorithms.
- To provide student a hand-on experience on MATLAB to implement various strategies.

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

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | **Elaborate** fuzzy logic and reasoning to handle uncertainty in engineering problems. | **K6** |
| **CO2** | **Make use of** genetic algorithms to combinatorial optimization problems | **K3** |
| **CO3** | **Distinguish** artificial intelligence techniques, including search heuristics, knowledge representation, planning and reasoning. | **K4** |
| **CO4** | **Formulate and apply** the principles of self-adopting and self organizing neuro fuzzy inference systems. | **K6** |
| **CO5** | **Evaluate and compare** solutions by various soft computing approaches for a given problem | **K5** |

**#based on suggested Revised BTL**

**Mapping of course outcomes with program outcomes**

| CO | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 3 | 1 | | 2 | | | | | | | | |
| CO2 | 1 | 3 | 2 | | | | | | | 1 | | |
| CO3 | 1 | 2 | 3 | | 1 | | | | | | | |
| CO4 | | 2 | 3 | | 1 | | | 1 | | | | |
| CO5 | 1 | | | 1 | | | | | | | 1 | |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

Department of Computer Science & Engineering
University College of Engineering, JNT University Kakinada

**Syllabus:**

**UNIT I: FUZZY SET THEORY:** Introduction to Neuro, Fuzzy and Soft Computing, Fuzzy Sets, Basic function and Terminology, Set-theoretic Operations, Member Function Formulation and Parameterization, Fuzzy Rules and Fuzzy Reasoning, Extension Principle and Fuzzy Relations, Fuzzy If-Then Rules, Fuzzy Reasoning, Fuzzy Inference Systems, Mamdani Fuzzy Models, Sugeno Fuzzy Models, Tsukamoto Fuzzy Models, Input Space Partitioning and Fuzzy Modeling. Case study: Fuzzy Relations (Max-min Composition)

**UNIT II: OPTIMIZATION:** Derivative based Optimization, Descent Methods, and The Method of Steepest Descent, Classical Newton's Method, Step Size Determination, Derivative-free Optimization, Genetic Algorithms, Simulated Annealing, and Random Search, Downhill Simplex Search.
Case study: travelling sales person problem (TSP) using genetic algorithms,Fuzzy Controller (Washing Machine)

**UNIT III: ARTIFICIAL INTELLIGENCE:** Introduction, Knowledge Representation, Reasoning, Issues and Acquisition: Prepositional and Predicate Calculus Rule Based knowledge Representation Symbolic Reasoning Under Uncertainty Basic knowledge Representation Issues Knowledge acquisition, Heuristic Search: Techniques for Heuristic search Heuristic Classification State Space Search: Strategies Implementation of Graph Search based on Recursion Patent-directed Search Production System and Learning.

**UNIT IV: NEURO FUZZY MODELING:** Adaptive Neuro-Fuzzy Inference Systems, Architecture Hybrid Learning Algorithm, Learning Methods that Cross-fertilize ANFIS and RBFN Coactive Neuro Fuzzy Modeling, Framework Neuron Functions for Adaptive Networks Neuro Fuzzy Spectrum.

**UNIT V: APPLICATIONS OF COMPUTATIONAL INTELLIGENCE:** Printed Character Recognition, Inverse Kinematics Problems, Automobile Fuel Efficiency Prediction, Soft Computing for Coloripe Prediction.
Case studies: Simple Genetic Application

**TEXT BOOKS:**

1. "Neuro-Fuzzy and Soft Computing", J.S.R.Jang, C.T.Sun and E.Mizutani, PHI, 2004, Pearson Education 2004
2. Artificial Intelligence by Saroj Koushik, Cengage Learning
3. "Artificial Intelligence and Intelligent Systems", N.P.Padhy, Oxford University Press, 2006

**REFERENCE BOOKS:**

1. Artificial Intelligence, Second Edition, Elaine Rich & Kevin Knight, Tata McGraw Hill Publishing Comp., New Delhi, , 2006
2. “Fuzzy Logic with Engineering Applications”, Timothy J.Ross, McGraw-Hill, 1997

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### SOCIAL NETWORKS & SEMANTICS WEB

**Course Objectives:**

- The learning objective of the course Social Network Analysis is to provide students with essential knowledge of network analysis applicable to real world data, with examples from today's most popular social networks.

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

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | **Demonstrate** social network analysis and measures. | **K2** |
| **CO2** | **Analyze** random graph models and navigate social networks data | **K4** |
| **CO3** | **Apply** the network topology and Visualization tools. | **K3** |
| **CO4** | **Analyze** the experiment with small world models and clustering models. | **K4** |
| **CO5** | **Compare** the application driven virtual communities from social network Structure. | **K5** |

**#based on suggested Revised BTL**

**Mapping of course outcomes with program outcomes**

| CO | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 1 | 3 | 2 | | | | | 1 | | | | |
| CO2 | | 2 | 2 | | 3 | | 1 | | | | | |
| CO3 | | 1 | 1 | | | 2 | | 2 | | | | 3 |
| CO4 | 1 | 1 | 2 | | 3 | 2 | 1 | 1 | | | | |
| CO5 | 1 | | | | 3 | 2 | | | 1 | | | |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**SYLLABUS:**

**UNIT I: Social Network Analysis:** Preliminaries and definitions, ErdosNumber Project, Centrality measures, Balance and Homophily.

**UNIT II: Random graph models:** Random graphs and alternative models, Models of network growth, Navigation in social Networks, Cohesive subgroups, Multidimensional Scaling, Structural equivalence, roles and positions.

**UNIT III: Network topology** and diffusion, Contagion in Networks, Complex contagion, Percolation and information, Navigation in Networks Revisited.

**UNIT IV:** Introduction to the Web Science and Semantic Web, Introduction to Ontologies, Ontology Languages for the Semantic Web – Resource Description Framework (RDF) – Lightweight ontologies: RDF Schema – Web Ontology Language (OWL) – A query language for RDF: SPARQL, Ontology Engineering Semantic web and Web 2.0 Applications of Semantic Web, Infrastructure Social Networks, Web 3.0 - Linked Data RDFa and the Open Graph Protocol schema.org and search enhancement Semantic

**UNIT V: Network structure** -Important vertices and page rank algorithm, towards rational dynamics in networks, basics of game theory, Coloring and consensus, biased voting, network formation games, network structure and equilibrium, behavioral experiments, Spatial and agent-based models.

**TEXT BOOKS:**

1. S. Wasserman and K. Faust. Social Network Analysis: Methods and Applications (Cambridge, Cambridge University Press, 1994)
2. D. Easley and J. Kleinberg, Networks, Crowds and Markets: Reasoning about a highly connected world

**REFERENCES:**

1. Michael C. Daconta, Leo J. Obrst, and Kevin T. Smith, “The Semantic Web: A Guide to the Future of XML, Web Services, and Knowledge Management”, Fourth Edition, Wiley Publishing, 2003.
2. John Davies, Rudi Studer, and Paul Warren John, “Semantic Web Technologies: Trends and Research in Ontology-based Systems”, Wiley and Son's, 2006.
3. John Davies, Dieter Fensel and Frank Van Harmelen, “Towards the Semantic Web: Ontology- Driven Knowledge Management”, John Wiley and Sons, 2003.

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**POOL 4:**

### DIGITAL MARKETING

**Course Objective:**

- The objective of this course is to understand the importance of digital marketing and its applications.

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

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | **Demonstrate** the applications of digital marketing in the globalized market | **K6** |
| **CO2** | **Apply** Channels of Digital Marketing | **K3** |
| **CO3** | **Organize** digital marketing plan | **K4** |
| **CO4** | **Analyze** Search engine marketing | **K5** |
| **CO5** | **Make use of** Online Advertising | **K2** |

**#based on suggested Revised BTL**

**Mapping of course outcomes with program outcomes**

| CO | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 3 | 3 | 2 | | 1 | 2 | | 3 | | 1 | 2 | |
| CO2 | | 3 | 2 | | 2 | 3 | | 3 | | 1 | | |
| CO3 | 2 | 1 | 3 | | | | | | 1 | 1 | 1 | |
| CO4 | 2 | | 2 | | 2 | | | | 1 | | | 3 |
| CO5 | 1 | | | | | | 2 | 3 | | | 3 | |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**SYLLABUS:**

**UNIT - I: Understanding Digital Marketing:** Concept, Components of Digital Marketing, Need and Scope of Digital Marketing, Benefits of Digital Marketing, Digital Marketing Platforms and Strategies, Comparison of Marketing and Digital Marketing, Digital Marketing Trends. Case study: Social Sharing with YouTube.

**UNIT - II: Channels of Digital Marketing:** Digital Marketing, Website Marketing, Search Engine Marketing, Online Advertising, Email Marketing, Blog Marketing, Social Media Marketing, Audio, Video and Interactive Marketing, Online Public Relations, Mobile Marketing, Migrating from Traditional Channels to Digital Channels.
**Marketing in the Digital Era:** Segmentation – Importance of Audience Segmentation, How different segments use Digital Media – Organisational Characteristics, Purchasing Characteristics, Using Digital Media to Reach, Acquisition and Retention of new customers, Digital Media for Customer Loyalty.

**UNIT - III: Digital Marketing Plan:** Need of a Digital Marketing Plan, Elements of a Digital Marketing Plan – Marketing Plan, Executive Summary, Mission, Situational Analysis, Opportunities and Issues, Goals and Objectives, Marketing Strategy, Action Plan, Budget, Writing the Marketing Plan and Implementing the Plan, Case study: Business cards, Blog graphics

**UNIT - IV: Search Engine Marketing and Online Advertising:** Importance of SEM, understanding Web Search – keywords, HTML tags, Inbound Links, Online Advertising vs. Traditional Advertising, Payment Methods of Online Advertising – CPM (Cost-per-Thousand) and CPC (Cost-per-click), Display Ads - choosing a Display Ad Format, Landing Page and its importance.

**UNIT - V: Social Media Marketing:** Understanding Social Media, Social Networking with Facebook, LinkedIn, Blogging as a social medium, Microblogging with Twitter, Social Media for Customer Reach, Acquisition and Retention. Measurement of Digital Media: Analyzing Digital Media Performance, Analyzing Website Performance, Analyzing Advertising Performance.

**TEXT BOOKS:**

1. Michael Miller, B2B Digital Marketing, 1e, Pearson, 2014.
2. Vandana Ahuja, Digital marketing, Oxford University Press 2015

**REFERENCES:**

1. Michael R Solomon, Tracy Tuten, Social Media Marketing, Pearson, 1e, 2015.
2. Judy Strauss & Raymond Frost, E-Marketing, Pearson, 2016
3. Richard Gay, Alan Charles worth and Rita Esen, Online marketing – A customer led approach Oxford University Press 2007.
4. Chuck Hemann& Ken Burbary, Digital Marketing Analytics, Pearson, 2019

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### WIRELESS NETWORKS SECURITY

**Course Objective:**

- The objective of this course is to understand the importance of Wireless networks security and its application

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

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | **Explain** the Threats in networks and provide Authentication to real time problems. | **K2** |
| **CO2** | **Identify** and investigate in-depth both early and contemporary threats to wireless networks security | **K3** |
| **CO3** | **Ability to analyze** and determine for any organization the database security requirements and appropriate solutions | **K4** |
| **CO4** | **Explain** IP Security Issues and solve real time problems. | **K2** |
| **CO5** | **List** the Basic specifications in Bluetooth Security. | **K1** |

**#based on suggested Revised BTL**

**Mapping of course outcomes with program outcomes**

| CO | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 1 | 2 | 3 | 1 | | 1 | | | | | 1 | |
| CO2 | 1 | 2 | 3 | | | | | | | 1 | | |
| CO3 | | 2 | 2 | | | 1 | | 1 | | | | |
| CO4 | 1 | 2 | 3 | 1 | 2 | 2 | | | | | 1 | |
| CO5 | 1 | 2 | 3 | | | 1 | | | | 1 | | |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**SYLLABUS:**

**UNIT-I: Introduction to Wireless**: History of Wireless Technologies, History of Wireless Security, State of the Wireless Security Industry, 2001

**Wireless Threats:** Uncontrolled Terrain, Communications Jamming, DoS Jamming, Injections and Modifications of Data, Man-in-the-Middle (MITM) Attack, Rogue Client, Rogue Network Access Points, Attacker Equipment, Covert Wireless Channels, Roaming Issues, Cryptographic Threats

**UNIT-II: Introduction to Wireless Security Protocols and Cryptography:** Recovery the FUD, OSI Model, OSI Simplified, Internet Model, Wireless LAN Security Protocols, Cryptography, SSL/TLS, Secure Shell Protocols, Terminal Access and File Transfer, Port Forwarding a Word of Caution,Man-in-the-Middle of SSL/TLS and SSH, WTLS, WEP,802.1x, IP Security

**Security Considerations to Wireless Devices:** Wireless Device Security Issues, Physical Security, Information Leakage, Device Security Features, Application Security, Detailed Device Analysis, Laptops, Personal Digital Assistants (PDAS), Wireless Infrastructure

**UNIT-III: Wireless Technologies and Applications:** Introduction to Cellular Networks- FDMA, TDMA, CDMA, Spread Spectrum Primer, Analogy, TDMA Vs CDMA, PDC, Security Threats, GSM Security, GSM Algorithm Analysis

**Introduction to Wireless Data Networks:** Cellular Digital Packet Data (CDPD), CDPD Architecture, CDPD Security, Mobitex- Mobitex Architecture, Mobitex Security Architecture, General Packet Radio Service (GPRS)- GPRS Architecture, Security Issues, Introduction to the Wireless Application Protocol (WAP)- WAP Device, Gateway, Security Model

**UNIT-IV: Wireless Standards and Technologies:** Current and Future Technologies- Infrared, Radio, Spread Spectrum, OFDM, Current and Future Standards- IEEE 802, 802.11, The ABC's of 802.11, 802.11b, 802.11a, 802.11g, 802.11j, 802.11h and 5GPP, 802.11e, 802.11i, 802.11f, IEEE 802.15, IEEE 802.16, IEEE 802.1x, ETSI, HomeRF, Ultrawideband Radio (UWB)

**Wireless Deployment Strategies:** Implementing Wireless LAN's- Security Considerations Common Wireless Network Applications, Enterprise Campus Designs, Wireless IST Design, Retail and Manufacturing Design, Small Office/Home Office Design (SOHO)

**UNIT-V: Bluetooth Security:** Basic Specifications, Pico-nets, Bluetooth Security Architecture, Scatter-nets, Security at the Baseband Layer and Link Layer, Frequency Hopping, Security Manager, Authentication, Encryption, Threats to Bluetooth Security

**TEXT BOOKS:**

1. Wireless Security, Merritt Maxim and David Pollino, Osborne/McGraw Hill, New Delhi, 2005
2. Wireless Security Models: Threats and Solutions, Nichols and Lekka, Tata McGraw Hill, New Delhi 2006

**REFERENCE BOOKS:**

1. Behrouz A.Forouzan, ―Cryptography & Network Security‖, Tata McGraw Hill, India, New Delhi, 2009
2. William Stallings, ―Cryptography and Network Security, Prentice Hall, New Delhi, 2006
3. Bruce Schneier, "Applied Cryptography", John Wiley & Sons, New York, 2004

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### HIGH PERFORMANCE COMPUTING

**Course Objectives:** The objective of the subject is to

- Introduce the basic concepts related to HPC architecture and parallel computing.
- To discuss various computational techniques for studying soft matter systems.
- To apply these concepts to examine complex bimolecular/materials systems that generally require large-scale HPC platform with hybrid CPU-GPU architectures.

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

| CO | Description | Knowledge Level (K)# |
|---|---|---|
| CO1 | **Design**, formulate, solve and implement high performance versions of standard single threaded algorithms. | K6 |
| CO2 | **Demonstrate** the architectural features in the GPU and MIC hardware accelerators. | K2 |
| CO3 | **Design** programs to extract maximum performance in a multi-core, shared memory execution environment processor. | K6 |
| CO4 | **Analyze** Symmetric and Distributed architectures. | K4 |
| CO5 | **Develop** and deploy large scale parallel programs on tightly coupled parallel systems using the message passing paradigm. | K6 |

**#based on suggested Revised BTL**

**Mapping of course outcomes with program outcomes**

| CO | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 1 | 3 | 2 | | 1 | 1 | | | | | | |
| CO2 | | 2 | 3 | 1 | 1 | 2 | | | | | | |
| CO3 | | 1 | 3 | 2 | | 1 | 1 | | | | | |
| CO4 | 2 | 3 | 2 | | 1 | 1 | | | | | | |
| CO5 | | 1 | 3 | 2 | | 2 | 1 | | | | | |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**SYLLABUS:**

**UNIT I: Graphics Processing Units:** Introduction to Heterogeneous Parallel Computing, GPU architecture, Thread hierarchy, GPU Memory Hierarchy.

**UNIT II: GPU Programming:** Vector Addition, Matrix Multiplication algorithms. 1D, 2D, and 3D Stencil Operations, Image Processing algorithms – Image Blur, Gray scaling. Histogramming, Convolution, Scan, Reduction techniques.

**UNIT III: Many Integrated Cores:** Introduction to Many Integrated Cores. MIC, Xeon Phi architecture, Thread hierarchy, Memory Hierarchy, Memory Bandwidth and performance considerations.

**UNIT IV: Shared Memory Parallel Programming:** Symmetric and Distributed architectures, OpenMP Introduction, Thread creation, Parallel regions. Work sharing, Synchronization.

**UNIT V: Message Passing Interface:** MPI Introduction, Collective communication, Data grouping for communication.

**TEXT BOOKS:**

1. Programming Massively Parallel Processors A Hands-on Approach, 3e, Wen-Mei W Hwu, David B Kirk and Morgan Kaufmann-2019
2. Intel Xeon Phi Coprocessor Architecture and Tools, Rezaur Rahman, Apress Open, 1st edition-2013
3. Using OpenMP, Barbara Chapman, Gabriele Jost, Rudd Vander Pas, MIT Press, 2008

**REFERENCE BOOKS:**

1. "A Parallel Algorithm Synthesis Procedure for High-Performance Computer Architectures" by Dunn Ian N, 2003

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### MULTI AGENT SYSTEMS

**Course Outcomes:**
At the end of the module, the student will be able to demonstrate:

- Understand the notion of an agent, how agents are distinct from other software paradigms (eg objects).
- Understand the key issues associated with constructing agents capable of intelligent autonomous action, and the main approaches taken to developing such agents;
- Understand the key issues in designing societies of agents that can effectively cooperate in order to solve problems, including an understanding of the key types of multi-agent interactions possible in such systems
- Understand the main application areas of agent-based solutions, and be able to develop a meaningful agent-based system using a contemporary agent development platform.

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

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | **how** agents are distinct from other software paradigms (e.g. objects) and understand the characteristics of applications | **K1** |
| **CO2** | **Identify** different types of multi-agent interactions possible in such systems | **K3** |
| **CO3** | **Develop** a meaningful agent-based system using a contemporary agent development platform. | **K6** |
| **CO4** | Compare and contrast between benevolent agents | **K5** |
| **CO5** | Discuss about FIPA framework. | **K6** |

***#Based on suggested Revised BTL***

**Mapping of course outcomes with program outcomes**

| CO | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 1 | 3 | 2 | | 1 | 1 | | | | | | |
| CO2 | 1 | 3 | 3 | | 2 | | | | | | | |
| CO3 | 1 | 3 | 3 | 2 | 1 | | | | | | 1 | |
| CO4 | | 2 | 1 | | | 3 | | 2 | | 1 | | |
| CO5 | 2 | 1 | 3 | | | | 1 | 1 | | | | |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**SYLLABUS:**

**UNIT-1: Introduction**- Agent, agents and objects, agents and expert systems, agents and distributed systems, typical application areas for agent systems.

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

**UNIT-III: Multi-Agent Systems:** Classifying multi-agent interactions - cooperative versus non-cooperative, zero-sum and other interactions, what is cooperation, cooperation occurs - the Prisoner's dilemma and Axelrod's experiments.

**UNIT-IV: Interactions between self-interested agents:** auctions & voting systems: negotiation; Interactions between benevolent agents: cooperative distributed problem solving (CDPS), partial global planning; coherence and coordination.

**UNIT-V: Interaction languages and protocols:** speech acts, KQML/KIF, the FIPA framework. Advanced topics: One issue selected from the contemporary research literature, perhaps by guest lecturer.

**TEXT BOOK:**

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

# GENERAL MINOR TRACKS

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### GENERAL MINOR TRACKS

**Note:**

1. **The student can opt any 4 subjects from each pool.**
2. **Concerned BoS can add or delete the subjects as per the decision of the board.**
3. **Pre requisites to be defined by the board for each course.**
4. **Compulsory MOOC/NPTEL Courses for 04 credits (02 courses@ 2 credits each)**

**Department of Mechanical Engineering**

5. **Students need to select the courses which were not offered in their regular programme.**

| S. No | Subject | L-T-P | Credit |
|---|---|---|---|
| 1 | Operating systems | 3-1-0 | 4 |
| 2 | Data Structures Using C | 3-1-0 | 4 |
| 3 | Computer organization and Architecture | 3-1-0 | 4 |
| 4 | Software Engineering | 3-1-0 | 4 |
| 5 | Design and analysis of algorithms | 3-1-0 | 4 |
| 6 | Computer Networks | 3-1-0 | 4 |
| 7 | Database Management Systems | 3-1-0 | 4 |
| 8 | Object oriented programming | 3-1-0 | 4 |
| 9 | Data warehousing and Data mining | 3-1-0 | 4 |
| 10 | Data Science | 3-1-0 | 4 |
| 11 | Artificial Intelligence | 3-1-0 | 4 |

Department of Computer Science & Engineering
University College of Engineering, JNT University Kakinada

## OPERATING SYSTEMS

**COURSE OBJECTIVES:**

- Study the basic concepts and functions of operating systems.
- Understand the structure and functions of OS.
- Learn about Processes, Threads and Scheduling algorithms.
- Understand the principles of concurrency and Deadlocks.
- Learn various memory management schemes.
- Study I/O management and File systems.
- Learn the basics of Linux system and perform administrative tasks on Linux Servers.

**COURSE OUTCOMES:**

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| CO1 | **Describe** Computer Operating System Functions, Structures and System Calls. | K1 |
| CO2 | **Demonstrate** various Process Management Concepts and CPU Scheduling Algorithms and Process Synchronization Techniques. | K3 |
| CO3 | **Illustrate** Memory Management Techniques and Page Replacement Algorithms. | K2 |
| CO4 | **Apply** Deadlock Prevention and Avoidance Techniques | K3 |
| CO5 | **Demonstrate** File System Concepts and Mass Storage Structures | K3 |

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | | 3 | 2 | | 1 | | | | | | | |
| CO2 | 2 | 3 | 2 | 1 | | | | | | | | |
| CO3 | | 3 | 2 | | 1 | | | | | | | |
| CO4 | 3 | 3 | 2 | | 2 | | | | | | | |
| CO5 | 2 | 3 | 2 | | 2 | | | | | | | |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**Department of Computer Science & Engineering**
**University College of Engineering, JNT University Kakinada**

**SYLLABUS:**

**UNIT-I: Introduction to Operating System Concept:** Types of operating systems, operating systems concepts, Evaluation of operating systems, operating systems services, structure of OS, Introduction to System call, System call types.

**UNIT-II: Process Management** – Process concept, The process, Process State Diagram , Process control block, Process Scheduling- Scheduling Queues, Schedulers, Operations on Processes, Interprocess Communication, Threading Issues, Scheduling-Basic Concepts, Scheduling Criteria, Scheduling Algorithms.

**UNIT-III: Memory Management:** Swapping, Contiguous Memory Allocation, Paging, structure of the Page Table, Segmentation **Virtual Memory Management:** Virtual Memory, Demand Paging, Page-Replacement Algorithms.

**UNIT-IV: Concurrency:** Process Synchronization, The Critical- Section Problem, Synchronization Hardware, Semaphores, Classic Problems of Synchronization, Monitors, Synchronization examples **Principles of deadlock**: System Model, Deadlock Characterization, Deadlock Prevention, Detection and Avoidance, Recovery form Deadlock

**UNIT-V: File System Interface:** Concept of a file, Access Methods, Directory structure, File system mounting, file sharing, protection. **File System Implementation:** File system structure, allocation methods, Disk scheduling, **Case studies: Android, UNIX, Windows**

**TEXT BOOK:**

1. Operating System Concepts, Abraham Silberschatz, Peter Baer Galvin and Greg Gagne 9th Edition, John Wiley and Sons Inc., 2012.
2. Operating Systems – Internals and Design Principles, William Stallings, 7th Edition, Prentice Hall, 2011.

**REFERENCES:**

1. Modern Operating Systems, Andrew S. Tanenbaum, Second Edition, Addison Wesley, 2001.
2. Operating Systems: A Design-Oriented Approach, Charles Crowley, Tata McGraw Hill Education, 1996.
3. Operating Systems: A Concept-Based Approach, D M Dhamdhere, Second Edition, Tata McGraw-Hill Education, 2007.
4. Operating Systems-S Halder, Alex A Aravind Pearson Education Second Edition 2016

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### DATA STRUCTURES USING C

**Course Objectives:**

- Solve problems using data structures such as linear lists, stacks, queues, hash tables
- Be familiar with advanced data structures such as balanced search trees, AVL Trees, and B Trees.

**Course Outcomes:** By the end of the course student will be able to understand

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Select appropriate data structures as applied to specified problem definition | **K5** |
| **CO2** | Summarize and understand the practical applications of several advanced techniques like Hashing and Analyzing and Implement appropriate sorting/searching technique for given problems | **K2** |
| **CO3** | Demonstrate the operations such as Insertion, Deletion and Search on Data structures like Binary Search Tree and solve the problems | **K3** |
| **CO4** | Demonstrate the operations such as Insertion, Deletion and Search on Advanced Data structures like Heaps, AVL trees and B Trees. | **K3** |
| **CO5** | Comparisons of trees like Red Black trees and B-Trees etc. and priority queue operations. | **K4** |

**#based on suggested Revised BTL**

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 |
|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 3 | 2 | 2 | | | | | | | |
| **CO2** | 2 | | 3 | | | | | 3 | | |
| **CO3** | 1 | 2 | 3 | | 3 | | | | | |
| **CO4** | | | | 2 | 3 | 1 | | 3 | | |
| **CO5** | 2 | | | | | | | 2 | | |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**SYLLABUS:**

**UNIT- I: Introduction to Data Structures:** Abstract Data Types (ADTs), The List ADT: Simple Array Implementation of Lists, Simple Linked Lists, Doubly Linked Lists, Circularly Linked Lists. The Stack ADT: The Stack Model, Implementation of Stacks, Applications of Stack. The Queue ADT: Queue Model, Array Implementation of Queues, Application of Queues. Stacks and Queue implementation using linked list.

**UNIT-II: Searching:** List Searches, Linear and Binary Search Methods.
**Sorting:** Selection Sort, Insertion Sort, Quick Sort, Merge Sort, Heap sort.
**Hashing:** Hash Function, Separate Chaining, Collision Resolution-Separate Chaining.

**UNIT- III: Trees :** Binary Trees- Implementation, Expression Trees. Binary Search Trees- find, findMin and findMax, insert, delete operations.

**UNIT- IV: Trees:** AVL Trees- Single and Double Rotation, Operations. B-Tree: searching, insertion, deletion

**UNIT –V: Trees:** Introduction to Red-Black, splay trees and Comparison of Search Trees **Priority Queues:** Priority Queue Models, Simple Implementations.

**TEXT BOOKS:**

1. Data Structures and Algorithm Analysis, 4th Edition, Mark Allen Weiss, Pearson.
2. Data Structures: A Pseudo Code Approach with C, 2nd Edition, Richard F.Gilberg, & Behrouz A. Forouzon, Cengage.

**REFERENCES BOOKS**:

1. Data Structures, Algorithms and Applications in java, 2/e, Sartaj Sahni,University Press.
2. Data Structures using C, 2/e, Reema Thareja

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### COMPUTER ORGANIZATION AND ARCHITECTURE

**Course Objectives:**

- The purpose of the course is to introduce principles of computer organization and the basic architectural concepts.
- It provides an in depth understanding of basic organization, design, programming of a simple digital computer, computer arithmetic, instruction set design, micro programmed control unit, pipelining and vector processing, memory organization and I/O systems

**Course Outcomes:** By the end of the course student will be able to

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| CO1 | **Demonstrate** an understanding of the different number systems, codes and Relate Postulates of Boolean algebra and minimize combinational functions | K1, K2 |
| CO2 | **Evaluate and learn** different combinational circuits, sequential circuits and able to design them | K5 |
| CO3 | **Organize,Determine and learns** basic structure of components register through language, micro operations and able to write micro programs | K3, K5 |
| CO4 | **Determine and** able to write data transfer and manipulators program and students able to learn micro programme control and central processing unit | K5 |
| CO5 | Able to **learns** the internal organization of computers and able to evaluate performance of them. | K1, K5 |

**#based on suggested Revised BTL**

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 2 | | 3 | 1 | | | | | | | | |
| CO2 | 2 | 3 | 3 | 1 | | | | | | | | |
| CO3 | 2 | 2 | 1 | 3 | | 1 | | 1 | | | | |
| CO4 | | 3 | 2 | | | | | 1 | | | | |
| CO5 | 1 | 2 | 3 | | | 2 | | | | | | |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**SYLLABUS:**

**UNIT I: Number System and Data Representation**: Introduction, Numbering Systems, Decimal to Binary Conversion, Binary Coded Decimal Numbers, Error Detecting Codes, Error Correcting Codes, Hamming Code for Error Correction.

**Boolean algebra and Logic gates :** Karnaugh map representation and minimization of Boolean functions using K-maps up to 4-variable;Don't care conditions, Digital Logic gates, Two-level realizations using gates -- AND-OR, OR-AND, NAND-NAND and NOR-NOR

**UNIT II: Combinational logic circuits-I:** Design of Half adder, full adder, half subtractor, full subtractor, Design of decoder, De-multiplexer, encoder, multiplexer. **Sequential circuits I:** Classification of sequential circuits (synchronous and asynchronous): basic flip-flops, truth tables and excitation tables (NAND RS latch, NOR RS latch, RS flip-flop. JK flip-flop, T flip-flop, D flip-flop with reset and clear terminals).

**Unit-III: Basic Structure Of Computers:** Computer Types, Functional unit, Basic Operational concepts, Bus structures, Software, Performance, multiprocessors and multi computers.
**Register Transfer Language And Micro-operations:** Register Transfer language. Register Transfer Bus and memory transfers, Arithmetic Micro-operations, Logic micro operations, shift micro operations. Instruction codes. Computer Registers, Computer instructions, Instruction cycle.

**UNIT IV: Micro Programmed Control:** Control memory, Address sequencing, micro program example, design of control unit.
**Central Processing Unit:** General Register Organization, Instruction Formats, Addressing modes, Data Transfer and Manipulation, Program Control.

**UNIT V: Memory Organization:** Memory Hierarchy, Main Memory, Auxiliary memory, Associate Memory, Cache Memory, Virtual memories Introduction to Shift registers and RAID **Input –Output Organization** Input-Output Interface, Asynchronous data transfer, Modes of Transfer, Priority Interrupts, DMA, Input Output Processor, Serial Communication.

**TEXT BOOKS:**

1. Digital Logic and Computer Design, Moriss Mano, 11th Edition, Pearson Education.
2. Computer Organization, 5$^{th}$ed.,Hamacher, Vranesicand Zaky,TMH,2002
3. Computer System Architecture, 3/e, MorisMano,Pearson/PHI.

**REFERENCE BOOKS:**

1. Computer System Organization &Architecture, John D.Carpinelli, Pearson, 2008
2. Computer System Organization, NareshJotwani, TMH, 2009
3. Computer Organization &Architecture: Designing for Performance, 7$^{th}$ed., William Stallings, PHI, 2006
4. Structured Computer Organization – Andrew S. Tanenbaum, 4th Edition, PHI/Pearson.

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### SOFTWARE ENGINEERING

**Course Objectives:**

- To understand the software life cycle models
- To understand the software requirements and SRS document.
- To understand the importance of modelling and modelling languages.
- To design and develop correct and robust software products.
- To understand the quality control and how to ensure good quality software.
- To understand the planning and estimation of software projects.
- To understand the implementation issues, validation and verification procedures.

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

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Define Software Engineering methodologies and software process | **K1** |
| **CO2** | Explain requirements engineering activities | **K2** |
| **CO3** | Classify various software design techniques. | **K2** |
| **CO4** | Apply coding principles and testing techniques. | **K3** |
| **CO5** | Explain management and estimation techniques in software development process. | **K2** |

**#based on suggested Revised BTL**

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | | 3 | 3 | | | 2 | | 1 | | | | |
| **CO2** | 3 | 2 | | | 2 | 1 | | | | | 2 | |
| **CO3** | | 3 | 2 | | | 1 | | | | | | |
| **CO4** | 2 | | | | 3 | 2 | | 3 | | 1 | | |
| **CO5** | | 3 | 2 | | | 2 | | | | | 1 | |

**(Please fill the above with Levels of Correlation, viz., L-1, M-2, H-3)**

**SYLLABUS:**

**UNIT-I: Software and Software Engineering:** The Nature of Software, Software Engineering, Software Process, Software Engineering Practice, Software Myths.

**UNIT-II: Process Models:** A Generic Process Model, Process Assessment and Improvement, Prescriptive Process Models, Specialized Process Models, The Unified Process, Personal and Team Process Models, Process Terminology.

**UNIT – III: Requirements Analysis And Specification:** Requirements Gathering and Analysis, Software Requirement Specification (SRS), Formal System Specification. Software Design: Overview of the Design Process, How to

Characterise of a Design?, Cohesion and Coupling, Layered Arrangement of Modules, Approaches to Software Design

**UNIT – IV: Function-Oriented Software Design:** Overview of SA/SD Methodology, Structured Analysis, Developing the DFD Model of a System, Structured Design, Detailed Design, Design Review, over view of Object Oriented design. User Interface Design: Characteristics of Good User Interface, Basic Concepts, Types of User Interfaces, Fundamentals of Component-based GUI Development, A User Interface Design Methodology.

**UNIT – V: Coding And Testing:** Coding, Code Review, Software Documentation, Testing, Unit Testing, Black-Box Testing, White-Box Testing, Debugging, Program Analysis Tool, Integration Testing, Testing Object-Oriented Programs, System Testing, Some General Issues Associated with Testing

**TEXT BOOKS:**

1. Software Engineering A practitioner’s Approach, Roger S. Pressman, Seventh Edition McGrawHill International Edition.
2. Fundamentals of Software Engineering, Rajib Mall, Third Edition, PHI.
3. Software Engineering, Ian Sommerville, Ninth edition, Pearson education

**REFERENCE BOOKS:**

1. Software Engineering : A Primer, Waman S Jawadekar, Tata McGraw-Hill, 2008
2. Software Engineering, A Precise Approach, PankajJalote, Wiley India,2010.
3. Software Engineering, Principles and Practices, Deepak Jain, Oxford University Press.
4. Software Engineering1: Abstraction and modeling, Diner Bjorner, Springer International edition, 2006.

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### DESIGN AND ANALYSIS OF ALGORITHMS

**Course Objectives:**

Upon completion of this course, students will be able to do the following:

- Ability to understand, analyze and denote time complexities of algorithms
- To introduce the different algorithmic approaches for problem solving through numerous example problems
- Describe the dynamic-programming paradigm and explain when an algorithmic design situation calls for it. Recite algorithms that employ this paradigm. Synthesize dynamic-programming algorithms, and analyze them.
- To provide some theoretical grounding in terms of finding the lower bounds of algorithms and the NP-completeness

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

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Analyze the performance of a given algorithm, denote its time complexity using the asymptotic notation for recursive and non-recursive algorithms | **K4** |
| **CO2** | List and describe various algorithmic approaches and Solve problems using divide and conquer &greedy Method | **K2** |
| **CO3** | Synthesize efficient algorithms dynamic programming approaches to solve in common engineering design situations. | **K3** |
| **CO4** | Organize important algorithmic design paradigms and methods of analysis: backtracking, branch and bound algorithmic approaches | **K4** |
| **CO5** | Demonstrate NP- Completeness theory ,lower bound theory and String Matching | **K6** |

**#based on suggested Revised BTL**

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | | 3 | 3 | | | 2 | | 1 | | | | |
| **CO2** | 3 | 2 | | | 2 | 1 | | | | | 2 | |
| **CO3** | | 3 | 2 | | | 1 | | | | | | |
| **CO4** | 2 | | | | 3 | 2 | | 3 | | 1 | | |
| **CO5** | | 3 | 2 | | | 2 | | | | | 1 | |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**SYLLABUS:**

**UNIT I: Introduction:** Algorithm Definition, Algorithm Specification, performance Analysis, Performance measurement, asymptotic notation, Randomized Algorithms.

**UNIT II:** Divide and Conquer: General Method, Defective chessboard, Binary Search, finding the maximum and minimum, Merge sort, Quick sort.

**The Greedy Method:** The general Method, knapsack problem, minimum-cost spanning Trees, Optimal Merge Patterns, Single Source Shortest Paths.

**UNIT III: Dynamic Programming:** The general method, multistage graphs, All pairs-shortest paths, optimal Binary search trees, 0/1 knapsack, The traveling salesperson problem.

**UNIT IV: Backtracking:** The General Method, The 8-Queens problem, sum of subsets, Graph coloring, Hamiltonian cycles, knapsack problem.

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

**TEXT BOOKS:**

1. Ellis Horowitz, SartajSahni, Sanguthevar Rajasekaran, "Fundamentals of Computer Algorithms", $2^{nd}$Edition, Universities Press.
2. Introduction to Algorithms Thomas H. Cormen, PHI Learning
3. Harsh Bhasin, "Algorithms Design & Analysis", Oxford University Press.

**REFERENCE BOOKS:**

1. Horowitz E. Sahani S: "Fundamentals of Computer Algorithms", $2^{nd}$Edition, Galgotia Publications,2008.
2. S. Sridhar, "Design and Analysis of Algorithms", Oxford University Press.

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### COMPUTER NETWORKS

**COURSE OBJECTIVES:**

- To provide insight about networks, topologies, and the key concepts.
- To gain comprehensive knowledge about the layered communication architectures (OSI and TCP/IP) and its functionalities.
- To understand the principles, key protocols, design issues, and significance of each layers in ISO and TCP/IP.
- To know the basic concepts of network services and various network applications.

**COURSE OUTCOMES (COS)**: At the end of the course, student will be able to

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Explain about the Layers in OSI model,TCP/IP and PC | **K5** |
| **CO2** | Discuss Data Link Layer, Media Access Control Protocols and ethernet | **K6** |
| **CO3** | Identify the Network Layer design issues and Routing Algorithms | **K4** |
| **CO4** | How to Explain the Transport Layer Protocols in networking. | **K2** |
| **CO5** | Explain Standard Client Server Protocols In Application Layer | **K5** |

***#Based on suggested Revised BTL***

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 2 | 2 | 3 | | 2 | | | 1 | | | | |
| **CO2** | 3 | 2 | 2 | 1 | 1 | 2 | | | | | | |
| **CO3** | 2 | 2 | 3 | | 3 | | | 1 | | | | |
| **CO4** | 1 | 2 | 1 | | | | | 1 | | | | |
| **CO5** | 3 | 2 | 2 | 1 | 2 | 2 | | | | | | |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**Syllabus:**

**UNIT - I: Networks:** Network Criteria, Physical Structures, Network Types.

**Layers in OSI model**: Physical Layer, Data Link Layer, Network layer, transport layer, session layer, presentation layer, application layer.

**TCP/IP protocol suite**: Physical and data link layers, Network layer, Transport layer, Application layer, History of Internet.

**Physical Layer:** Transmission media: Introduction, Guided media: Twisted pair cable, Coaxial cable, Fiber Optic cable, Unguided Media: Radio waves, Microwaves, Infrared

**UNIT II: Data Link Layer:** Design issues, **Framing:** fixed size framing, variable size framing, flow control, error control, CRC, services provided to Network Layer, Elementary Data Link Layer protocols: simplex protocol, Simplex stop and wait, Simplex protocol for Noisy Channel.

**Media Access Control Protocols:** Random Access: ALOHA, Carrier sense multiple access (CSMA), CSMA with Collision Detection, CSMA with Collision Avoidance, Controlled Access: Reservation, Polling, Token Passing.
**Standard Ethernet:** MAC Sub Layer, physical layer, Fast Ethernet: MAC Sub Layer, physical layer.

**UNIT-III: Network layer: Network Layer design issues:** store-and forward packet switching, services provided transport layers, implementation connection less services, implementation connection oriented services, comparison of virtual –circuit and datagram subnets.
**Routing Algorithms**-shortest path routing, flooding, distance vector routing, link state routing, Hierarchical routing.
**Internetworking and IP protocols:** How networks differ, How net works can be connected, internetworking, tunneling, IPV4 Protocol, Packet format IP addresses, Subnets, CIDR, classful and Special addressing, Network Address Translation (NAT),IPV6 Address structure, address space, IPV6 Advantages, packet format, extension Headers, Transition from IPV4 to IPV6.

**Congestion control algorithms:** Approaches to congestion control, Traffic aware routing, Admission control, Traffic throttling, choke Packets, Load shedding, random early detection.

**UNIT –IV: Transport Layer Protocols:** Introduction, Services, Port numbers,
User Datagram Protocol: User datagram, UDP services, UDP Applications. **Transmission control Protocol**: TCP services, TCP features, Segment, A TCP connection, State transition diagram, Windows in TCP, Flow control and error control, TCP Congestion control, TCP Timers.

**Unit-V: Application Layer: Introduction:** Providing Services, Application layer paradigms**,** Client Server paradigms
**Standard Client Server Protocols**: WWW and HTTP, FTP, Electronic mail, Telnet, Secure Shell, DNS.

**TEXT BOOKS :**

1. Data Communications and Networks – Behrouz A. Forouzan, Fifth Edition TMH.
2. Computer Networks — Andrew S Tanenbaum, Fifth Edition. Pearson Education/PHI

**REFERENCES:**

1. Data Communications and Networks- Achut S Godbole, Atul Kahate
2. Computer Networks, Mayank Dave, CENGAGE

Department of Computer Science & Engineering
University College of Engineering, JNT University Kakinada

## DATA BASE MANAGEMENT SYSTEMS

**Pre-requisite:** Basic Knowledge on Data Structures

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

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| CO1 | **Illustrate** the concept of databases, database management systems, database languages, database structures and their work | K2 |
| CO2 | **Apply** ER modeling and Relational modeling for designing simple databases. | K3 |
| CO3 | **Summarize** the concepts related to relational model and SQL and **Write** database queries using relational algebra and structured query language. | K2 |
| CO4 | **Design and develop** databases from the real world by applying the concepts of Normalization. | K6 |
| CO5 | **Outline** the issues associated with Transaction Management and Recovery, Tree Structured Indexing | K2 |

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 2 | 1 | 3 | | | 1 | | | | 2 | | |
| CO2 | | 3 | 3 | | | 2 | | 2 | | | | |
| CO3 | | 3 | 3 | | | 2 | | 2 | | | | |
| CO4 | 3 | 3 | | | 2 | | 2 | | | | | |
| CO5 | 3 | 3 | | | 2 | 3 | 2 | | | 1 | | |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**SYLLABUS:**

**UNIT – 1: Overview of Database System**: Database System Applications, Purpose of Database Systems, View of Data, Database Languages, Relational Databases, Database Design, Data Storage and Querying, Transaction Management, Database Architecture, Data Mining and Informational Retrieval, Specialty Databases, Database Users and Administrators, History of Database Systems**. [Text Book -2]**

**UNIT – 2: Introduction to Database Design:** Database Design and ER Diagrams, Entities, Attributes and Entity Sets, Relationships and Relationship Sets, Additional Features of the ER Model, Conceptual Design with the ER Model, Extended ER features **[Text Book -1]**

**UNIT – 3: Relational Model:** Introduction to the Relational Model, Integrity Constraints over Relations, Enforcing Integrity Constraints, Querying Relational Data, Logical Database Design: ER to Relational, Introduction to Views, Destroying/Altering Tables and Views **[Text Book -1]]**

**UNIT – 4: SQL: Queries, Constraints, Triggers:** The Form of a Basic SQL Query, UNION, INTERSECT and EXCEPT, Nested Queries, Aggregate Operators, Null Values, Complex Integrity Constraints in SQL, Triggers, Exceptions, Procedures, Functions **[Text Book -**

**UNIT – 5: Normal Forms:** Introduction to Schema Refinement, Functional Dependencies, Reasoning about FDs, Normal Forms, Properties of Decompositions, Normalization. **[Text Book -1]**

**TEXT BOOKS**:

1. Data base Management Systems, 3/e, Raghurama Krishnan, Johannes Gehrke, Mc Graw-Hill
2. Data base System Concepts, 6/e, Abraham Silberschatz, Henry F. Korth, S. Sudarshan, Mc Graw-Hill
3. Database Systems, 9/e, Carlos Coronel, Steven Morris, Peter Rob, Cengage

**REFERENCE BOOKS**:

1. Database Systems, 6/e Ramez Elmasri, Shamkant B. Navathe, Pearson
2. Introduction to Database Systems, 8/e, C J Date, Pearson

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### OBJECT ORIENTED PROGRAMMING

**Course Objectives:** This course is designed to provide a comprehensive study of the C programming language. It stresses the strengths of C, which provide students with the means of writing efficient, maintainable and portable code. The nature of C language is emphasized in the wide variety of examples and applications.

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

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Demonstrate basics of object oriented mode, differences between conventional and oops programming, the top-down and bottom-up approach I/O streams in C++ | **K2** |
| **CO2** | Explain Write, compile and debug programs and Use different data types, classes, objects and member functions in C++ language. | **K5** |
| **CO3** | Make use of Basic concept in C++ programming, Operators, control structures, functions, overloading, and recursion. | **K3** |
| **CO4** | Build dynamic memory management techniques using pointers, constructors, destructors, virtual functions. | **K3** |
| **CO5** | Classify inheritance with the understanding of early and late binding, usage of exception handling, generic programming. | **K2** |
| **CO6** | Apply advanced features of C++ specifically templates, operator overloading, standard template libraries (STL). | **K3** |

#***Based on suggested Revised BTL***

**Mapping of course outcomes with program outcomes**

| CO | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 3 | | | | | 2 | | | | | | |
| CO2 | | 2 | | | | | | 1 | | | | |
| CO3 | | 3 | 2 | | | | | | 1 | | | |
| CO4 | | | 3 | | 2 | | | | | | | |
| CO5 | | | | | 3 | | | | | | | |
| CO6 | | 1 | | | 2 | | | 3 | | | | |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**Department of Computer Science & Engineering**
**University College of Engineering, JNT University Kakinada**

**SYLLABUS:**

**UNIT – 1: Introduction to C++ :** Difference between C and C++- Evolution of C++- The Object Oriented Technology- Disadvantage of Conventional Programming- Key Concepts of Object Oriented Programming- Advantage of OOP- Object Oriented Language.

**UNIT – 2 Classes and Objects &Constructors and Destructor :** Classes in C++- Declaring Objects- Access Specifiers and their Scope- Defining Member Function- Overloading Member Function- Nested class, Constructors and Destructors, Introduction- Constructors and Destructor- Characteristics of Constructor and Destructor-Application with Constructor- Constructor with Arguments (parameterized Constructor-Destructors)

**UNIT – 3 Operator Overloading and Type Conversion & Inheritance :**
The Keyword Operator- Overloading Unary Operator- Operator Return Type- Overloading Assignment Operator (=)- Rules for Overloading Operators, Inheritance, Reusability- Types of Inheritance- Virtual Base Classes- Object as a Class Member- Abstract Classes- Advantages of Inheritance-Disadvantages of Inheritance

**UNIT – 4 Pointers & Binding Polymorphisms and Virtual Functions :** Pointer, Features of Pointers- Pointer Declaration- Pointer to Class- Pointer Object- The this Pointer- Pointer to Derived Classes and Base Class, Binding Polymorphisms and Virtual Functions, Introduction- Binding in C++- Virtual Functions- Rules for Virtual Function- Virtual Destructor

**UNIT – 5 Templates, Exception Handling :**Generic Programming with Templates, Need for Templates- Definition of class Templates- Normal Function Templates- Difference Between Templates and Macros- Linked Lists with Templates, Exception Handling- Principles of Exception Handling- The Keywords try throw and catch- Multiple Catch Statements –Specifying Exceptions.

**TEXT BOOKS:**
1. A First Book of C++, Gary Bronson, Cengage Learning.
2. The Complete Reference C++, Herbert Schildt, TMH.
3. Programming in C++, Ashok N Kamthane, Pearson 2nd Edition

**REFERENCE BOOKS:**
1. Object Oriented Programming C++, Joyce Farrell, Cengage.
2. C++ Programming: from problem analysis to program design, DS Malik, Cengage Learning

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### DATA WAREHOUSING AND DATA MINING

**Pre-requisites:** Data Structures, Algorithms, Probability & Statistics, Data Base Management Systems

**Course Objectives:**

The main objective of the course is to

- Introduce basic concepts and techniques of data warehousing and data mining
- Examine the types of the data to be mined and apply pre-processing methods on raw data
- Discover interesting patterns, analyze supervised and unsupervised models and estimate the accuracy of the algorithms.

**Course Outcomes:**

By the end of the course student will be able to

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Illustrate the importance of Data Warehousing, Data Mining and its functionalities and Design schema for real time data warehousing applications. | **K3, K6** |
| **CO2** | Demonstrate on various Data Preprocessing Techniques viz. data cleaning, data integration, data transformation and data reduction and Process raw data to make it suitable for various data mining algorithms. | **K3** |
| **CO3** | Choose appropriate classification technique to perform classification, model building and evaluation. | **K4** |
| **CO4** | Make use of association rule mining techniques viz. Apriori and FP Growth algorithms and analyze on frequent itemsets generation. | **K3, K4** |
| **CO5** | Identify and apply various clustering algorithm (with open source tools), interpret, evaluate and report the result. | **K2** |

**Mapping of course outcomes with program outcomes**

| CO | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | | 3 | 3 | | 2 | | | 1 | | | | |
| CO2 | 2 | 3 | 2 | | 1 | | | | | | | |
| CO3 | | 2 | | 1 | 1 | 2 | | 3 | | | | |
| CO4 | 2 | 3 | 2 | | 1 | | | | | | | |
| CO5 | | 2 | | 1 | 1 | 2 | | 3 | | | | |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**SYLLABUS:**

**UNIT–I: Data Warehousing and Online Analytical Processing:** Data Warehouse: Basic concepts, Data Warehouse Modelling: Data Cube and OLAP, Data Warehouse Design and Usage, Data Warehouse Implementation, Introduction: Why and What is data mining, What kinds of data need to be mined and patterns can be mined, Which technologies are used, Which kinds of applications are targeted.

**UNIT II: Data Pre-processing:** An Overview, Data Cleaning, Data Integration, Data Reduction, Data Transformation and Data Discretization.**[Text Book 1]**

**UNIT–III: Classification:** Basic Concepts, General Approach to solving a classification problem, Decision Tree Induction: Attribute Selection Measures, Tree Pruning, Scalability and Decision Tree Induction, Visual Mining for Decision Tree Induction.**[Text Book 1]**

**UNIT–IV: Association Analysis:** Problem Definition, Frequent Item set Generation, Rule Generation: Confident Based Pruning, Rule Generation in Apriori Algorithm, Compact Representation of frequent item sets, FP-Growth Algorithm. . **[Text Book 2]**

**UNIT–V: Cluster Analysis:** Overview, Basics and Importance of Cluster Analysis, Clustering techniques, Different Types of Clusters; K-means: The Basic K-means Algorithm, K-means Additional Issues, Bi-secting K Means, **[Text Book 2]**

**TEXT BOOKS:**

1. Data Mining concepts and Techniques, 3/e, Jiawei Han, Michel Kamber, Elsevier,2011.
2. Introduction to Data Mining: Pang-Ning Tan & Michael Steinbach, Vipin Kumar, Pearson,2012.

**REFERENCE BOOKS:**

1. Data Mining Techniques and Applications: An Introduction, Hongbo Du, Cengage Learning.
2. Data Mining: VikramPudi and P. Radha Krishna, Oxford Publisher.
3. Data Mining and Analysis - Fundamental Concepts and Algorithms; Mohammed J. Zaki, Wagner Meira, Jr, Oxford
4. Data Warehousing Data Mining & OLAP, Alex Berson, Stephen Smith, TMH.
   http://onlinecourses.nptel.ac.in/noc18_cs14/preview
5. (NPTEL course by Prof.PabitraMitra)
   http://onlinecourses.nptel.ac.in/noc17_mg24/preview
6. (NPTEL course by Dr. NandanSudarshanam& Dr. BalaramanRavindran)
   http://www.saedsayad.com/data_mining_map.htm

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### DATA SCIENCE

**Course Objectives:**
From the course the student will learn

- Provide you with the knowledge and expertise to become a proficient data scientist.
- Demonstrate an understanding of statistics and machine learning concepts that are vital for data science;
- Produce Python code to statistically analyse a dataset;
- Critically evaluate data visualisations based on their design and use for communicating stories from data

**Course Outcomes:**

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Acquire the knowledge and expertise to become a proficient data scientist | **K3** |
| **CO2** | Demonstrate an understanding of statistics and machine learning concepts that are vital for data science | **K3** |
| **CO3** | Explain how data is collected, managed and stored for data science | **K2** |
| **CO4** | Interpret the key concepts in data science, including their real-world applications and the toolkit used by data scientists | **K2** |
| **CO5** | Illustrate data collection and management scripts using MongoDB | **K3** |

**Mapping of course outcomes with program outcomes**

| CO | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 1 | 3 | 3 | | 2 | 2 | | 1 | | | | 1 |
| CO2 | 2 | 3 | 2 | | 1 | | | | | | | |
| CO3 | | 2 | | 1 | 1 | 2 | | 3 | | 1 | | |
| CO4 | 2 | 3 | 2 | | 1 | | | | | | | |
| CO5 | | 2 | | 1 | 1 | 2 | | 3 | | 1 | | |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**SYLLABUS:**

**UNIT-I: Introduction to Core Concepts and Technologies-** Introduction, Terminology, data science process, data science toolkit, Types of data, Example applications.

**UNIT-II: Data Collection and Management-** Introduction, Sources of data, Data collection and APIs, Exploring and fixing data, Data storage and management, using multiple data sources.

**UNIT III: Data Analysis-** Introduction, Terminology and concepts, Introduction to statistics,, Variance, Distribution properties and arithmetic, Samples/CLT, Basic machine learning algorithms, Linear regression, SVM, Naive Bayes.

**UNIT-IV: Data Visualisation-** Introduction, Types of data visualisation, **Data for visualisation-** Data types, Data encodings, Retinal variables, Mapping variables to encodings, Visual encodings.

**UNIT-V: Applications of Data Science-** Technologies for visualisation, Bokeh (Python), recent trends in various data collection and analysis techniques, various visualization techniques, application development methods of used in data science.

**TEXT BOOKS:**

1. "The Art of Data Science", 1st edition, Roger D. Peng and Elizabeth matsui, Lean Publications, 2015
2. "Algorithms for Data Science", 1st edition, **Steele**, Brian, **Chandler**, John, **Reddy**, Swarna, springers Publications, 2016

**REFERENCE BOOKS:**

1. Doing Data Science: Straight Talk From The Frontline, 1st edition, Cathy O'Neil and Rachel Schutt, O'Reilly, 2013
2. Mining of Massive Datasets, 2nd edition, Jure Leskovek, Anand Rajaraman and Jeffrey Ullman, v2.1, Cambridge University Press, 2014

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### ARTIFICAL INTELLIGENCE

**Course Objectives:** The student will be able to

- Know the methodology of Problem solving
- Implement basic AI algorithms
- Design and carry out an empirical evolution of different algorithms on a problem formalization

**Course Outcomes (COs)**: At the end of the course, student will be able to

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | **Understand** the fundamental concepts in Artificial Intelligence | **K2** |
| **CO2** | **Analyze** the applications of search strategies and problem reductions | **K2, K4,K5** |
| **CO3** | **Apply** the mathematical logic concepts. | **K2, K3** |
| **CO4** | **Develop** the Knowledge representations in Artificial Intelligence. | **K4** |
| **CO5** | **Explain** the Fuzzy logic systems. | **K6** |

***#Based on suggested Revised BTL***

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | | 3 | 3 | | 2 | | | 1 | | | | |
| **CO2** | 2 | 3 | 2 | | 1 | | | | | | | |
| **CO3** | | 2 | | 1 | 1 | 2 | | 3 | | | | |
| **CO4** | 2 | 3 | 2 | | 1 | | | | | | | |
| **CO5** | | 2 | | 1 | 1 | 2 | | 3 | | | | |

**(Please fill the above with Levels of Correlation, viz., L, M, H)**

**SYLLABUS:**

**UNIT I: Introduction to artificial intelligence:** Introduction ,history, intelligent systems, foundations of AI, applications, tic-tac-tie game playing, development of AI languages, current trends in AI.

**UNIT II: Problem solving: state-space search and control strategies:** Introduction, general problem solving, characteristics of problem

**Search Strategies:** exhaustive searches, heuristic search techniques, iterative-deepening A*, constraint satisfaction

**UNIT III: Logic concepts:** Introduction, propositional calculus, proportional logic, natural deduction system, axiomatic system, semantic tableau system in proportional logic, predicate logic

**UNIT IV: Knowledge representation:** Introduction, approaches to knowledge representation, knowledge representation using semantic network, extended semantic networks for KR **advanced knowledge representation techniques:** Introduction, conceptual dependency theory, script structure.

**UNIT V: Expert system and applications:** Introduction phases in building expert systems, expert system versus traditional systems, rule-based expert systems blackboard systems truth maintenance systems, application of expert systems, list of shells and tools.

**TEXT BOOKS:**

1. Artificial Intelligence- Saroj Kaushik, CENGAGE Learning,
2. Artificial intelligence, A modern Approach , 2nd ed, Stuart Russel, Peter Norvig, PEA
3. Artificial Intelligence- Rich, Kevin Knight, Shiv Shankar B Nair, 3rd ed, TMH
4. Introduction to Artificial Intelligence, Patterson, PHI

**REFERENCE BOOKS:**

1. Atificial intelligence, structures and Strategies for Complex problem solving, -George F Lugar, 5th ed, PEA
2. Introduction to Artificial Intelligence, Ertel, Wolf Gang, Springer
3. Artificial Intelligence, A new Synthesis, Nils J Nilsson, Elsevier

# MINOR COURSES SPECIALIZED TRACKS

**Department of Computer Science & Engineering**
**University College of Engineering, JNT University Kakinada**

## MINOR courses for SPECIALIZED TRACKS

**Note : 1. A student can opt Four subjects from any one of the specialized track @ 4 credits per subject**

2. **Concerned BoS can add or delete the subjects as per the decision of the board.**
3. **Pre requisites to be defined by the board for each course.**
4. **Compulsory MOOC/NPTEL Courses for 04 credits (02 courses@ 2 credits each)**
5. **Students need to select the courses which were not offered in their regular programme.**

| S.NO. | COURSE NAME | L-T-P | CR | PRE-REQ. | OFFERED TO |
|---|---|---|---|---|---|
| **TRACK-1 NETWORKING & SECURITY** | | | | | |
| 1 | TCP/IP Protocol Suite | 3-1-0 | 4 | | CSE/IT |
| 2 | Network Architecture and Design | 3-1-0 | 4 | | CSE/IT |
| 3 | Network Security | 3-1-0 | 4 | | CSE/IT |
| 4 | Cryptography | 3-1-0 | 4 | | CSE/IT |
| 5 | Computer Forensics | 3-1-0 | 4 | | CSE/IT |
| 6 | ethical hacking | 3-1-0 | 4 | | |
| **TRACK-2 SOFTWARE ENGINEERING** | | | | | |
| 1 | Software Metrics and Measurements | 3-1-0 | 4 | | CSE/IT |
| 2 | Software Verification and Validation | 3-1-0 | 4 | | CSE/IT |
| 3 | Software Architecture and Design Patterns | 3-1-0 | 4 | | CSE/IT |
| 4 | Software Project Management | 3-1-0 | 4 | | CSE/IT |
| 5 | Fault Tolerant Computing | 3-1-0 | 4 | | CSE/IT |
| 6 | Software Testing Methodologies | 3-1-0 | 4 | | |
| **TRACK-3 DISTRIBUTED & CLOUD COMPUTING** | | | | | |
| 1 | Enterprise Storage Systems | 3-1-0 | 4 | | CSE/IT |
| 2 | Parallel Algorithms | 3-1-0 | 4 | | CSE/IT |
| 3 | Cloud Networking | 3-1-0 | 4 | | CSE/IT |
| 4 | Cloud Computing | 3-1-0 | 4 | | CSE/IT |
| 5 | High Performance Computing | 3-1-0 | 4 | | CSE/IT |
| 6 | Advanced Computer Architecture | 3-1-0 | 4 | | |
| **TRACK-4 COMPUTATIONAL INTELLIGENCE** | | | | | |
| 1 | Artificial intelligence | 3-1-0 | 4 | | CSE/IT |
| 2 | Machine Learning | 3-1-0 | 4 | | CSE/IT |
| 3 | Natural Language Processing | 3-1-0 | 4 | | CSE/IT |
| 4 | Neural Networks | 3-1-0 | 4 | | CSE/IT |
| 5 | Multi Agent Systems | 3-1-0 | 4 | | CSE/IT |
| 6 | Deep Learning | 3-1-0 | 4 | | |

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**MINOR COURSES Specialized Tracks**

**Track-1 Networking & Security**

### TCP/IP PROTOCOL SUITE

**Course Objectives:**

- To provide insight about networks, topologies, and the key concepts.
- To gain comprehensive knowledge about the layered communication architectures (OSI and TCP/IP) and its functionalities.
- To understand the principles, key protocols, design issues, and significance of each layers in OSI and TCP/IP.

**Course Outcomes (COs)**: At the end of the course, student will be able to

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| CO1 | **Demonstrate** different network models for networking links OSI, TCP/IP, and get knowledge about various communication techniques, methods and protocol standards. | K2 |
| CO2 | **Analyze** Network layer protocols, services and Addressing in IPv4, IPv6 and Mobile IP | K4 |
| CO3 | **Analyze** various networking protocols like Unicast routing protocols and Congestion control protocols. | K4 |
| CO4 | **Discuss** various transport layer services like TCP, UDP and SCTP | K6 |
| CO5 | **Determine** application layer services and client server protocols working with the client server paradigms like WWW, HTTP, FTP, e-mail, SMTP and SNMP etc. | K5 |

***#Based on suggested Revised BTL***

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | | 3 | 2 | | 1 | | | | | | | |
| CO2 | 2 | 3 | 2 | 1 | | | | | | | | |
| CO3 | | 3 | 2 | | 1 | | | | | | | |
| CO4 | 3 | 3 | 2 | | 2 | | | | | | | |
| CO5 | 2 | 3 | 2 | | 2 | | | | | | | |

**(Please fill the above with Levels of Correlation, viz., L, M, H)**

**UNIT I: The OSI Model and TCP/IP Protocol Suit:** Protocol layers, The OSI model, TCP/IP Protocol suit, Addressing, Network topologies and N/W models: LAN, MAN, WAN and Internet, Underlying technologies: Wired Local Area network, Wireless LANs, Point to point WANs, Switched WAN, Connecting devices.

**UNIT II: Introduction to Network Layer:** Introduction, Switching, Packet Switching at Network layer, Network layer services, Other Network layer issues,

**IPV4 Address:** Introduction, Classful addressing, Classless addressing, Special addresses, NAT, **IPV4**: Introduction, Datagrams, Fragmentation, Options, Checksum, IP over ATM, Security, IP package. **IPV6 Package:** Introduction, Packet format, Translation from IPV4 to IPV6.

**UNIT – III: Unicast Routing Protocols:** Introduction, Intra and Inter-domain routing, Distance vector routing, RIP, Link state routing, OSPF, Path vector routing, BGP. **Congestion Control:** Approaches to congestion Control, Traffic aware routing, Admission control, Traffic throttling, Choke packets, Load shedding, Random early detection, QOS, Traffic shaping, Leaky bucket and token bucket.

**UNIT – IV : Transport Layer Services:** Introduction to Transport layer, Transport layer protocols, UDP, UDP services, UDP applications, UDP package, TCP services, TCP features, Segment, A TCP connection, Windows in TCP, Flow control, Error control, Congestion control, TCP timers, TCP package.

**UNIT –V: Introduction to Application Layer:** Client-server paradigm, peer-to-peer paradigm, Host-configuration: DHCP, Introduction, DHCP operation, Configuration, DNS: Need for DNS, Name space, DNS in the Internet, Resolution, DNS messages, Types of records, Compression, Encapsulation, Registrars, DDNS, Security of DNS. **WWW and HTTP:** Architecture, Web documents, HTTP. **Network Management:** SNMP, Management components, SMI, MID, SNMP, UDP ports, Security.

**TEXT BOOKS:**

1. Behrouz A Forouzan, “TCP/IP Protocol Suite”, TMH, 4th Edition.
2. Andrew S Tanenbaum, “Computer Networks”, Pearson Education/PHI, 5th Edition.

**REFERENCES BOOKS:**

1. BEHROUZ A. Forouzan, “Data Communications and Networks”, Fifth Edition TMH.
2. Achut S Godbole, Atul Kahate, “Data Communications and Networks”, TMH.
3. Mayank Dave, “Computer Networks”, CENGAGE.

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### NETWORK ARCHITECTURE AND DESIGN

**Course Objectives:**

- The goal of this course is to give a background in computer networks mechanisms and basic concepts used in the design of protocols and network architectures.
- These mechanisms include any of the protocol stack layers and a wide range of technologies either for wired networks or for wireless networks Outline.

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

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| CO1 | **Evolution** of the Network Architecture | K5 |
| CO2 | **Analyze** Routing and Inter-Networking Routing Algorithms | K4 |
| CO3 | **Analyze** Network and Transport Protocols IPv6. IPv4-IPv6 coexistence | K4 |
| CO4 | **Discuss** Resource Management Quality of Service principles. Quality of Service and Quality of Experience (QoS and QoE). | K6 |
| CO5 | **Determine** Software Defined Networks | K5 |

*#Based on suggested Revised BTL*

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | | 3 | 2 | | 1 | 2 | | | | | | |
| CO2 | 2 | 3 | 2 | 1 | | | | | | | | |
| CO3 | | 3 | 2 | | 1 | | | | | | | |
| CO4 | 3 | 3 | 2 | | 2 | | | | | | | |
| CO5 | 2 | 3 | 2 | | 2 | | | | | | | |

**(Please fill the above with Levels of Correlation, viz., L-1, M-2, H-3)**

**SYLLABUS:**

**UNIT-I: Evolution of the Network Architecture:** Internet Design Principles. Key protocols and their evolution. Internet structure, Exchange Points. Economic relationships among stakeholders. **Trends in the Evolution of the Network Architecture** Naming and Addressing. Addressing and Routing. Mobility. New Network Architectures.

**UNIT-II:** Routing and Inter-Networking Routing Algorithms. Classless Inter-domain Routing. Inter-domain Routing. IDR. BGP. IBGP. BGP attributes. Scalability of BGP. Transport Network (Backbone) Optical Transport Network. IP over SDH. IP over WDM/ASON. IP over WDM/Gigabit Ethernet. MPLS. From MPLS to GMPLS.

**UNIT-III:** New Network and Transport Protocols IPv6. IPv4-IPv6 coexistence. Mobile IP. IP Multicast. Other IP protocols (HIP). Multipath TCP. Other Transport protocols (QUIC).

**UNIT-IV:** Resource Management Quality of Service principles. Quality of Service and Quality of Experience (QoS and QoE). Integrated Services Architecture. Differentiated Services Architecture.

**UNIT-V:** 5G Network Architecture Framework for 5G networks. Core, edge and access networks. Virtualization technologies (NFV). Software Defined Networks (SDN). Orchestration and management. Slicing.

**TEXT BOOK:**

1. Architecture of Network Systems “The Morgan Kaufmann Series in Computer Architecture and Design”, 1st Edition

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### NETWORK SECURITY

**Course Objective:** To understand various protocols for network security to protect against the threats in the networks.

**Course Outcomes (COs)**: At the end of the course, student will be able to

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | **Understand** the basics of network security. | **K2** |
| **CO2** | **Illustrate** symmetric and asymmetric key cryptographic techniques | **K6** |
| **CO3** | **Explain** the security mechanism in IEEE 802.11 and 802.11i mechanisms. | **K3** |
| **CO4** | **Identify** Various network security applications, IPSec, Firewall, IDS, Web security, Email security and malicious software etc | **K3** |
| **CO5** | **Discuss** the Intrusion Detection system and approaches for IDS/IPS | **K6** |

***#Based on suggested Revised BTL***

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 2 | 2 | 3 | | | 1 | | | | | | |
| CO2 | 2 | 2 | 3 | 2 | 1 | 1 | | | | | | |
| CO3 | 1 | 2 | 1 | 3 | 2 | | | | | | | |
| CO4 | 1 | 2 | 2 | | 1 | 1 | | | | | | |
| CO5 | 1 | 2 | 2 | 1 | 2 | 3 | | | | | | |

**(Please fill the above with Levels of Correlation, viz., L, M, H)**

**SYLLABUS:**

**UNIT I: Introduction:** Computer Security Concepts, The OSI Security Architecture, Security Attacks, Security Services, Security Mechanisms, A model for Network Security **Key Management and Distribution:** Symmetric Key Distribution Using Symmetric Encryption, Symmetric Key Distribution using Asymmetric Encryption, Distribution of Public keys.

**UNIT II: User Authentication Protocols:** Introduction to Symmetric and Asymmetric Key Distribution, Remote User Authentication Principles, Remote User Authentication using Symmetric Encryption, Kerberos, Remote User Authentication using Asymmetric Encryption, Federated Identity Management

**UNIT III: Transport-Level Security:** Web Security issues, Secure Sockets Layer (SSL), Transport Layer Security(TLS),HTTPS, Secure Shell(SSH)
**Wireless Network Security:** IEEE 802.11 Wireless LAN Overview, IEEE 802.11i Wireless LAN Security, Wireless Application Protocol Overview.

**UNIT IV: Electronic Mail Security:** Pretty Good Privacy (PGP), S/MIME, Domain keys Identified Mail (DKIM). **IP Security:** IP Security Overview, IP Security Policy, Encapsulating Security Payload, Combining Security Associations, Internet Key Exchange, Cryptographic Suites.

**UNIT V: Web Security:** Web Security Considerations, Secure Socket Layer and Transport Layer Security, Secure Electronic Transaction.
**Intrusion Detection:** Overview, Approaches for IDS/IPS, Signature based IDS, Host based IDS/IPS

**TEXT BOOKS:**

1. Cryptography and Network Security-Principles and Practices: Williams Stallings Pearson Education $6^{th}$ edition.
2. Cryptography and Network Security: Atul Kahate, Mc Graw Hill, $3^{rd}$ edition
3. Introduction to Computer Networks & Cyber Security, Chwan Hwa Wu, J.David Irwin, CRC Press

**REFERENCE BOOKS:**

1. Cryptography and Network Security : Forouzan Mukhopadhyay, Mc Graw Hill, $3^{rd}$ Edition
2. Information Security, Principles, and Practice: Mark Stamp, Wiley India.
3. Introduction to Network Security: Neal Krawetz, CENGAGE Learning
4. Network Security and Cryptography: Bernard Menezes, CENGAGE Learning

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### CRYPTOGRAPHY

**Course Objectives:** The main objectives of the course are to
- Understand the mathematics behind Cryptography.
- Understand the security concerns and vulnerabilities
- Familiarize with different types of cryptosystems

**Course Outcomes (COs)**: At the end of the course, student will be able to

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | **Learn** the importance of number theory in designing cryptosystems; | **K2** |
| **CO2** | **Design** public and private key cryptosystems; | **K4** |
| **CO3** | **Discuss** cryptanalysis of various cryptosystems. | **K5** |
| **CO4** | **Analyze** different types of attacks on various cryptosystems | **K6** |
| **CO5** | **Create** an awareness for the design of various cryptographic primitives | **K2** |

***#Based on suggested Revised BTL***

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 1 | 2 | 3 | | 1 | 2 | | | | | | |
| **CO2** | 1 | 3 | 2 | 1 | 2 | 1 | | 1 | | | | |
| **CO3** | 1 | 2 | 2 | 1 | 1 | 3 | | | | | | |
| **CO4** | 2 | 2 | 3 | 2 | 1 | 2 | | | | | | |
| **CO5** | 2 | 2 | 1 | 3 | 3 | | | 2 | | | | |

**(Please fill the above with Levels of Correlation, viz., L, M, H)**

**SYLLABUS:**

**UNIT I: Basics of Algebra and Number Theory**: Integer Arithmetic, Modular Arithmetic, Algebraic structures, Prime Numbers, Fermat's and Euler's Theorem, Factorization, Chinese Remainder Theorem, Linear and Quadratic Congruence, Discrete Logarithms. **Introduction to Security**: Security Goals, Security services Security Mechanisms.

**UNIT II: Security Principles. Introduction to Cryptography:** Kerckhoff's Principle, Classification of Cryptosystems, Cryptanalytic attacks, Cipher Properties (Confusion, Diffusion).Traditional Secret Key Ciphers: Substitution Ciphers (mono alphabetic ciphers, poly alphabetic ciphers), Transposition Ciphers, Stream and Block Ciphers.

**UNIT III:** Data Encryption Standard (DES) (Fiestel and Non-Fiestel Ciphers, Structure of DES, DES Attacks, 2-DES, 3-DES), Advanced Encryption Standard (AES) (Structure, Analysis), Cryptographic Hash Functions– Properties, Secure HashAlgorithm-Message Authentication Code (MAC).

**Department of Computer Science & Engineering**
**University College of Engineering, JNT University Kakinada**

**UNIT IV:** Public Key Cryptosystems (PKC): Types of PKC, Trapdoor, one way functions, RSA Cryptosystem (Integer Factorisation Trapdoor ,Key Generation, Encryption, Decryption), ElGamal Cryptosystem (Discrete Logarithm Trapdoor, Key Generation, Encryption, Decryption), Diffie-Hellman Key Exchange Protocol.

**UNIT V:** Digital Signature: Signing, Verification, Digital signature forgery (Existential forgery, Selective forgery, Universal forgery), RSA Digital Signature Scheme ,ElGamal Signature Scheme. IP Security Overview, IP Security Architecture, Authentication Header, Encapsulating Security Payload, Intruders, Intrusion Detection.

**TEXT BOOKS:**

1. Behrouz A. Forouzan and DebdeepMukhopadhyay, Cryptography & Network Security, Second Edition, Tata McGraw Hill, New Delhi,2010
2. Douglas R. Stinson, "Cryptography: Theory and Practice", Third Edition, CRCPress.
3. William Stallings, "Cryptography and Network Security – Principles and Practices", Pearson Education, Fourth Edition,2006.

**REFERENCE BOOKS:**

1. AtulKahate, "Cryptography and Network Security", 2nd Edition, Tata McGraw Hill, 2003.
2. Bernard Menezes, Network Security and Cryptography-Cengage Learning India,2011
3. Bruce Schneier, "Applied Cryptography: Protocols, Algorthms, and Source Code in C", Second Edition, John Wiley and Sons Inc,2001.
4. Thomas Mowbray, "Cybersecurity : Managing Systems Conducting Testing, and Investigating Intrusions", John Wiley,2013
5. Wenbo Mao, " Modern Cryptography- Theory & Practice", Pearson Education,2006.

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### COMPUTER FORENSICS

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

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | **Ability to design**, implement computer forensics methods | **K2** |
| **CO2** | **develop** Evidence Collections and Crime Scene Computer Evidence Processing Steps | **K3** |
| **CO3** | **design** and implement Networks Forensics and validating forensic data, | **K3** |
| **CO4** | **design** computer forensic tool needs, computer forensics software tools | **K4** |
| **CO5** | **Ability** to understanding file systems, exploring Microsoft File Structures | **K3** |

**# Based on suggested Revised BTL**

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 1 | 2 | 1 | 2 | 3 | 1 | | 1 | | | | |
| **CO2** | 1 | 1 | 2 | 3 | | | | 3 | | | 2 | |
| **CO3** | 2 | 1 | 2 | 3 | 1 | | 1 | | | | | |
| **CO4** | | 2 | 1 | 3 | 2 | 1 | | | | | | |
| **CO5** | | 2 | 1 | 2 | 3 | 1 | | 1 | | | | |

**SYLLABUS:**

**UNIT-I: Computer Forensics Fundamentals:** What is Computer Forensics? Forensics in Law Enforcement, Computer Forensics Assistance to Human Resources/Employment Proceedings, Computer Forensics Services, Benefits of Professional Forensics Methodology, Steps taken by Computer Forensics Case Study of Computer Forensics Technology: Types of Military Computer Forensic Technology, Types of Law Enforcement.

**UNIT-II:** 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 Collection Artifacts, Collection Steps Controlling Contamination: The Chain of Custody 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.

**Department of Computer Science & Engineering**
**University College of Engineering, JNT University Kakinada**

**UNIT – III: Computer Forensics analysis and validation:** Determining what data to collect and analyze, validating forensic data, addressing data-hiding techniques, and performing remote acquisitions **Processing Crime and Incident Scenes:** Identifying digital evidence, collecting evidence in private-sector incident scenes, processing law enforcement crime scenes, preparing for a search, securing a computer incident or crime scene, seizing digital evidence at the scene, storing digital evidence, obtaining a digital hash, reviewing a case

**UNIT – IV: Current Computer Forensic tools:** evaluating computer forensic tool needs, computer forensics software tools, computer forensics hardware tools, validating and testing forensics software E-Mail Investigations: Exploring the role of e-mail in investigation, exploring the roles of the client and server in e-mail, investigating e-mail crimes and violations, **Cell phone and mobile device forensics**: Understanding mobile device forensics, understanding acquisition procedures for cell phones and mobile devices.

**UNIT – V: Working with Windows and DOS Systems**: understanding file systems, exploring Microsoft File Structures, Examining NTFS disks, Understanding whole disk encryption, windows registry, Microsoft startup tasks, MS-DOS startup tasks, virtual machines.

**TEXT BOOKS:**

1. Computer Forensics, Computer Crime Investigation By John R. Vacca, Firewall Media, New Delhi.
2. Computer Forensics And Investigations By Nelson, Phillips Enfinger, Steuart, Cengage Learning

**REFERENCE BOOKS:**

1. Real Digital Forensics By Keith J. Jones, Richard Bejtiich, Curtis W. Rose, Addison-Wesley Pearson Education
2. Forensic Compiling, A Tractitioneris Guide By Tony Sammes And Brian Jenkinson, Springer International Edition.
3. Computer Evidence Collection & Presentation By Christopher L.T. Brown, Firewall Media.
4. Homeland Security, Techniques & Technologies By Jesus Mena, Firewall Media.
5. Software Forensics Collecting Evidence From The Scene Of A Digital Crime By Robert M. Slade, Tmh 2005

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### ETHICAL HACKING

**Course Objectives:**

- The aim of the course is to introduce the methodologies and framework of ethical hacking for enhancing the security.
- The course includes-Impacts of Hacking; Types of Hackers; Information Security Models, Information Security Program, Business Perspective, Planning a Controlled Attack

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

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | **Ability** to understand the concepts of Ethical Hacking and Penetration Testing | **K5** |
| **CO2** | **Analyze** the information gathering techniques like Footprinting, social Engineering attacks and it's countermeasures. | **K4** |
| **CO3** | **Analyze** the concepts of Scanning and Enumeration | **K4** |
| **CO4** | **Demonstrate** the Web Hacking techniques and protecting the system from being attacked | **K2** |
| **CO5** | **Elaborate** the concepts of password cracking techniques and Steganography | **K6** |

***#Based on suggested Revised BTL***

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 1 | 1 | 2 | | 2 | 1 | | 3 | | | | |
| **CO2** | 1 | 2 | 3 | | 1 | 2 | | | | | | |
| **CO3** | | | 1 | | 3 | | | | | | | |
| **CO4** | 1 | 2 | 3 | | 1 | 2 | | | | | | |
| **CO5** | | 1 | 2 | 3 | | 1 | 2 | | | | | |

**(Please fill the above with Levels of Correlation, viz., L, M, H)**

**SYLLABUS:**

**UNIT-I: Introduction to Ethical Hacking, Ethics and Legality:** Defining Ethical Hacking: Important Terminologies, Purpose of Ethical Hacking, Phases of Ethical Hacking, types of hacking technologies, types of ethical hacks, penetration test, vulnerability assessment vs penetration test, categories and types of penetration test.

**UNIT- II : Gathering Target Information:** Reconnaissance, Footprinting and Social Engineering Reconnaissance, Information-Gathering Methodology: Footprinting, Using Google to Gather Information, Understanding DNS Enumeration, Understanding Whois and ARIN Lookups, Identifying Types of DNS Records, Using Trace route in Foot printing, Understanding Email Tracking, Understanding Web Spiders.

**UNIT- III: Gathering Network and Host Information:** Scanning and Enumeration: Scanning, Types of Port Scanning, Ping Sweep Techniques, nmap Command Switches, Scan Types, TCP Communication Flag Types, War-Dialing Techniques, Banner Grabbing and OS Fingerprinting Techniques, Scanning Anonymously, Enumeration, Null Sessions, SNMP Enumeration, Windows 2000 DNS Zone Transfer.

**UNIT- IV: Web Hacking:** Google, Web Servers, Web Application Vulnerabilities, and Web-Based Password Cracking Techniques, How Web Servers Work, Types of Web Server Vulnerabilities, Attacking a Web Server, Patch-Management Techniques, Web Server Hardening Methods, Web Application Vulnerabilities, Web Application Threats and Countermeasures, Google Hacking, Web-Based Password-Cracking Techniques.

**UNIT- V: System Hacking:** Password Cracking, Escalating Privileges, and Hiding Files, The Simplest Way to Get a Password, Types of Passwords, Passive Online Attacks, Active Online Attacks, Offline Attacks, Nonelectronic Attacks, Cracking a Password, Understanding the LAN Manager Hash, Cracking Windows 2000 Passwords, Redirecting the SMB Logon to the Attacker, SMB Relay MITM Attacks and Countermeasures, NetBIOS DoS Attacks, Password-Cracking Countermeasures, Understanding Keyloggers and Other Spyware Technologies.

**TEXT BOOKS:**

1. Kimberly Graves, "Certified Ethical Hacker", Wiley India Pvt. Ltd, 2010.
2. Rafay Baloch, “Ethical Hacking and Penetration Testing Guide”, 2014.

**REFERENCE BOOKS:**

1. HACKING: BE a Hacker with Ethics, Harsh Bothra, Khanna Publications, 2019
2. Mastering Modern Web Penetration Testing, Prakhar Prasad, Packt Publishing, October 2016

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**TRACK-2 Software Engineering**

### SOFTWARE METRICS AND MEASUREMENTS

**Course Objectives:**

- To provide a solid background knowledge about software metrics.
- To educate various metrics and models to assess software.
- To provide hands on experience on using and implementing metrics.

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

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | **Acquire** knowledge about software metrics. | **K5** |
| **CO2** | **Assess** software using various metrics and models. | **K4** |
| **CO3** | **Make Use of** and implement different metrics. | **K2** |
| **CO4** | **distinguishing** characteristics of object-oriented metrics | **K5** |
| **CO5** | **Discuss** Commercial software measurement tools | **K6** |

***# Based on suggested Revised BTL***

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 1 | 2 | 3 | | 1 | 2 | | | | | | |
| **CO2** | 2 | 2 | 3 | | 3 | 1 | 1 | 1 | | | 1 | |
| **CO3** | | 2 | 3 | | 3 | | 1 | | | | 1 | |
| **CO4** | 2 | 3 | 2 | 1 | 2 | | | | | | | |
| **CO5** | | 2 | 3 | | 3 | 1 | 1 | | | | 1 | |

**(Please fill the above with Levels of Correlation, viz., L-1, M-2, H-3)**

**SYLLABUS:**

**UNIT-I**: **Basics of measurement:** scope of software metrics, representational theory of measurement, measurement and models, measurement scales, meaningfulness in measurement, goal-based framework for software, measurement, Software-metrics, data collection and analysis: What is good data, how to define the data, how to collect the data, how to store and extract data, analyzing software-measurement data, frequency distributions, various statistical techniques.

**UNIT-II**: **Measuring internal product attributes:** Measuring external product attributes: Modelling software quality, measuring aspects of software quality, software reliability, basics of software reliability, software reliability problem, parametric reliability growth models, predictive accuracy, recalibration of software-reliability growth predictions, importance of operational environment, wider aspects of software reliability.

**UNIT-III: Metrics for object-oriented systems:** The intent of object-oriented metrics, distinguishing characteristics of object-oriented metrics, various object-oriented metric suites – LK suite, CK suite and MOOD metrics.

**UNIT-IV: Metrics for component-based systems:** The intent of component-based metrics, distinguishing characteristics of component-based metrics, various component-based metrics. Resource measurement: Measuring productivity, teams, tools, and methods.

**UNIT-V**: **Measures and metrics of industry leaders** – Measures, metrics and innovation –Measurements, metrics and outsource litigation – Measurements, metrics and behavioral changes – Commercial software measurement tools. Measuring Process Maturity - Process Capability - Value of Process Improvement – Process Adoption – Process Compliance. Function Point Metrics to Measure Software Process Improvement - Software Process Improvement Sequences.

**TEXT BOOKS:**

1. N.E. Fenton and S.L. Pfleeger, *Software Metrics – A Rigorous and Practical Approach*, International Thomson Computer Press , 1997
2. S.H. Kan, *Metrics and Models in Software Quality Engineering*, Addison Wesley, New York , 2004

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### SOFTWARE VERIFICATION AND VALIDATION

**Course Objectives:** This course makes students understand the concepts and theory related to software testing.

- Understand different testing techniques used in designing test plans, developing test suites, and evaluating test suite coverage.
- Understand how software developers can integrate a testing framework into code development in order to incrementally develop and test code.

**Course outcomes:** At the end of the course, student will be able to

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| CO1 | **Acquire** knowledge about Faults and Failures, Correctness and reliability, Testing and debugging. | K5 |
| CO2 | **Explain** Reviews in testing lifecycle | K3 |
| CO3 | **Analyze** structural testing and analysis, error-oriented testing and analysis | K5 |
| CO4 | **distinguishing** Test generations from requirement and Test generation pats | K5 |
| CO5 | **Discuss** The ISO 9000 quality standards | K6 |

***# Based on suggested Revised BTL***

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 1 | 2 | 3 | | 2 | | | 3 | | | 1 | |
| CO2 | | 2 | 3 | | | 1 | | 1 | | | | |
| CO3 | 1 | 2 | 3 | | 1 | 2 | | | | | | |
| CO4 | | 1 | 2 | 1 | 1 | 2 | 3 | | | | | |
| CO5 | 1 | 2 | 3 | | 1 | 2 | | | | | | |

**(Please fill the above with Levels of Correlation, viz., L-1, M-2, H-3)**

**SYLLABUS:**

**UNIT I: Introduction**: Terminology, evolving nature of area, Errors, Faults and Failures, Correctness and reliability, Testing and debugging, Static and dynamic testing, Exhaustive testing: Theoretical foundations: impracticality of testing all data, impracticality of testing all paths, no absolute proof of correctness.

**UNIT II: Software Verification and Validation basics**: Introduction, Verification, Verification Workbench, Methods of Verification, Types of reviews on the basis od Stage Phase, Entities involved in verification, Reviews in testing lifecycle, Coverage in Verification, Concerns of Verification, Validation, Validation Workbench, Levels of Validation, Coverage in Validation, Acceptance Testing, Management of Verification and Validation, Software development verification and validation activities.

**UNIT III: Software Verification and Validation Approaches and their Applicability**: Software technical reviews; Software testing: levels of testing - module, integration, system, regression; Testing techniques and their applicability-functional testing and analysis, structural testing and analysis, error-oriented testing and analysis, hybrid approaches, integration strategies, transaction flow analysis, stress analysis, failure analysis, concurrency analysis, performance analysis; Proof of correctness; simulation and prototyping; Requirement tracing.

**UNIT IV: Test Generation**: Test generations from requirements, Test generation pats, Data flow analysis, Finite State Machines models for flow analysis, Regular expressions based testing, Test Selection, Minimizations and Prioritization, Regression Testing. Program Mutation Testing: Introduction, Mutation and mutants, Mutation operators, Equivalent mutants, Fault detection using mutants, Types of mutants, Mutation operators for C and Java.

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

**TEXT BOOKS :**

1. Software Engineering, A practitioner's Approach- Roger S. Pressman, 6th edition.McGrawHill International Edition.
2. Software Engineering- Sommerville, 7th edition, Pearson education.

**REFERENCE BOOKS:**

1. Software Verification and Validation: An Engineering and Scientific Approach,
2. Marcus S. Fisher, Springer, 2007
3. Foundations of Software Testing, Aditya P. Mathur,Pearson Education, 2008Software Testing: Principles and Practices,SrinivasanDesikan, Gopalaswamy Ramesh, Pearson Education India,2006

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### SOFTWARE ARCHITECTURE AND DESIGN PATTERNS

**Course Objectives:** The main objective is to introduce the student to architecture of software and design Patterns. The other objectives include

- Understand the creational and structural patterns.
- Be capable of applying his knowledge to create an architecture for given application.

**Pre-requisites: Basics of C, C++**

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

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | **Understand** the architecture, creating it and moving from one to any, different structural patterns | **K1** |
| **CO2** | **Analyze** the architecture and build the system from the components | **K4** |
| **CO3** | **Design** creational and structural patterns. | **K5** |
| **CO4** | **Learn** about behavioral patterns. | **K1** |
| **CO5** | **Discuss** about case study in utilizing architectural structures. | **K6** |

***#Based on suggested Revised BTL***

**Mapping of course outcomes with program outcomes**

| CO | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 2 | 1 | 3 | | 2 | | | | | | | |
| CO2 | 1 | 3 | 2 | | 2 | 1 | | | | | | |
| CO3 | | | 3 | | 3 | | | 2 | | | 1 | |
| CO4 | 2 | 3 | | 2 | | | | 1 | | | 1 | |
| CO5 | 1 | 1 | 2 | | | 2 | | 2 | | | 3 | |

**(Please fill the above with Levels of Correlation, viz., L, M, H)**

**SYLLABUS:**

**UNIT- I: Envisioning Architecture:** The Architecture Business Cycle, What is Software Architecture, Architectural patterns, reference models, reference architectures, architectural structures and views.

**Creating architecture:** Quality Attributes, Achieving qualities, Architectural styles and patterns, designing the Architecture, Documenting software architectures, Reconstructing Software Architecture

**UNIT –II: Analyzing Architectures:** Architecture Evaluation, Architecture design decision making, ATAM, CBAM. **Moving from one system to many:** Software Product Lines, Building systems from off the shelf components, Software architecture in future.

**UNIT-III: Patterns:** Pattern Description, Organizing catalogs, role in solving design problems , Selection and usage. **Creational and Structural patterns:** Abstract factory, builder, factory method, prototype, singleton, adapter, bridge, composite, façade, flyweight, Proxy.

**UNIT- IV: Behavioral patterns:** Chain of responsibility, command, Interpreter, iterator, mediator, memento, observer, state, strategy, template method, visitor.

**UNIT –V: Case Studies:** A-7E – A case study in utilizing architectural structures, The World Wide Web - a case study in interoperability, Air Traffic Control – a case study in designing for high availability, Celsius Tech – a case study in product line development

**TEXT BOOKS:**

1. Len Bass,Paul Clements&Rick Kazman, *Software Architecture in Practice*, 2nd Edition, Pearson Education, 2003.
2. Erich Gamma, *Design Patterns*, 1st Edition, Pearson Education,1995.

**REFERENCE BOOKS:**

1. Luke Hohmann , *Beyond Software architecture*, Addison wesley, 2003.
2. David M. Dikel, David Kane and James R. Wilson, *Software architecture*, 1st Edition, Prentice Hall,2001
3. F.Buschmann , *Pattern Oriented Software Architecture*, Wiley&Sons,1st Edition,2001

**Web References :**

1. http://en.wikibooks.org/wiki/Introduction_to_Software_Engineering/Architecture/Design_Patterns

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### SOFTWARE PROJECT MANAGEMENT

**Course Objectives:** At the end of the course, the student shall be able to:

- To describe and determine the purpose and importance of project management from the perspectives of planning, tracking and completion of project.
- To compare and differentiate organization structures and project structures.
- To implement a project to manage project schedule, expenses and resources with the application of suitable project management tools

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

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| CO1 | **Apply** the process to be followed in the software development life-cycle models. | K3 |
| CO2 | **Apply** the concepts of project management• & planning. Implement the project plans through managing people, communications and change. | K3 |
| CO3 | **Conduct** activities necessary to successfully complete and close the Software projects. | K2 |
| CO4 | **Implement** communication, modeling, and construction & deployment practices in software development. | K6 |
| CO5 | **Illustrate** Life Cycle Expectations Pragmatic Software | K6 |

**#based on suggested Revised BTL**

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 2 | 2 | 3 | | 1 | 2 | | 3 | | | 1 | |
| CO2 | 1 | 2 | 3 | | 1 | 2 | | 3 | | | 1 | |
| CO3 | 1 | 2 | 3 | 1 | 2 | | | | | | | |
| CO4 | | 2 | 2 | | 3 | 1 | | 1 | | | | |
| CO5 | 1 | 2 | 3 | 1 | 2 | | | | | | | |

**(Please fill the above with Levels of Correlation, viz., L-1, M-2, H-3)**

**SYLLABUS:**

**UNIT-I: Conventional Software Management:** The waterfall Model, Conventional Software Management Performance, Evolution of Software Economics: software Economics. Pragmatic Software Cost Estimation. **Improving Software Economics:** Reducing Software Product Size, Improving Software Processes, Improving Team Effectiveness, Improving Automation, Achieving Required Quality, Peer Inspections.

**UNIT-II: Conventional and Modern Software Management:** Principles of Conventional Software Engineering, Principles of Modern Software Management, and Transitioning to an interactive Process. **Life Cycle Phases:** Engineering and Production Stages Inception, Elaboration, Construction, Transition phases.

**UNIT-III: Artifacts of the Process:** The Artifact Sets. Management Artifacts, Engineering Artifacts, Programmatic Artifacts. **Model Based Software Architectures:** A Management Perspective and Technical Perspective.

**UNIT-IV: Flows of the Process:** Software Process Workflows. Inter Trans Workflows. **Checkpoints of the Process:** Major Mile Stones, Minor Milestones, Periodic Status Assessments. **Interactive Process Planning:** Work Breakdown Structures, Planning Guidelines, Cost and Schedule Estimating. Interaction Planning Process, Pragmatic Planning.

**UNIT-V: Project Organizations and Responsibilities:** Line-of-Business Organizations, Project Organizations and Evolution of Organizations. **Process Automation:** Building Blocks, the Project Environment. Project Control and Process Instrumentation: Server Care Metrics, Management Indicators, Quality Indicators, Life Cycle Expectations Pragmatic Software.

**TEXT BOOKS:**

1. Software Project Management, Walker Royce, Pearson Education, 2005.
2. Software Project Management, Bob Hughes, 4th edition, Mike Cotterell, TMH.

**REFERENCE BOOKS:**

1. Software Project Management, Joel Henry, Pearson Education.
2. Software Project Management in practice, Pankaj Jalote, Pearson Education, 2005.
3. Effective Software Project Management, Robert K.Wysocki, Wiley,2006.

**Department of Computer Science & Engineering**
**University College of Engineering, JNT University Kakinada**

## FAULT TOLERANT COMPUTING

**Course Objectives:**

- To understand the fault diagnosis and tolerant design approach.
- To illustrate the framework of test pattern generation using semi and full automatic approach.

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

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| CO1 | **Know** the fundamental concepts in fault tolerant design. | K1 |
| CO2 | **Design** requirements of self check-in circuits. | K4 |
| CO3 | **Design** for testability rules and techniques for combinational circuits. | K4 |
| CO4 | **Discuss** about scan architectures. | K6 |
| CO5 | **Design** of built-in-self test. | K4 |

***#Based on suggested Revised BTL***

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | | 3 | 2 | | 1 | | | | | | | |
| CO2 | 2 | 3 | 2 | 1 | | | | | | | | |
| CO3 | | 3 | 2 | | 1 | | | | | | | |
| CO4 | 3 | 3 | 2 | | 2 | | | | | | | |
| CO5 | 2 | 3 | 2 | | 2 | | | | | | | |

**(Please fill the above with Levels of Correlation, viz., L, M, H)**

**SYLLABUS:**

**UNIT – I: Fault Tolerant Design:** Basic concepts: Reliability concepts, Failures & faults, Reliability and Failure rate, Relation between reliability and mean time between failure, maintainability and availability, reliability of series, parallel and parallel-series combinational circuits.

**Fault Tolerant Design:** Basic concepts-static, dynamic, hybrid, triple modular redundant system (TMR), 5MR reconfiguration techniques, Data redundancy, Time redundancy and software Redundancy concepts.

**UNIT – II: Self Checking circuits & Fail safe Design:** Self Checking Circuits: Basic concepts of self checking circuits, Design of Totally self checking checker, Checkers using m out of n codes, Berger code, Low cost residue code.

**Fail Safe Design:** Strongly fault secure circuits, fail safe design of sequential circuits using partition theory and Berger code, totally self checking PLA design.

**UNIT – III: Design for Testability:** Design for testability for combinational circuits: Basic concepts of Testability, Controllability and observability, The Reed Muller's expansion technique, use of control and syndrome testable designs.
**Design for testability by means of scan:** Making circuits Testable, Testability Insertion, Full scan DFT technique- Full scan insertion, flip-flop Structures, Full scan design and Test, Scan Architectures-full scan design, Shadow register DFT, Partial scan methods, multiple scan design, other scan designs.

**UNIT – IV: Logic Built-in-self-test:** BIST Basics-Memory-based BIST,BIST effectiveness, BIST types, Designing a BIST, Test Pattern Generation-Engaging TPGs, exhaustive counters, ring counters, twisted ring counter, Linear feedback shift register, Output Response Analysis- Engaging ORA's, One's counter, transition counter, parity checking, Serial LFSRs, Parallel Signature analysis, BIST architectures-BIST related terminologies, A centralized and separate Board-level BIST architecture, Built-in evaluation and self test(BEST), Random Test socket(RTS), LSSD On-chip self test, Self –testing using MISR and SRSG, Concurrent BIST.

**UNIT – V: Standard IEEE Test Access Methods:** Boundary Scan Basics, Boundary scan architecture- Test access port, Boundary scan registers, TAP controller, the decoder unit, select and other units, Boundary scan Test Instructions-Mandatory instructions, Board level scan chain structure-One serial scan chain, multiple-scan chain with one control test port, multiple-scan chains with one TDI,TDO but multiple TMS, Multiple-scan chain, multiple access port, RT Level boundary scan-inserting boundary scan test hardware for CUT, Two module test case, virtual boundary scan tester, Boundary Scan Description language

**TEXT BOOKS:**
1. Fault Tolerant & Fault Testable Hardware Design- Parag K.Lala, 1984,PHI
2. Digital System Test and Testable Design using HDL models and Architectures - Zainalabedin Navabi, Springer International Edition.

**REFERENCE BOOKS:**
1. Digital Systems Testing and Testable Design-Miron Abramovici, Melvin A.Breuer and Arthur D. Friedman, Jaico Books
2. Essentials of Electronic Testing- Bushnell & Vishwani D.Agarwal, Springers.
3. Design for Test for Digital IC's and Embedded Core Systems- Alfred L. Crouch, 2008, Pearson Education.

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### SOFTWARE TESTING METHODOLOGIES

**Course Objectives:**
- learn to test software effectively
- Learn practical ways to design high quality tests during all phases of software development.
- Learn the theory behind criteria-based test design and to apply that theory in practice. Topics include test design, test automation, test coverage criteria, and how to test software in cutting-edge software development environments.

**Pre-requisites: Basics of C, C++**

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

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | **Explain** about Software Testing Life Cycle | **K3** |
| **CO2** | **Discuss** about Black Box testing techniques | **K6** |
| **CO3** | **Demonstrate** Structured Walkthroughs | **K5** |
| **CO4** | **Compare** Progressives Vs regressive testing | **K4** |
| **CO5** | **Explain** Software Quality metrics, | **K3** |

***#Based on suggested Revised BTL***

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | | 3 | 2 | | 1 | | | | | | | |
| CO2 | 2 | 3 | 2 | 1 | | | | | | | | |
| CO3 | | 3 | 2 | | 1 | | | | | | | |
| CO4 | 3 | 3 | 2 | | 2 | | | | | | | |
| CO5 | 2 | 3 | 2 | | 2 | | | | | | | |

**(Please fill the above with Levels of Correlation, viz., L-1, M-2, H-3)**

**SYLLABUS:**

**UNIT I: Software Testing:** Introduction, Evolution, Myths & Facts, Goals, Psychology, Definition, Model for testing, Effective Vs Exhaustive Software Testing. **Software Testing Terminology and Methodology:** Software Testing Terminology, Software Testing Life Cycle, relating test life cycle to development life cycle Software Testing Methodology.

**UNIT II: Verification and Validation:** Verification & Validation Activities, Verification, Verification of Requirements, High level and low level designs, How to verify code, Validation **Dynamic Testing I: Black Box testing techniques:** Boundary Value Analysis, Equivalence class Testing, State Table based testing, Decision table based testing, Cause-Effect Graphing based testing, Error guessing

**Department of Computer Science & Engineering**
**University College of Engineering, JNT University Kakinada**

**UNIT III: Dynamic Testing II: White-Box Testing:** need, Logic coverage criteria, Basis path testing, Graph matrices, Loop testing, data flow testing, mutation testing **Static Testing:** inspections, Structured Walkthroughs, Technical reviews

**UNIT IV: Validation activities:** Unit testing, Integration Testing,. Function testing, system testing, acceptance testing **Regression testing:** Progressives Vs regressive testing, Regression testability, Objectives of regression testing, When regression testing done?, Regression testing types, Regression testing techniques

**UNIT V: Efficient Test Suite Management:** Test case deisgnWhy does a test suite grow, Minimizing the test suite and its benefits, test suite prioritization, Types of test case prioritization, prioritization techniques, measuring the effectiveness of a prioritized test suite **Software Quality Management:** Software Quality metrics, SQA models **Debugging:** process, techniques, correcting bugs, Basics of testing management tools, test link and Jira.

**TEXT BOOKS:**

1. Software Testing, Principles and Practices, Naresh Chauhan, Oxford
2. Foundations of Software testing, Aditya P Mathur, 2ed, Pearson
3. Software Testing- Yogesh Singh, CAMBRIDGE

**REFERENCE BOOKS:**

1. Software testing techniques - Baris Beizer, International Thomson computer press, second edition.
2. Software Testing, Principles, techniques and Tools, M G Limaye, TMH
3. Effective Methods for Software testing, Willian E Perry, 3ed, Wiley

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**TRACK- 3 Distributed & Cloud Computing**

### ENTERPISE STORAGE SYSTEMS

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

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| CO1 | **Evaluate** various storage classifications and technologies. | K5 |
| CO2 | **Analyze** storage architectures, processes, components and how they relate to virtualization. | K4 |
| CO3 | **Justify** the implementation of a range of storage solutions to enable business continuity. | K5 |
| CO4 | **Analyze** storage security design, implementation, monitoring and management | K4 |
| Co5 | **Apply** The Business Continuity applications in a systems. | K4 |

***#Based on suggested Revised BTL***

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 2 | 2 | 3 | | 1 | | 1 | | | | | |
| CO2 | 2 | 3 | 3 | | 1 | 2 | 1 | | | | | |
| CO3 | 2 | 2 | 2 | | | 1 | | | | | | |
| CO4 | 3 | 3 | 3 | | 2 | 1 | 1 | | | | | |
| C05 | 2 | 1 | 2 | | 2 | 3 | | 3 | | | | |

**(Please fill the above with Levels of Correlation, viz., L-1, M-2, H-3)**

**SYLLABUS:**

**UNIT-I: STORAGE SYSTEMS:** Data Classification, Storage Evolution and Data Centre infrastructure. Host components, Connectivity, Storage, and Protocols. Components of a disk drive, physical disk and factors affecting disk drive performance. RAID level performance and availability considerations. Components and benefits of an intelligent storage system.

**UNIT-II: STORAGE NETWORKING TECHNOLOGIES:** Direct-Attached Storage (DAS) architecture, Storage Area Network (SAN) attributes components, topologies, connectivity options and zoning. FC protocol stack, addressing, flow control, and classes of service. Networked Attached Storage (NAS) components, protocols, IP Storage Area Network (IP SAN) iSCSI, FCIP and FCoE architecture. Content Addressed Storage (CAS) elements, storage, and retrieval processes.

**UNIT-III: VIRTUALIZATION:** Block-level and file-level storage virtualization technology, virtual provisioning and cloud computing.

**UNIT-IV: BUSINESS CONTINUITY:** Business Continuity measurement, terminologies, and planning. Backup designs, architecture, topologies, and technologies in SAN and NAS environments. Local and Remote replication using host and array-based replication technologies such as Synchronous and Asynchronous methods.

**UNIT-V: STORAGE SECURITY AND MANAGEMENT:** Storage security framework and various security domains. Security implementation in SAN, NAS and IP-SAN networking. Monitoring and Storage management activities and challenges

**TEXT BOOKS**:

1. EMC, “Information Storage and Management” Wiley; 2 edition [ISBN: 978-0470294215], 2012.
2. Richard Barker, Paul Massiglia, “Storage area network essentials”, Wiley New York [ISBN: 978-0471034452] 2002.

**REFERENCE BOOKS**:

1. Ulf Troppens, Rainer Erkens, Wolfgang Mueller-Friedt, Rainer Wolafka, Nils Haustein, “Storage Networks Explained” Wiley; 2 edition, [ISBN: 978-0470741436], 2009
2. W. Curtis Preston, “Using SANs and NAS”, O'Reilly & Associates Sebastopol, Calif. [ISBN: 978-0596001537], 2002
3. Himanshu Dwivedi, “Securing storage”, Addison-Wesley Upper Saddle River, NJ [ISBN: 978-0321349958], 2006.

**Department of Computer Science & Engineering**
**University College of Engineering, JNT University Kakinada**

## PARALLEL ALGORITHMS

**Course Objectives :**

- To design the parallel algorithms for real world problems.
- To understand a wide variety of parallel architectures.
- To implement the parallel algorithms on available parallel systems
- To design algorithms suited for multiprocessor systems using MPI and OpenMP
- To analyze the parallel algorithms

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

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| CO1 | **How** to design the parallel algorithms for real world problems | K2 |
| CO2 | **Explain** wide variety of parallel architectures. | K3, K5 |
| CO3 | **Describe** and implement the parallel algorithms on available parallel Systems. | K2, K3 |
| CO4 | **Able to design** algorithms suited for multiprocessor systems using MPI And OpenMP | K2 |
| CO5 | **Analyze** the parallel algorithms. | K6 |

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 1 | 3 | 3 | | 2 | 1 | | | | | | |
| CO2 | 1 | 1 | 2 | | 1 | | | | | | | |
| CO3 | | 2 | 3 | | 2 | 3 | | | | | | |
| CO4 | 2 | 3 | 3 | | 3 | 2 | 1 | | | | | |
| CO5 | 1 | 1 | 2 | | 1 | 2 | | | | | | |

**(Please fill the above with Levels of Correlation, viz., L-1, M-2, H-3)**

**SYLLABUS:**

**UNIT- I: Introduction** Computational demand in various application areas, advent of parallel processing, terminology-pipelining, Data parallelism and control parallelism-Amdahl's law.

**UNIT II: Scheduling:** Organizational features of Processor Arrays, Multi processors and multi-computers. Mapping and scheduling aspects of algorithms. Mapping into meshes and hyper cubes-Load balancing-List scheduling algorithm Coffman-graham scheduling algorithm for parallel processors.

**UNIT III: Algorithms** Elementary Parallel algorithms on SIMD and MIMD machines, Analysis of these algorithms. Matrix Multiplication algorithms on

SIMD and MIMD models. Fast Fourier Transform algorithms. Implementation on Hyper cube architectures. Solving linear file -system of equations, parallelizing aspects of sequential methods back substitution and Tri diagonal.

**UNIT IV: Sorting:** Parallel sorting methods, Odd-even transposition Sorting on processor arrays, Biotonic ,merge sort on shuffle - exchange ID , Array processor,2D-Mesh processor and Hypercube Processor Array. Parallel Quick-sort on Multi processors. Hyper Quick sort on hypercube multi computers. Parallel search operations. Ellis algorithm and Manber and ladner's Algorithms for dictionary operations.

**UNIT V: Searching** Parallel algorithms for Graph searching, All Pairs shortest paths and minimum cost spanning tree. Parallelization aspects of combinatorial search algorithms with Focus on Branch and Bound Methods and Alpha-beta Search methods.

**TEXT BOOK:**

1. Parallel computing theory and practice, Michel J.Quinn
2. Programming Parallel Algorithms, Guy E. Blelloch, Communications of the ACM
3. Parallel Programming in C with MPI and OpenMP, Michael J. Quinn, McGraw Hill Education (India) Pvt. Ltd, 2003, sixteenth reprint 2016.

**REFERENCE BOOKS:**

1. Parallel computing ,Theory and Practice,2nd edition,M.J.Quinn, McGraw Hill Education(India) Pvt Ltd
2. Introduction to Parallel computing , W.P. Petersen and P.Arbenz, Oxford Univ. Press
3. Introduction to Parallel computing ,Ananth Grama, Anshul Gupta,G .Karypis, and V.Kumar, Pearson Education.

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### CLOUD NETWORKING

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

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| CO1 | **understand** the characteristics of cloud networking | K2 |
| CO2 | **Design** to assess efficiency of switch fabric technology | K5 |
| CO3 | **Implement** and know data centric networking standards | K4 |
| CO4 | **Implement** new network architecture and Clos | K4 |
| CO5 | **Ability to understand** Network Aggregation | K2 |

***#Based on suggested Revised BTL***

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 1 | 2 | 2 | | 1 | | 3 | | | | | |
| CO2 | 1 | 2 | 2 | | 3 | 2 | 1 | | | | | |
| CO3 | 2 | 1 | 2 | | 3 | 3 | | | | | | |
| CO4 | 1 | 2 | 2 | 2 | 3 | | | | | | | |
| CO5 | | 2 | 2 | | 1 | | 3 | | | | 1 | |

**(Please fill the above with Levels of Correlation, viz., L-1, M-2, H-3)**
**SYLLABUS:**

**UNIT-I:Welcome to Cloud Networking**: Introduction Networking basics what is a cloud data center? B What is cloud networking? Characteristics of cloud networking. **Data Center Evolution—Mainframes to the Cloud**: The data center evolution Computer networks, Ethernet, Enterprise versus cloud data centers.

**UNIT-II: Switch Fabric Technology**: Switch fabric architecture overview, Switch fabric topologies, Congestion management, Flow control, Traffic management, Switch chip architecture examples, **Cloud Data Center Networking Topologies:** Traditional multitiered enterprise networks**,** Data center network switch types**,** Flat data center networks**,** Rack scale architectures**,** Network function virtualization.

**UNIT-III: Data Center Networking Standards:** Ethernet data rate standards, Virtual local area networks, Data center bridging, Improving network bandwidth, Remote direct memory access.

**Server Virtualization and Networking:** VM overview, Virtual switching, PCI express, Edge virtual bridging, VM migration. **Network Virtualization:** Multi-tenant environments, Traditional network tunnelling protocols, VXLAN, NVGRE, Tunnel locations, Load Balancing.

**UNIT-IV: The Motivations for a New Network Architecture:** The Application-Network Shuffle, The Network Design from the Turn of the Century, The Charms of Bridging, Building Scalable Bridging Networks, The Trouble with the Access-Aggregation-Core Network Design, **Clos: Network Topology for a New World:**Introducing the Clos Topology, A Deeper Dive into the Clos Topology, Use of Homogeneous Equipment, Routing as the Fundamental Interconnect Model, Oversubscription in a Clos Topology, Interconnect Link Speeds, Scaling the Clos Topology, Comparing the Two Three-Tier Models, Application Matchup.

**UNIT-V: Network Disaggregation:** What Is Network Disaggregation?, Why Is Network Disaggregation Important? , Controlling Costs, Avoiding Vendor Lock-In, Standardization of Features, What Made Network Disaggregation Possible Now?, Difference in Network Operations with Disaggregation, Purchase and Support, First Boot, Open Network Installer Environment, How Does ONIE Work?**The Players in Network Disaggregation:** Hardware, Packet-Switching Silicon, ODMs, CPU Complex.

**TEXT BOOKS:**

1. Cloud Networking: Understanding Cloud-Based Data Center Networks- Gary Lee,Morgan Kaufmann.
2. Cloud Native Data Center Networking by Dinesh G Dutt, O'Reilly Media, Inc. by Dinesh G. Dutt

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### CLOUD COMPUTING

**Course Objectives:**

- Ability to understand various service delivery models of a cloud computing architecture.
- Ability to understand the ways in which the cloud can be programmed and deployed.
- Understanding cloud service providers.
- Understand and learn the Characteristics of IaaS and Characteristics of PaaS
- Learn and gain knowledge about the Cloud Service providers.

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

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Explain different types of Computing | **K3** |
| **CO2** | Illustrate Four types of Cloud Deployment Models | **K4** |
| **CO3** | Demonstrate different Phases of Cloud Migration Approaches for Cloud Migration | **K5** |
| **CO4** | Analyze and Develop Cloud Service Models | **K3** |
| **CO5** | Design applications for an organization which use cloud environment. | **K4** |

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 1 | 2 | 1 | | 2 | | | | 2 | | | |
| **CO2** | 2 | | 2 | 3 | | 2 | | | | 2 | 1 | |
| **CO3** | 1 | | 3 | | 1 | 3 | | | 3 | 2 | | |
| **CO4** | | 2 | 1 | 2 | | | | | | 2 | 1 | |
| **CO5** | 1 | 2 | | | 1 | | | | 1 | 3 | | |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**SYLLABUS:**

**UNIT-I: Computing Paradigms:** High-Performance Computing, Parallel Computing, Distributed Computing, Cluster Computing, Grid Computing, Cloud Computing, Quantum Computing, Optical Computing, Nano computing.

**UNIT-2: Cloud Computing Fundamentals:** Motivation for Cloud Computing, The Need for Cloud Computing, Definition of Cloud computing, Cloud Computing Is a Service, Principles of Cloud computing, Five Essential Characteristics, Four Cloud Deployment Models.

**UNIT-3: Cloud Computing Architecture and Management:** Cloud architecture, Layer, Anatomy of the Cloud, Network Connectivity in Cloud Computing, Applications, Managing the Cloud Infrastructure Managing the Cloud application, Migrating Application to Cloud, Phases of Cloud Migration Approaches for Cloud Migration.

**UNIT-4: Cloud Service Models:** Infrastructure as a Service, Characteristics of IaaS. Suitability of IaaS, Pros and Cons of IaaS, Summary of IaaS Providers, Platform as a Service, Characteristics of PaaS, Suitability of PaaS, Pros and Cons of PaaS, Summary of PaaS Providers, Software as a Service, Characteristics of SaaS, Suitability of SaaS, Pros and Cons of SaaS, Summary of SaaS Providers.

**UNIT-5**:Python for Amazon Web Services, Python for Google Cloud Platform, Python for Windows Azure, Python for MapReduce, Python Packages of Interest, Python Web Application Framework – Django, Designing a RESTful Web API.

**TEXT BOOKS:**

1. Essentials of cloud Computing: K. Chandrasekhran, CRC press, 2014
2. ArshadeepBhaga, Vijay Madisetti, “Cloud Computing AHandson Approach”, Universities Press, 2018.

**REFERENCE BOOKS:**

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

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### HIGH PERFORMANCE COMPUTING

**Course Objectives:** The objective of the subject is to

- Introduce the basic concepts related to HPC architecture and parallel computing.
- To discuss various computational techniques for studying soft matter systems.
- To apply these concepts to examine complex bimolecular/materials systems that generally require large-scale HPC platform with hybrid CPU-GPU architectures.

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

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | **Design,** formulate, solve and implement high performance versions of standard single threaded algorithms. | **K6** |
| **CO2** | **Demonstrate** the architectural features in the GPU and MIC hardware accelerators. | **K2** |
| **CO3** | **Design** programs to extract maximum performance in a multi-core, shared memory execution environment processor. | **K6** |
| **CO4** | **Analyze** Symmetric and Distributed architectures. | **K4** |
| **CO5** | **Develop** and deploy large scale parallel programs on tightly coupled parallel systems using the message passing paradigm. | **K6** |

**#based on suggested Revised BTL**

**Mapping of course outcomes with program outcomes:**

| CO | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 1 | 3 | 2 | | 1 | 1 | | | | | | |
| CO2 | | 2 | 3 | 1 | 1 | 2 | | | | | | |
| CO3 | | 1 | 3 | 2 | | 1 | 1 | | | | | |
| CO4 | 2 | 3 | 2 | | 1 | 1 | | | | | | |
| CO5 | | 1 | 3 | 2 | | 2 | 1 | | | | | |

**(Please fill the above with Levels of Correlation, viz., L, M, H)**

**SYLLABUS:**

**UNIT I: Graphics Processing Units:** Introduction to Heterogeneous Parallel Computing, GPU architecture, Thread hierarchy, GPU Memory Hierarchy.

**UNIT II: GPU Programming:** Vector Addition, Matrix Multiplication algorithms. 1D, 2D, and 3D Stencil Operations, Image Processing algorithms – Image Blur, Gray scaling. Histogramming, Convolution, Scan, Reduction techniques.

**UNIT III: Many Integrated Cores:** Introduction to Many Integrated Cores. MIC, Xeon Phi architecture, Thread hierarchy, Memory Hierarchy, Memory Bandwidth and performance considerations.

**UNIT IV: Shared Memory Parallel Programming:** Symmetric and Distributed architectures, OpenMP Introduction, Thread creation, Parallel regions. Work sharing, Synchronization.

**UNIT V: Message Passing Interface:** MPI Introduction, Collective communication, Data grouping for communication.

**TEXT BOOKS:**

1. Programming Massively Parallel Processors A Hands-on Approach, 3e, Wen-Mei W Hwu, David B Kirk and Morgan Kaufmann-2019
2. Intel Xeon Phi Coprocessor Architecture and Tools, Rezaur Rahman, Apress Open, 1st edition-2013
3. Using OpenMP, Barbara Chapman, Gabriele Jost, Rudd Vander Pas, MIT Press, 2008

**REFERENCE BOOKS:**

1. "A Parallel Algorithm Synthesis Procedure for High-Performance Computer Architectures" by Dunn Ian N, 2003

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### ADVANCED COMPUTER ARCHITECTURE

**Course Objectives:**

- The main objective of the course is to provide a deep understanding on advanced computer architectures and low-level system software such as pipelined and Multiprocessor systems

**Course Outcomes (COs)**: At the end of the course, student will be able to

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | **Understand** the advanced concepts of computer architecture. (Knowledge and understanding) | **K2** |
| **CO2** | **Compare** and contrast the parallel architectures | **K2, K4,K5** |
| **CO3** | **Illustrate** parallel programming concepts | **K2, K3** |
| **CO4** | **Investigate** modern design structures of Pipelined and Multiprocessors systems. | **K4** |
| **CO5** | **Acquainted** with recent computer architectures and I/O devices, as well as the low-level language required to drive/manage these types of advanced hardware. | **K6** |
| **CO6** | **Identify** the limitations of ILP. | **K2** |

***#Based on suggested Revised BTL***

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 1 | 2 | 3 | | 1 | 1 | | | | | | |
| **CO2** | 2 | 2 | 2 | | 1 | | 3 | | | | | |
| **CO3** | | 1 | 2 | | 3 | 1 | 1 | | | | | |
| **CO4** | 3 | 2 | 2 | 2 | | 1 | | 3 | | | | |
| **CO5** | 2 | 2 | | 1 | | 3 | | | | | | |

**(Please fill the above with Levels of Correlation, viz., L-1, M-2, H-3)**

**SYLLABUS:**

**UNIT–I: Theory of Parallelism**: Parallel Computer Models, The State of Computing, Multiprocessors and Multicomputer, Multi-vector and SIMD Computers, PRAM and VLSI Models, Program and Network Properties, Conditions of Parallelism, Program Partitioning and Scheduling, Program Flow Mechanisms, System Interconnect Architectures, Principles of Scalable Performance, Performance Metrics and Measures, Parallel Processing Applications, Speedup Performance Laws, Scalability Analysis and Approaches.

**UNIT-II: Hardware Technologies:** Processors and Memory Hierarchy, Advanced Processor Technology, Superscalar and Vector Processors, Memory Hierarchy Technology, Virtual Memory Technology.

**UNIT-III:** Bus, Cache, and Shared Memory, Bus Systems, Cache Memory Organizations, Shared Memory Organizations, Sequential and Weak Consistency

Models, Pipelining and Superscalar Techniques, Linear Pipeline Processors, Nonlinear Pipeline Processors, Instruction Pipeline Design, Arithmetic Pipeline Design.

**UNIT-IV: Parallel and Scalable Architectures:** Multiprocessors and Multicomputers, Multiprocessor System Interconnects, Cache Coherence and Synchronization Mechanisms, Three Generations of Multicomputers, Message-Passing Mechanisms, Multivector and SIMD Computers, Vector Processing Principles, Multivector Multiprocessors, Compound Vector Processing, SIMD Computer Organizations(Upto 8.4), Scalable, Multithreaded and Dataflow Architectures, Latency-Hiding Techniques, Principles of Multithreading, Fine-Grain Multicomputers, Scalable and Multithreaded Architectures, Dataflow and Hybrid Architectures.

**UNIT–V: Software for parallel programming**: Parallel Models, Languages, and Compilers, Parallel Programming Models, Parallel Languages and Compilers, Dependence Analysis of Data Arrays, Parallel Program Development and Environments, Synchronization and Multiprocessing Modes. Instruction and System Level Parallelism, Instruction Level Parallelism, Computer Architecture, Contents, Basic Design Issues, Problem Definition, Model of a Typical Processor, Compiler-detected Instruction Level Parallelism.

**TEXT BOOKS:**

1. "Advanced Computer Architecture (SIE): Parallelism, Scalability, Programmability", Kai Hwang and Naresh Jotwani, McGraw Hill Education 3rd Edition. 2015
2. "Computer Architecture and Parallel Processing", Kai Hwang and Faye Briggs, Mc Graw-Hill International Edition, 2000

**REFERENCE BOOKS:**

1. John L. Hennessy and David A. Patterson, Computer Architecture: A quantitative approach, 5th edition, Morgan Kaufmann Elseveir, 2013
2. Introduction to High Performance Computing for Scientists and Engineers, G. Hager and G. Wellein, CRC Press, Taylor & Francis Group.
3. Sima D, Fountain T and Kacsuk P, "Advanced Computer Architectures: A Design Space Approach", Addison Wesley, 2000.

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**TRACK- 4 Computational Intelligence**

### ARTIFICAL INTELLIGENCE

**Course Objectives:** The student will be able to

- Know the methodology of Problem solving
- Implement basic AI algorithms
- Design and carry out an empirical evolution of different algorithms on a problem formalization

**Course Outcomes (COs)**: At the end of the course, student will be able to

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | **Understand** the fundamental concepts in Artificial Intelligence | **K2** |
| **CO2** | **Analyze** the applications of search strategies and problem reductions | **K2, K4,K5** |
| **CO3** | **Apply** the mathematical logic concepts. | **K2, K3** |
| **CO4** | **Develop** the Knowledge representations in Artificial Intelligence. | **K4** |
| **CO5** | **Design** the Fuzzy logic based expert systems. | **K6** |

***#Based on suggested Revised BTL***

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | | 3 | 3 | | 2 | | | 1 | | | | |
| **CO2** | 2 | 3 | 2 | | 1 | | | | | | | |
| **CO3** | | 2 | | 1 | 1 | 2 | | 3 | | | | |
| **CO4** | 2 | 3 | 2 | | 1 | | | | | | | |
| **CO5** | | 2 | | 1 | 1 | 2 | | 3 | | | | |

**(Please fill the above with Levels of Correlation, viz., L, M, H)**

**SYLLABUS:**

**UNIT I: Introduction to artificial intelligence:** Introduction ,history, intelligent systems, foundations of AI, applications, tic-tac-tie game playing, development of AI languages, current trends in AI **Problem solving: state-space search and control strategies:** Introduction, general problem solving, characteristics of problem

**UNIT II: Search Strategies:** exhaustive searches, heuristic search techniques, iterative-deepening A*, constraint satisfaction **Problem reduction and game playing:** Introduction, problem reduction, game playing, alpha-beta pruning, two-player perfect information games

**UNIT III: Logic concepts:** Introduction, propositional calculus, proportional logic, natural deduction system, axiomatic system, semantic tableau system in proportional logic, resolution refutation in proportional logic, predicate logic

**UNIT IV: Knowledge representation:** Introduction, approaches to knowledge representation, knowledge representation using semantic network, extended semantic networks for KR, knowledge representation using frames **advanced knowledge representation techniques:** Introduction, conceptual dependency theory, script structure.

**UNIT V: Expert system and applications:** Introduction phases in building expert systems, expert system versus traditional systems, rule-based expert systems blackboard systems truth maintenance systems, application of expert systems, list of shells and tools. **Fuzzy sets and fuzzy logic:** Introduction, fuzzy sets, fuzzy set operations, types of membership functions, multi valued logic, fuzzy logic, linguistic variables and hedges, fuzzy propositions, inference rules for fuzzy propositions, fuzzy systems.

**TEXT BOOKS:**

1. Artificial Intelligence- Saroj Kaushik, CENGAGE Learning,
2. Artificial intelligence, A modern Approach , 2nd ed, Stuart Russel, Peter Norvig, PEA
3. Artificial Intelligence- Rich, Kevin Knight, Shiv Shankar B Nair, 3rd ed, TMH
4. Introduction to Artificial Intelligence, Patterson, PHI

**REFERENCE BOOKS:**

1. Atificial intelligence, structures and Strategies for Complex problem solving, - George F Lugar, 5th ed, PEA
2. Introduction to Artificial Intelligence, Ertel, Wolf Gang, Springer
3. Artificial Intelligence, A new Synthesis, Nils J Nilsson, Elsevier

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### MACHINE LEARNING

**Course Objectives:**

- Identify problems that are amenable to solution by AI methods, and which AI methods may be suited to solving a given problem.
- Formalize a given problem in the language/framework of different AI methods (e.g., as a search problem, as a constraint satisfaction problem, as a planning problem, as a Markov decision process, etc).

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

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| CO1 | **Explain** the definition and usage of the term 'the internet of things' in different contexts. | K2 |
| CO2 | **Demonstrate** on various network protocols used in IoT. | K2 |
| CO3 | **Analyze** on various key wireless technologies used in IoT systems, such as WiFi, 6LoWPAN, Bluetooth and ZigBee. | K4 |
| CO4 | **Illustrate** on the role of big data, cloud computing and data analytics in IoT system. | K5 |
| CO5 | **Design** a simple IoT system made up of sensors, wireless network connection, data analytics and display/actuators, and write the necessary control software. | K6 |

**#based on suggested Revised BTL**

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | | 3 | 3 | | 2 | | | 1 | | | | |
| CO2 | 2 | 3 | 2 | | 1 | | | | | | | |
| CO3 | | 2 | | 1 | 1 | 2 | | 3 | | | | |
| CO4 | 2 | 3 | 2 | | 1 | | | | | | | |
| CO5 | | 2 | | 1 | 1 | 2 | | 3 | | | | |

**(Please fill the above with Levels of Correlation, viz., L-1, M-2, H-3)**

**SYLLABUS:**

**UNIT I: Introduction:** Towards Intelligent Machines Well posed Problems, Example of Applications in diverse fields, Data Representation, Domain Knowledge for Productive use of Machine Learning, Diversity of Data: Structured / Unstructured, Forms of Learning, Machine Learning and Data Mining, Basic Linear Algebra in Machine Learning Techniques.

**UNIT II: Supervised Learning:** Rationale and Basics: Learning from Observations, Bias and Why Learning Works: Computational Learning Theory, Occam's Razor

Principle and Over fitting Avoidance Heuristic Search in inductive Learning, Estimating Generalization Errors, Metrics for assessing regression, Metris for assessing classification.

**UNIT III: Statistical Learning:** Machine Learning and Inferential Statistical Analysis, Descriptive Statistics in learning techniques, Bayesian Reasoning: A probabilistic approach to inference, K-Nearest Neighbor Classifier. Discriminant functions and regression functions, Linear Regression with Least Square Error Criterion, Logistic Regression for Classification Tasks, Fisher's Linear Discriminant and Thresholding for Classification, Minimum Description Length Principle.

**UNIT IV: Support Vector Machines (SVM):** Introduction, Linear Discriminant Functions for Binary Classification, Perceptron Algorithm, Large Margin Classifier for linearly seperable data, Linear Soft Margin Classifier for Overlapping Classes, Kernel Induced Feature Spaces, Nonlinear Classifier, and Regression by Support vector Machines. **Learning with Neural Networks:** Towards Cognitive Machine, Neuron Models, Network Architectures, Perceptrons, Linear neuron and the Widrow-Hoff Learning Rule, The error correction delta rule.

**UNIT V:** Multilayer Perceptron Networks and error back propagation algorithm, Radial Basis Functions Networks. **Decision Tree Learning**: Introduction, Example of classification decision tree, measures of impurity for evaluating splits in decision trees, ID3, C4.5, and CART decision trees, pruning the tree, strengths and weakness of decision tree approach.

**TEXTBOOKS:**

1. Applied Machine Learning,1st edition, M.Gopal, McGraw Hill Education,2018
2. Machine Learning: An Algorithmic Perspective, Stephen Marsland, Taylor & Francis (CRC) 1st Edition-2014

**REFERENCE BOOKS:**

1. Machine Learning Methods in the Environmental Sciences, Neural Networks, William WHsieh, Cambridge Univ Press. *1* edition (August 31, 2009)
2. Richard o. Duda, Peter E. Hart and David G. Stork, pattern classification, John Wiley &SonsInc., 2nd Edition-2001
3. Chris Bishop, Neural Networks for Pattern Recognition, Oxford University Press, 1995.
4. Machine Learning by Peter Flach , Cambridge-1st Edition 2012

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### NATURAL LANGUAGE PROCESSING

**Course Objectives:**

- This course introduces the fundamental concepts and techniques of natural language processing (NLP).
- Students will gain an in-depth understanding of the computational properties of natural languages and the commonly used algorithms for processing linguistic information.
- The course examines NLP models and algorithms using both the traditional symbolic and the more recent statistical approaches.
- Enable students to be capable to describe the application based on natural language processing and to show the points of syntactic, semantic and pragmatic processing.

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

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | **Demonstrate** a given text with basic Language features. | **K2** |
| **CO2** | **Explain** a rule based system to tackle morphology/syntax of a language. | **K5** |
| **CO3** | **Design** an innovative application using NLP components. | **K6** |
| **CO4** | **Design** a tag set to be used for statistical processing for real-time applications. | **K6** |
| **CO5** | **Compare and contrast** the use of different statistical approaches for different types of NLP applications. | **K2** |

***#Based on suggested Revised BTL***

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **CO1** | 1 | 2 | 3 | | 2 | | | 1 | | | | |
| **CO2** | 2 | 3 | 2 | | 1 | | | | | | | |
| **CO3** | 1 | 2 | 3 | 1 | 1 | 2 | | 3 | | | | |
| **CO4** | 2 | 3 | 2 | | 1 | | | | | | | |
| **CO5** | 1 | 2 | | 1 | 1 | 2 | | 3 | | | | |

**(Please fill the above with Levels of Correlation, viz., L-1, M-2, H-3)**

**SYLLABUS:**

**UNIT-I:Introduction:** What is Natural Language Processing (NLP), Origins of NLP, Language and Knowledge, The challenges of NLP, Language and Grammar, Processing Indian Languages, NLP Applications, Some successful Early NLP Systems, Information Retrieval, **Language Modelling:** Introduction, Various Grammar-based Language Models, Statistical Language Model.

**UNIT-II: Word Level Analysis:** Introduction, Regular Expressions, Finite State Automata, Morphological Parsing, Spelling Error Detection and Correction, Words and Word Classes, Part-of-Speech Tagging, **Syntactic Analysis:** Introduction, Context-Free Grammar, Constituency, Parsing, Probabilistic Parsing, Indian Languages.

**UNIT-III: Semantic Analysis and Pragmatics:** Introduction, Meaning Representation, Lexical Semantics, Ambiguity, Word Sense Disambiguation, **Discourse Processing:** Introduction, Cohesion, Reference Resolution, Discourse Coherence and Structure.

**UNIT-IV: Natural Language Generation:** Introduction, Architectures of NLG Systems, Generation task and Representations, Applications of NLG, **Machine Translation:** Introduction, Problems in Machine Translation, Characteristics of Indian Languages, Machine Translation Approaches, Direct Machine Translation, Rule-based Machine Translation, Corpus-based Machine Translation, Semantic or Knowledge-based MT Systems, Translation involving Indian Languages.

**UNIT-V: NLP Applications:** Introduction, Information Extraction, Automatic Text Summarization, Question-Answering System, **Lexical Resources:** Introduction, WordNet, FrameNet, Stemmers, Part-of-Speech Tagger, Research Corpora, Journals and Conferences in the Area.

**TEXT BOOKS:**

1. TANVEER Siddiqui, U.S. Tiwary, ―Natural Language Processing and Information Retrieval, Oxford University Press, 2008.
2. Daniel Jurafsky, James H. Martin―Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech, Pearson Publication, 2014.

**REFERENCE BOOKS:**

1. Steven Bird, Ewan Klein and Edward Loper, ―Natural Language Processing with Python, First Edition, OReilly Media, 2009.
2. Breck Baldwin, ―Language processing with Java and Ling Pipe Cookbook, Atlantic Publisher, 2015.
3. Richard M Reese, ―Natural Language Processing with Java, OReilly Media, 2015.

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### NEURAL NETWORKS

**Course Objectives:**

- The main objective of Neural Network Techniques to Improve Data Analysis Solutions is to strengthen the dialogue between the statistics and soft computing research communities in order to cross-pollinate both fields and generate mutual improvement activities.
- Also introduce the neural networks for classification, regression and to give design methodologies for artificial neural networks.

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

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | **Explain** Fundamentals of Neural Networks | **K2,K5** |
| **CO2** | **Explain** Architecture of Back-propagation(BP) Networks | **K5** |
| **CO3** | **Design** Associative Memory for Real coded pattern pairs | **K6** |
| **CO4** | **Design** ART1 and ART2 Architecture and algorithms | **K6** |
| **CO5** | **Compare** Hybrid system and Neural Networks | **K2** |

***#Based on suggested Revised BTL***

**Mapping of course outcomes with program outcomes**

| CO | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 3 | 1 | | 2 | | | | | | | | |
| CO2 | 1 | 3 | 2 | | | | | | | 1 | | |
| CO3 | 1 | 2 | 3 | | 1 | | | | | | | |
| CO4 | | 2 | 3 | | 1 | | | 1 | | | | |
| CO5 | 1 | | | 1 | | | | | | | 1 | |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**SYLLABUS:**

**UNIT-I: Introduction:** Neural Network, Fuzzy logic, Genetic Algorithm. **Fundamentals of Neural Networks**: What is Neural Network, Model of Artificial Neuron, Learning rules and various activation functions.

**UNIT-II: Neural Network Architecture:** Single layer Feed-forward networks. Multilayer Feed-forward networks. Recurrent Network , **Back propagation Networks:** Back Propagation networks, Architecture of Back-propagation(BP) Networks, Back-propagation Learning, Variation of Standard Back propagation algorithms.

**Department of Computer Science & Engineering**
**University College of Engineering, JNT University Kakinada**

**UNIT-III: Associative Memory:** Auto correlators, Heterocorrelators, Wang et al's Multiple Training Encoding Strategy, Exponential BAM, Associative Memory for Real coded pattern pairs, Applications.

**UNIT-IV: Adaptive Resonance Theory:** Cluster Structure, Vector Quantization, Classical ART Network, Simplified ART Architecture,ART1 and ART2 Architecture and algorithms, Applications, Sensitivities of ordering of data.

**UNIT-V: Introduction about Fuzzy set theory:** Fuzzy versus Crisp, Crisp and fuzzy sets, Crisp and Fuzzy relations, **Integration of Neural Network, Fuzzy logic and Genetic Algorithm:** Hybrid system. Neural Networks, Fuzzy logic, and Genetic Algorithm Hybrids.

**TEXT BOOKS:**

1. Bishop, C. M. Neural Networks for Pattern Recognition. Oxford University Press. 1995.
2. Neural Networks, Fuzzy Logic and Genetic Algorithms, by S.Rajasekaran and G.A. Vijayalakshmi Pai.

**REFERENCES:**

1. Neuro-Fuzzy Systems, Chin Teng Lin, C. S. George Lee, PHI.
2. Build_ Neural_ Network_ With_ MS_ Excel_ sample by Joe choong

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### MULTI AGENT SYSTEMS

**Course Outcomes:**
At the end of the module, the student will be able to demonstrate:

- Understand the notion of an agent, how agents are distinct from other software paradigms (eg objects) and understand the characteristics of applications that lend themselves to an agent-oriented solution;
- Understand the key issues associated with constructing agents capable of intelligent autonomous action, and the main approaches taken to developing such agents;
- Understand the key issues in designing societies of agents that can effectively cooperate in order to solve problems, including an understanding of the key types of multi-agent interactions possible in such systems
- Understand the main application areas of agent-based solutions, and be able to develop a meaningful agent-based system using a contemporary agent development platform.

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

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| CO1 | **how** agents are distinct from other software paradigms (eg objects) and understand the characteristics of applications | K1 |
| CO2 | **Identify** different types of multi-agent interactions possible in such systems | K3 |
| CO3 | **Develop** a meaningful agent-based system using a contemporary agent development platform. | K6 |
| CO4 | **Compare** and contrast between benevolent agents | K5 |
| CO5 | **Discuss** about FIPA framework. | K6 |

***#Based on suggested Revised BTL***

**Mapping of course outcomes with program outcomes**

| CO | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 1 | 3 | 2 | | 1 | 1 | | | | | | |
| CO2 | 1 | 3 | 3 | | 2 | | | | | | | |
| CO3 | 1 | 3 | 3 | 2 | 1 | | | | | | 1 | |
| CO4 | | 2 | 1 | | | 3 | | 2 | | 1 | | |
| CO5 | 2 | 1 | 3 | | | | 1 | 1 | | | | |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**Department of Computer Science & Engineering**
**University College of Engineering, JNT University Kakinada**

**SYLLABUS:**

**UNIT-1: Introduction:** what is an agent?, agents and objects; agents and expert systems; agents and distributed systems; typical application areas for agent systems.

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

**UNIT-III: Multi-Agent Systems:** Classifying multi-agent interactions - cooperative versus non-cooperative; zero-sum and other interactions; what is cooperation? how cooperation occurs - the Prisoner's dilemma and Axelrod's experiments;

**UNIT-IV: Interactions between self-interested agents:** auctions & voting systems: negotiation; Interactions between benevolent agents: cooperative distributed problem solving (CDPS), partial global planning; coherence and coordination;

**UNIT-V: Interaction languages and protocols:** speech acts, KQML/KIF, the FIPA framework. Advanced topics: One issue selected from the contemporary research literature, perhaps by guest lecturer.

**TEXT BOOK:**

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

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### DEEP LEARNING

**Course Objectives:** At the end of the course, the students will be expected to:

- Learn deep learning methods for working with sequential data,
- Learn deep recurrent and memory networks,
- Learn deep Turing machines,
- Apply such deep learning mechanisms to various learning problems.
- Know the open issues in deep learning, and have a grasp of the current research directions.

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

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| CO1 | **Demonstrate** the basic concepts fundamental learning techniques and layers. | K2 |
| CO2 | **Discuss** the Neural Network training, various random models. | K6 |
| CO3 | **Explain** different types of deep learning network models. | K5 |
| CO4 | **Classify** the Probabilistic Neural Networks. | K2 |
| CO5 | **Implement** tools on Deep Learning techniques. | K3 |

**#based on suggested Revised BTL**

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 1 | 2 | 1 | 1 | 2 | | | | 2 | | | |
| CO2 | 2 | 1 | 3 | 2 | 3 | 1 | | | | | 1 | |
| CO3 | 1 | | 3 | | 1 | 3 | | | 3 | 2 | | |
| CO4 | | 2 | | 2 | 3 | 1 | | | | | 1 | |
| CO5 | 1 | 2 | | 3 | 1 | | | | 2 | 1 | | |

**(Please fill the above with Levels of Correlation, viz., L-1, M-2, H-3)**

**SYLLABUS:**

**UNIT 1: Foundations of Neural Networks and Deep Learning -** Neural Networks: Biological Neuron, Perceptron, Multi Layer Perceptron. Training Neural Networks: Back propagation, Activation Functions, Loss Function, Hyper parameters.

**UNIT-2: Fundamentals of Deep Learning**-Definition of Deep Learning, Common Architecture Principles of Deep Networks, Building Blocks of Deep Learning. Architectures of Deep Learning: Unsupervised Pre trained Networks, Convolution Neural Networks(CNN’s), Recurrent Neural Networks, Recursive Neural Networks,

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**UNIT-3: Deep Learning Research-** Linear factor models: Probabilistic PCA And Factor Analysis, Independent Component Analysis, Sparse Coding, Manifold Interpretation of PCA, Auto Encoders: Regularized Auto encoders, Representational Power, Layer Size and Depth, Denoising Auto encoders, Applications of Auto encoders.

**UNIT-4: Deep Generating Models-** Boltzmann Machines, Restricted Boltzmann Machines, Deep Belief Networks, Deep Boltzmann Machines, Convolution Boltzmann Machines, Back propagation through Random Operations, Directed Generative Nets, Generating Static Networks.

**UNIT-5: Applications-** Large Scale Deep Learning, Image Recognition, Speech Recognition, Natural Language Processing, Other Applications. Building Deep Networks: DL4J suite of Tools, Basic Concepts of DL4J API, Modeling CSV Data with Multilayer Perceptron Networks, Modeling Handwritten Images Using CNN's

**TEXT BOOKS:**

1. Deep Learning A practitioner's approach- josh Patterson and Adam Gibson, OREILLY.
2. Goodfellow, I., Bengio,Y., and Courville, A., Deep Learning, MIT Press, 2016.

**REFERENCE BOOKS:**

1. Artificial Neural Networks, Yegnanarayana, B., PHI Learning Pvt. Ltd, 2009.
2. Matrix Computations, Golub, G.,H., and Van Loan,C.,F, JHU Press,2013.
3. Neural Networks: A Classroom Approach, Satish Kumar, Tata McGraw-Hill Education, 2004.
4. Bishop, C. ,M., Pattern Recognition and Machine Learning, Springer, 2006.

# Other Branches Syllabus R20

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### EEE Branch
### R20 Regulations
### Python Programming

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

| Cos | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Understand and comprehend the basics of python programming. Demonstrate the principles of structured programming and be able to describe, design, implement, and test structured programs using currently accepted methodology | K2 |
| **CO2** | Demonstrate the principles of structured programming and be able to describe, design, implement, and test structured programs using currently accepted methodology. Explain the use of the built-in data structures list, sets, tuples and dictionary | K3 |
| **CO3** | Understand of functions and its applications, Modules. | K3 |
| **CO4** | Identify real-world applications using oops, files and exception handling provided by python. | K3 |
| **CO5** | Formulate and implement a program to solve a real-world problem using GUI and Turtle graphics. | K3 |

**UNIT – 1 Introduction and Strings:** History of Python, Need of Python Programming, Applications Basics of Python Programming Python Basics, Objects- Python Objects, Standard Types, Other Built-in Types, Internal Types, Standard Type Operators, Standard Type Built-in Functions, Categorizing the Standard Types, Unsupported Types Numbers - Introduction to Numbers, Integers, Floating Point Real Numbers, Complex Numbers, Operators, Built-in Functions, Related Modules Sequences - Strings, Lists, and Tuples, Mapping and Set Types. Introduction to Numbers, Integers, Floating Point Real Numbers, Complex Numbers, Operators, Built-in Functions, Related Modules Sequences - Strings, Lists, and Tuples, Mapping and Set Types.

**UNIT – 2: FILES**: File Objects, File Built-in Function [ open() ], File Built-in Methods, File Built-in Attributes, Standard Files, Command-line Arguments, File System, File Execution, Persistent Storage Modules.

**Errors and Exceptions**: Difference between an error and Exception, Handling Exception, try except block, Raising Exceptions, User Defined Exceptions Detecting and Handling Exceptions, Context Management, **Modules**: Modules and Files, Namespaces, Importing Modules, Importing Module Attributes, Module Built-in Functions, Packages, Other Features of Modules

**UNIT – 3: OOP in Python and Regular Expressions:** Classes and objects, Inheritance, Overriding Methods, Data-hiding, Polymorphism, introduction to regular expressions Introduction, Special Symbols and Characters, Res and

**Department of Computer Science & Engineering**
**University College of Engineering, JNT University Kakinada**

**Python Multithreaded Programming:** Introduction, Threads and Processes, Python, Threads, and the Global Interpreter Lock, Thread Module, Threading Module.

**UNIT – 4: GUI Programming and Testing:** Introduction, Tkinter and Python Programming, Brief Tour of Other GUIs, Related Modules and Other GUIs WEB Programming: Introduction, Wed Surfing with Python, Creating Simple Web Clients, Advanced Web Clients, CGI-Helping Servers Process Client Data, Building CGI Application Advanced CGI, Web (HTTP) Servers, **Testing:** Why testing is required? Basic concepts of testing, Unit testing in Python, Writing Test cases, Running Tests.

**UNIT – 5 Database Programming:** Introduction, Python Database Application Programmer's Interface (DB-API), Object Relational Managers (ORMs), Related Modules

**TEXT BOOKS:**

1. Core Python Programming, W.Chun, Pearson.
2. Fundamentals of Python First Programs, Kenneth. A. Lambert, Cengage.

**REFERENCE BOOKS:**

1. Python Programming: A Modern Approach, Vamsi Kurama, Pearson.
2. Think Python, Allen Downey, Green Tea Press
3. Introduction to Python Programming, Gowrishankar.S, Veena A, CRC Press.

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**EEE Branch & MECH Branch**
**R20 Regulations**
**Python Programming Lab**

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

| Cos | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Understand the Numbers, Math functions, Strings, List, Tuples andDictionaries in Python | K2 |
| **CO2** | Demonstrate different Decision-Making statements and Functions. | K4 |
| **CO3** | Interpret Object oriented programming in Python. | K3 |
| **CO4** | Demonstrate and summarize different File handling operations. | K4 |
| **CO5** | Identify how to design GUI Applications in Python and evaluate different database operations | K3 |
| **CO6** | Design and develop Client Server network applications using Python. | K4 |

***#Based on suggested Revised BTL***

**Exercise- 1**:

a. Write a program to demonstrate different number data types in Python.
b. Write a program to perform different Arithmetic Operations on numbers in Python
c. Write a program to create, concatenate and print a string and accessing sub-string from a given string.

**Exercise- 2**:

a. Write a python script to print the current date in the following format "WED 09 02:26:23 IST 2020"
b. Write a Python program to convert temperatures to and from Celsius, Fahrenheit [Formula: c/5 = f-32/9]
c. Write a Python script that prints prime numbers less than 20.

**Exercise- 3**:

a. Write a python program to find factorial of a number using Recursion.
b. Write a program that accepts the lengths of three sides of a triangle as inputs. The program output should indicate whether or not the triangle is a right triangle (Recall from the Pythagorean Theorem that in a right triangle, the square of one side equals the sum of the squares of the other two sides).
c. Write a python program to define a module to find Fibonacci Numbers and import the module to another program.

**Exercise- 4**:

a. Write a script named copyfile.py. This script should prompt the user for the names of two text files. The contents of the first file should be input and written to the second file.
b. Write a program that inputs a text file. The program should print all of the unique words in the file in alphabetical order.
c. Write a Python class to convert an integer to a roman numeral.

**Exercise- 5:**

a. Write a Python class to implement pow(x, n)
b. Write a Python class to reverse a string word by word.

**Exercise- 6:**

a. Write a program to count the numbers of characters in the string and store them in a dictionary data structure
b. Write a program to use split and join methods in the string and trace a birthday with a dictionary data structure.

**Exercise- 7** :

a. Write a program combine lists that combines these lists into a dictionary.
**b.** Write a program to count frequency of characters in a given file. Can you use character frequency to tell whether the given file is a Python program file, C program file or a text file?

**Exercise- 8:**

a. Write a function ball collide that takes two balls as parameters and computes if they are colliding. Your function should return a Boolean representing whether or not the balls are colliding.

   **Hint:** Represent a ball on a plane as a tuple of (x, y, r), r being the radius If (distance between two balls centers) <= (sum of their radii) then (they are colliding)

b. Find mean, median, mode for the given set of numbers in a list

**Exercise- 9:**

a. Write a function nearly equal to test whether two strings are nearly equal. Two strings a and b are nearly equal when a can be generated by a single mutation on b.
b. Write a function dups to find all duplicates in the list.
c. Write a function unique to find all the unique elements of a list.

Department of Computer Science & Engineering
University College of Engineering, JNT University Kakinada

**Exercise- 10:**

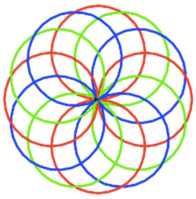
a. Write a program that defines a matrix and prints
b. Write a program to perform addition of two square matrices
c. Write a program to perform multiplication of two square matrices

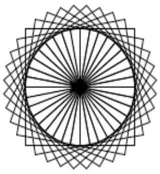
**Exercise- 11:**

a. Install packages requests, flask and explore them. using (pip)
b. Write a script that imports requests and fetch content from the page. Eg. (Wiki)
c. Write a simple script that serves a simple HTTP Response and a simple HTML Page

**Exercise- 12:**

a. Write a GUI for an Expression Calculator using tk .
b. Write a program to implement the following figures using turtle





**Exercise- 13:**

a. Write a test-case to check the function even numbers which return True on passing a list of all even numbers
b. Write a test-case to check the function reverse string which returns the reversed string

**Exercise- 14:**

a. In algebraic expressions, the symbol for multiplication is often left out, as in 3x+4y or 3(x+5). Computers prefer those expressions to include the multiplication symbol, like 3*x+4*y or 3*(x+5). Write a program that asks the user for an algebraic expression and then inserts multiplication symbols where appropriate.
b. Write a program that asks the user to enter a length in feet. The program should then give the user the option to convert from feet into inches, yards, miles, millimeters, centimeters, meters, or kilometers. Say if the user enters a 1, then the program converts to inches, if they enter a 2, then the program converts to yards, etc. While this can be done with if statements, it is much shorter with lists and it is also easier to add new conversions if you use lists.

**Exercise- 15:**

a. Write a program that reads a file consisting of email addresses, each on its own line. Your program should print out a string consisting of those email addresses separated by semicolons.

b. Write a program that reads a list of temperatures from a file called temps.txt, converts those temperatures to Fahrenheit, and writes the results to a file called ftemps.txt.

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

## R20 Regulations
**PE &PCE DEPARTMENT**

### SOFT COMPUTING TECHNIQUES

**Course Objectives:** In the course the student will Learn soft computing concepts and techniques and foster their abilities in designing and implementing soft computing based solutions for real-world problems.

**Course Outcomes (COs)**: At the end of the course, student will be able to

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| **CO1** | Able to apply fuzzy logic and reasoning to handle uncertainty in engineering problems. | **K2** |
| **CO2** | Make use of genetic algorithms to combinatorial optimization problems | **K3** |
| **CO3** | Apply artificial intelligence techniques, including search heuristics, knowledge representation, planning and reasoning. | **K5** |
| **CO4** | Learn and apply the principles of self adopting and self organizing neuro fuzzy inference systems | **K4** |
| **CO5** | Evaluate and compare solutions by various soft computing approaches for a given problem | **K3** |

#***Based on suggested Revised BTL***

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 2 | | 3 | 2 | | | | | 2 | | 2 | 1 |
| CO2 | 1 | 3 | | 1 | 1 | 2 | | | | 2 | 1 | |
| CO3 | 1 | 2 | 2 | 2 | 2 | 3 | | | 2 | 1 | 3 | |
| CO4 | 1 | 1 | 3 | 3 | 2 | 2 | | | | 2 | | 1 |
| CO5 | 2 | 2 | | 2 | | 1 | | | 3 | 1 | 2 | |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**SYLLABUS:**

**UNIT-I : FUZZY SET THEORY:** Introduction to Neuro – Fuzzy and Soft Computing, Fuzzy Sets, Basic Definition and Terminology, Set-theoretic Operations, Member Function Formulation and Parameterization, Fuzzy Rules and Fuzzy Reasoning, Extension Principle and Fuzzy Relations. Fuzzy Inference Systems, Mamdani Fuzzy Models, Sugeno Fuzzy Models.

**UNIT-II: OPTIMIZATION:** Derivative based Optimization, Descent Methods, The Method of Steepest Descent, Classical Newton's Method, Step Size Determination, Derivative-free Optimization, Genetic Algorithms.

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

**UNIT-III : ARTIFICIAL INTELLIGENCE :** Introduction, Knowledge Representation, Reasoning, Issues and Acquisition: Prepositional and Predicate Calculus Rule Based knowledge Representation Symbolic Reasoning, Heuristic Search: Techniques for Heuristic search Heuristic Classification.

**UNIT-IV: NEURO FUZZY MODELING:** Adaptive Neuro-Fuzzy Inference Systems, Architecture – Hybrid Learning Algorithm, Learning Methods that Cross-fertilize ANFIS and RBFN –Framework Neuron Functions for Adaptive Networks – Neuro Fuzzy Spectrum.

**UNIT-V: APPLICATIONS OF COMPUTATIONAL INTELLIGENCE :** Printed Character Recognition, Inverse Kinematics Problems, Automobile Fuel Efficiency Prediction, Soft Computing for Color Recipe Prediction.

**TEXT BOOKS:**

1. J.S.R.Jang, C.T.Sun and E.Mizutani, “Neuro-Fuzzy and Soft Computing”, PHI, 2004, Pearson Education 2004
2. N.P.Padhy, “Artificial Intelligence and Intelligent Systems”, Oxford University Press, 2006.

**REFERENCES:**

1. Elaine Rich & Kevin Knight, Artificial Intelligence, Second Edition, Tata Mcgraw Hill Publishing Comp., 2006, New Delhi.
2. Timothy J.Ross, “Fuzzy Logic with Engineering Applications”, McGraw-Hill, 1997.
3. Davis E.Goldberg, “Genetic Algorithms: Search, Optimization and Machine Learning”, Addison Wesley, N.Y., 1989.
4. S. Rajasekaran and G.A.V.Pai, “Neural Networks, Fuzzy Logic and Genetic Algorithms”, PHI,
5. R.Eberhart, P.Simpson and R.Dobbins, “Computational Intelligence - PC Tools”, AP Professional, Boston, 1996.
6. Amit Konar, “Artificial Intelligence and Soft Computing Behaviour and Cognitive model of the human brain”, CRC Press, 2008

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

## R20 Regulations
### PE &PCE DEPARTMENT
### DATA SCIENCE

**Course Objectives:**
From the course the student will learn

- Provide you with the knowledge and expertise to become a proficient data scientist.
- Demonstrate an understanding of statistics and machine learning concepts that are vital for data science;
- Produce Python code to statistically analyse a dataset;
- Critically evaluate data visualisations based on their design and use for communicating stories from data

**Course Outcomes:**

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| CO1 | Acquire the knowledge and expertise to become a proficient data scientist | K3 |
| CO2 | Demonstrate an understanding of statistics and machine learning concepts that are vital for data science | K3 |
| CO3 | Explain how data is collected, managed and stored for data science | K2 |
| CO4 | Interpret the key concepts in data science, including their real-world applications and the toolkit used by data scientists | K2 |
| CO5 | Illustrate data collection and management scripts using MongoDB | K3 |

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 1 | 2 | 3 | 2 | | | | | 2 | | 2 | 1 |
| CO2 | 2 | 2 | | 1 | 1 | 2 | | | | 2 | 1 | |
| CO3 | 1 | 3 | 2 | | | | | | 2 | | 3 | |
| CO4 | 1 | 2 | | 3 | | 2 | | | | 2 | | 1 |
| CO5 | 2 | 3 | 1 | 2 | | 1 | | | 3 | 1 | 2 | |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**SYLLABUS:**

**UNIT-I: Introduction to Core Concepts and Technologies-** Introduction, Terminology, data science process, data science toolkit, Types of data, Example applications.

**UNIT-II: Data Collection and Management-** Introduction, Sources of data, Data collection and APIs, Exploring and fixing data, Data storage and management, using multiple data sources.

**UNIT III: Data Analysis-** Introduction, Terminology and concepts, Introduction to statistics, Variance, Distribution properties and arithmetic, Basic machine learning algorithms, Linear regression, SVM, Naive Bayes.

**UNIT-IV: Data Visualisation-** Introduction, Types of data visualisation, **Data for visualisation-** Data types, Data encodings, Retinal variables, Mapping variables to encodings, Visual encodings.

**UNIT-V: Applications of Data Science-** Technologies for visualisation, Bokeh (Python), recent trends in various data collection and analysis techniques, various visualization techniques, application development methods of used in data science.

**TEXT BOOKS:**

1. "The Art of Data Science", 1st edition, Roger D. Peng and Elizabeth matsui, Lean Publications, 2015
2. "Algorithms for Data Science", 1st edition, **Steele**, Brian, **Chandler**, John, **Reddy**, Swarna, springers Publications, 2016

**REFERENCE BOOKS:**

1. Doing Data Science: Straight Talk From The Frontline, 1st edition, Cathy O'Neil and Rachel Schutt, O'Reilly, 2013
2. Mining of Massive Datasets, 2nd edition, Jure Leskovek, Anand Rajaraman and Jeffrey Ullman, v2.1, Cambridge University Press, 2014

## Department of Computer Science & Engineering
## University College of Engineering, JNT University Kakinada

### R20 Regulations
**PE &PCE DEPARTMENT**

**CLOUD COMPUTING**

**Course Objectives:**

- Ability to understand various service delivery models of a cloud computing architecture.
- Ability to understand the ways in which the cloud can be programmed and deployed.
- Understanding cloud service providers.
- Understand and learn the Characteristics of IaaS and Characteristics of PaaS
- Learn and gain knowledge about the Cloud Service providers.

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

| CO | Course Outcomes | Knowledge Level (K)# |
|---|---|---|
| CO1 | Explain different types of Computing | K3 |
| CO2 | Illustrate Four types of Cloud Deployment Models | K4 |
| CO3 | Demonstrate different Phases of Cloud Migration Approaches for Cloud Migration | K5 |
| CO4 | Analyze and Develop Cloud Service Models | K3 |
| CO5 | Design applications for an organization which use cloud environment. | K4 |

**Mapping of course outcomes with program outcomes**

| | PO1 | PO2 | PO3 | PO4 | PO5 | PO6 | PO7 | PO8 | PO9 | PO10 | PO11 | PO12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CO1 | 1 | 2 | 3 | 2 | | | | | 2 | | 2 | 1 |
| CO2 | 2 | | 3 | 1 | 1 | 2 | | | | 2 | 1 | |
| CO3 | | | 3 | | | | | | 3 | 1 | | 1 |
| CO4 | 1 | | 2 | 3 | 2 | 2 | | | | 2 | | |
| CO5 | 2 | 2 | | 3 | 2 | 1 | | | 3 | 1 | 2 | |

**(Levels of Correlation, viz., 1-Low, 2-Moderate, 3-High)**

**SYLLABUS:**

**UNIT-I: Computing Paradigms:** High-Performance Computing, Parallel Computing, Distributed Computing, Cluster Computing, Grid Computing, Cloud Computing, Quantum Computing, Optical Computing, Nano computing.

**UNIT-2: Cloud Computing Fundamentals:** Motivation for Cloud Computing, The Need for Cloud Computing, Definition of Cloud computing, Cloud Computing Is a Service, Principles of Cloud computing, Five Essential Characteristics, Four Cloud Deployment Models.

**UNIT-3: Cloud Computing Architecture and Management:** Cloud architecture, Layer, Anatomy of the Cloud, Network Connectivity in Cloud Computing, Applications, Managing the Cloud Infrastructure Managing the Cloud application, Migrating Application to Cloud, Phases of Cloud Migration Approaches for Cloud Migration.

**UNIT-4: Cloud Service Models:** Infrastructure as a Service, Characteristics of IaaS. Suitability of IaaS, Pros and Cons of IaaS, Summary of IaaS Providers, Platform as a Service, Characteristics of PaaS, Suitability of PaaS, Pros and Cons of PaaS, Summary of PaaS Providers, Software as a Service, Characteristics of SaaS, Suitability of SaaS, Pros and Cons of SaaS, Summary of SaaS Providers.

**UNIT-5**: Python for Amazon Web Services, Python for Google Cloud Platform, Python for Windows Azure, Python for MapReduce, Python Packages of Interest, Python Web Application Framework – Django, Designing a RESTful Web API.

**TEXT BOOKS:**

1. Essentials of cloud Computing: K. Chandrasekhran, CRC press, 2014
2. ArshadeepBhaga, Vijay Madisetti, “Cloud Computing AHandson Approach”, Universities Press, 2018.

**REFERENCE BOOKS:**

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