# GAYATRI VIDYA PARISHAD COLLEGE FOR DEGREE AND P.G. COURSES (A)

RUSHIKONDA, VISAKHAPATANAM 530045 | website: www.gvpcdpgc.edu.in

(Approved by A.I.C.T.E | Affiliated to Andhra University | An ISO 9001:2015 Certified Institute)

## ENGINEERING AND TECHNOLOGY PROGRAM

## DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING - (AI and ML)

**B. Tech Computer Science and Engineering AI and ML**

**(R-22 Regulation)**

**I Year – I Semester**

| Course code | Category | Course Title | Hours per week | | | Internal Marks | External Marks | Total Marks | Credits |
|---|---|---|---|---|---|---|---|---|---|
| | | | L | T | P | | | | |
| CSM1101 | BS | Engineering Mathematics-I (Partial Differentiation, Multiple Integrals, Fourier Series and Applications) | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM1102 | BS | Green Chemistry | 3 | 1 | 0 | 30 | 70 | 100 | 3 |
| CSM1103 | HSS | English | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM1104 | ES | Computer Programming Using 'C' | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM1105 | ES | IT Essentials | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM1106 | HSS | Communication Skills Lab | 0 | 0 | 3 | 50 | 50 | 100 | 1.5 |
| CSM1107 | BS | Green Chemistry Lab | 0 | 0 | 3 | 50 | 50 | 100 | 1.5 |
| CSM1108 | ES | Computer Programming using 'C' LAB | 0 | 0 | 3 | 50 | 50 | 100 | 1.5 |
| **Total Credits** | | | | | | | | | **19.5** |

# B. Tech Computer Science and Engineering ( AI and ML)
# (R-22 Regulation)

### I Year – II Semester

| Course code | Category | Course Title | Hours per week | | | Internal Marks | External Marks | Total Marks | Credits |
|---|---|---|---|---|---|---|---|---|---|
| | | | L | T | P | | | | |
| CSM1201 | BS | Engineering Mathematics-II (Matrix Algebra, Ordinary Differential Equations and Laplace Transforms) | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM1202 | BS | Engineering Physics | 3 | 1 | 0 | 30 | 70 | 100 | 3 |
| CSM1203 | ES | Data Structures Using 'C' | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM1204 | ES | Engineering Graphics | 1 | 0 | 4 | 30 | 70 | 100 | 3 |
| CSM1205 | ES | Discrete Mathematical Structures | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM1206 | ES | IT Workshop Lab | 0 | 0 | 3 | 50 | 50 | 100 | 1.5 |
| CSM1207 | BS | Engineering Physics Lab | 0 | 0 | 3 | 50 | 50 | 100 | 1.5 |
| CSM1208 | ES | Data Structures Lab | 0 | 0 | 3 | 50 | 50 | 100 | 1.5 |
| **Total Credits** | | | | | | | | | **19.5** |

# B. Tech Computer Science and Engineering (AI and ML)
# (R-22 Regulation)

### II Year – I Semester

| Course code | Category | Course Title | Hours per week | | | Internal Marks | External Marks | Total Marks | Credits |
|---|---|---|---|---|---|---|---|---|---|
| | | | L | T | P | | | | |
| CSM2101 | ES | Digital Logic Design & Computer Architecture | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM2102 | PC | Artificial Intelligence – I | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM2103 | HSS | Managerial Economics & Financial Accounting | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM2104 | PC | Operating Systems | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM2105 | PC | Object Oriented Programming Through Java | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM2106 | PC | Artificial Intelligence Lab | 0 | 0 | 3 | 50 | 50 | 100 | 1.5 |
| CSM2107 | PC | Object Oriented Programming Through Java Lab | 0 | 0 | 3 | 50 | 50 | 100 | 1.5 |
| CSM2108 | PC | Operating Systems Lab | 0 | 0 | 3 | 50 | 50 | 100 | 1.5 |
| CSM2109 | SC | Skill Course – 1 Python Programming | 1 | 0 | 2 | 50 | 50 | 100 | 2 |
| CSM2110 | MC | Environmental Science | 3 | 0 | 0 | 30 | 70 | 100 | 0 |
| **Total Credits** | | | | | | | | | **21.5** |

# B. Tech Computer Science and Engineering(AI and ML)
# (R-22 Regulation)

### II Year – II Semester

| Course code | Category | Course Title | Hours per week | | | Internal Marks | External Marks | Total Marks | Credits |
|---|---|---|---|---|---|---|---|---|---|
| | | | L | T | P | | | | |
| CSM2201 | ES | Mathematical Foundations for Machine Learning | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM2202 | PC | Artificial Intelligence – II | 3 | 1 | 0 | 30 | 70 | 100 | 3 |
| CSM2203 | PC | Database Management Systems | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM2204 | PC | Design and Analysis of Algorithms | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM2205 | BS | Probability & Statistical Methods | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM2206 | PC | Neural Networks Lab | 0 | 0 | 3 | 50 | 50 | 100 | 1.5 |
| CSM2207 | PC | Database Management Systems Lab | 0 | 0 | 3 | 50 | 50 | 100 | 1.5 |
| CSM2208 | SC | Skill Course – 2 Design Thinking and Innovation | 1 | 0 | 2 | 50 | 50 | 100 | 2 |
| CSM2209 | MOOCS | Massive Open Online Courses (MOOCs) (NPTEL/COURSERA/ UDEMY) | 0 | 0 | 3 | 50 | 50 | 100 | 1.5 |
| CSM2210 | MC | NCC/NSS | 0 | 0 | 2 | - | - | - | 0 |
| **Total Credits** | | | | | | | | | **21.5** |
| **Summer Internship – I (Evaluation will be done in 3-1)** | | | | | | | | | |
| **Honors & Minors** | | | | | | | | | **4** |

# B. Tech Computer Science and Engineering (AI and ML)
## (R-22 Regulation)
### III Year – I Semester

| Course code | Category | Course Title | Hours per week | | | Internal Marks | External Marks | Total Marks | Credits |
|---|---|---|---|---|---|---|---|---|---|
| | | | L | T | P | | | | |
| CSM3101 | PC/PCC | Data Warehousing and DataMining | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM3102 | PC/PCC | Object Oriented Software Engineering | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM3103 | PC/PCC | Computer Networks | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM3104 | OEC/JOE | Open Elective – I | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM3105 | PEC | Elective-I | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM3106 | PC/PCC | Computer Networks Lab | 0 | 0 | 3 | 50 | 50 | 100 | 1.5 |
| CSM3107 | PC/PCC | Software Engineering& Mini Project Lab | 0 | 0 | 3 | 50 | 50 | 100 | 1.5 |
| CSM3108 | SAC/SC | Skill Course – 3 Web Technologies | 1 | 0 | 2 | 50 | 50 | 100 | 2 |
| CSM3109 | MC | Technical Communication& Soft Skills | 2 | 0 | 0 | 100 | 0 | 100 | 0 |
| Summer Internship, Two months mandatory after 2nd year to be evaluated during 5th Semester | | | 0 | 0 | 3 | 0 | 100 | 100 | 1.5 |
| **Total Credits** | | | | | | | | | **21.5** |

| Title of the Program | L | T | P | Credits |
|---|---|---|---|---|
| **Honors/Minor Courses (The hours distribution can be 3-0-2 or 3-1-0 also)** | **3** | **1** | **0** | **4** |

**Elective-I**
Introduction to Data Science
**Expert Systems**
Pattern Recognition

**Open Elective – I**
Offered by Mechanical Engineering Offered by Civil Engineering Offered by ECE
Offered by CSE
Offered by CSE(AI&ML): Introduction to Artificial Intelligence

# B. Tech Computer Science and Engineering (AI and ML)
# (R-22 Regulation)
## III Year – II Semester

| Course code | Category | Course Title | Hours per week | | | Internal Marks | External Marks | Total Marks | Credits |
|---|---|---|---|---|---|---|---|---|---|
| | | | L | T | P | | | | |
| CSM3201 | PC/PCC | Machine Learning | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM3202 | PC/PCC | Soft Computing | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM3203 | PC/PCC | Automata Theory & Compiler Design | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM3204 | OEC/JOE | Open Elective - II | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM3205 | PEC | Elective - II | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM3206 | PC/PCC | Machine Learning Lab | 0 | 0 | 3 | 50 | 50 | 100 | 1.5 |
| CSM3207 | PC/PCC | Soft Computing Lab | 0 | 0 | 3 | 50 | 50 | 100 | 1.5 |
| CSM3208 | PEC | Elective – II Lab | 0 | 0 | 3 | 50 | 50 | 100 | 1.5 |
| CSM3209 | SAC/SC | Skill Course – 4 Android Programming | 1 | 0 | 2 | 50 | 50 | 100 | 2 |
| CSM3210 | MC | Essence of Indian Traditional Knowledge | 3 | 0 | 0 | 30 | 70 | 100 | 0 |
| **Total Credits** | | | | | | | | | **21.5** |
| **Summer Industrial Research Internship (2 months) Mandatory** | | | | | | | | | |

| Title of the Program | L | | P | Credits |
|---|---|---|---|---|
| **Honors/Minor Courses (The hours distribution can be 3-0-2 or 3-1-0 also)** | **3** | **1** | **0** | **4** |

**Elective-II**

**Introduction to Arduino and Raspberry PI**
Big Data Analytics
Cryptography & Network Security

**Open Elective – II**
Offered by Mechanical Engineering Offered by Civil Engineering Offered by ECE
Offered by CSE
Offered by CSE(AI&ML): Introduction to Machine Learning

# B. Tech Computer Science and Engineering (AI and ML)
## (R-22 Regulation)
### IV Year – I Semester

| Course code | Category | Course Title | Hours per week | | | Internal Marks | External Marks | Total Marks | Credits |
|---|---|---|---|---|---|---|---|---|---|
| | | | L | T | P | | | | |
| CSM4101 | PEC | Elective -III | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM4102 | PEC | Elective -IV | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM4103 | PEC | Elective -V | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM4104 | OEC/JOE | Open Elective-III | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM4105 | OEC/JOE | Open Elective-IV | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM4106 | HSS/HSMS | Professional Ethics and Universal Human Values (Understanding Harmony) | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM4107 | SAC/SC | Skill Course – 5:<br>POWER BI | 1 | 0 | 2 | 50 | 50 | 100 | 2 |
| CSM4108 | Industrial/ Research Internship (2 months Mandatory after 6th Semester to be evaluated in 7th Semester) | | 0 | 0 | 0 | 0 | 100 | 100 | 3 |
| **Total Credits** | | | | | | | | | **23** |

| Title of the Program | L | T | P | Credits |
|---|---|---|---|---|
| **Honors/Minor Courses (The hours distribution can be 3-0-2 or 3-1-0 also)** | **3** | **1** | **0** | **4** |

**Elective-III**

**Natural Language Processing**
Cloud Computing

Reinforcement Learning

**Elective-IV**

Computer Vision
Wireless Sensor Networks
**Cyber Security & Digital Forensics**

**Elective- V**
**Deep Learning**
R-Programming
Robotics

**Open Elective - III**

Offered by ME

Offered by CE

Offered by ECE

**Open Elective - IV**

Offered by ME
Offered by CE
Offered by ECE

# B. Tech Computer Science and Engineering (AI and ML)
## (R-22 Regulation)
### IV Year – II Semester

| Course code | Category | Course Title | Hours per week | | | Internal Marks | External Marks | Total Marks | Credits |
|---|---|---|---|---|---|---|---|---|---|
| | | | L | T | P | | | | |
| Proj | Major Project | Project Work, Seminar & Internship in Industry | 0 | 0 | 0 | 50 | 50 | 100 | 12 |
| | | | | | | | | **Total Credits** | **12** |

# B. Tech Computer Science and Engineering with AI & ML
# (R-22 Regulation)

### I Year – I Semester

| Course code | Category | Course Title | Hours per week | | | Internal Marks | External Marks | Total Marks | Credits |
|---|---|---|---|---|---|---|---|---|---|
| | | | L | T | P | | | | |
| CSM1101 | BS | Engineering Mathematics-I (Partial Differentiation, Multiple Integrals, Fourier Series and Applications) | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM1102 | BS | Green Chemistry | 3 | 1 | 0 | 30 | 70 | 100 | 3 |
| CSM1103 | HSS | English | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM1104 | ES | Computer Programming Using 'C' | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM1105 | ES | IT Essentials | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM1106 | HSS | Communication Skills Lab | 0 | 0 | 3 | 50 | 50 | 100 | 1.5 |
| CSM1107 | BS | Green Chemistry Lab | 0 | 0 | 3 | 50 | 50 | 100 | 1.5 |
| CSM1108 | ES | Computer Programming using 'C' LAB | 0 | 0 | 3 | 50 | 50 | 100 | 1.5 |
| **Total Credits** | | | | | | | | | **19.5** |

# GAYATRI VIDYA PARISHAD COLLEGE FOR DEGREE AND PG COURSES (AUTONOMOUS)
# ENGINEERING AND TECHNOLOGY PROGRAM
# RUSHIKONDA CAMPUS, VISAKHAPATNAM–530045.

| Course code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int. | Ext | |
| CS1101 | Engineering Mathematics-I ( Partial Differentiation, Multiple Integrals, Fourier Series and Applications) | 3 | 0 | 0 | 30 | 70 | 3 |

## COURSE OBJECTIVES

1. To transmit the knowledge of Partial differentiation.
2. To know of getting maxima and minima of function of two variables and finding errors and approximations.
3. To evaluate double and triple integrals, volumes of solids and area of curved surfaces.
4. To expand a periodical function as Fourier series and half-range Fourier series.

## COURSE OUTCOMES:

CO 1: To determine the partial derivatives of functions of two or more variables.
CO 2: Evaluate maxima and minima, errors and approximations.
CO 3: Ability to evaluate double and triple integrals.
CO 4: Ability to find volumes of solids and area of curved surfaces.
CO 5: To expand a periodical function as Fourier series and half-range Fourier series.

## UNIT-I

**Partial Differentiation:** Introduction - Functions of two or more variables - Partial derivatives - Homogeneous functions – Euler's theorem - Total derivative - Change of variables – Jacobins.

## UNIT-II

**Applications of Partial Differentiation:** Geometrical interpretation -Tangent plane and Normal to a surface -Taylor's theorem for functions of two variables - Errors and approximations -Total differential. Maxima and Minima of functions of two variables - Lagrange's method of undetermined multipliers - Differentiation under the integral Sign - Leibnitz's rule.

## UNIT-III

**Multiple Integrals:** Introduction - Double Integrals - Change of Order of Integration - Double Integrals in Polar Coordinates - Triple Integrals - Change of Variables.

**UNIT-IV**

**Multiple Integrals-Applications:** Area enclosed by plane curves - Volumes of solids - Area of a curved surface - Beta Function - Gamma Function - Relation between Beta and Gamma Functions.

**UNIT-V**

**Fourier Series:** Introduction - Euler's Formulae - Conditions for a Fourier Expansion - Functions having points of discontinuity - Change of Interval - Odd and Even Functions - Expansions of Odd or Even Periodic Functions, Half-Range Series - Parseval's Formula.

**Text Book:**

Scope and Treatment as in "Higher Engineering Mathematics", by Dr. B.S. Grewal, 43rd Edition, Khanna publishers.

**Reference Books:**

1. Graduate Engineering Mathematics by V B Kumar Vatti., I.K.International publishing house Pvt. Ltd.
2. Advanced Engineering Mathematics by Erwin Kreyszig.
3. A text book of Engineering Mathematics, by N.P. Bali and Dr. Manish Goyal, Lakshmi Publications.
4. Advanced Engineering Mathematics by H.K. Dass. S. Chand Company.
5. Higher Engineering Mathematics by B.V. Ramana, Tata Mc Graw Hill Company.
6. Higher Engineering Mathematics by Dr. M.K.Venkataraman.

# GAYATRI VIDYA PARISHAD COLLEGE FOR DEGREE AND PG COURSES (AUTONOMOUS)
# ENGINEERING AND TECHNOLOGY PROGRAM
# RUSHIKONDA CAMPUS, VISAKHAPATNAM–530045.

| Course code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int. | Ext | |
| CS1102 | Green Chemistry | 3 | 0 | 0 | 30 | 70 | 3 |

**COURSE OBJECTIVES**

CO 1: To apply the basic knowledge of Chemistry to the Engineering Discipline.
CO 2: To develop knowledge about water and its treatment for industrial and potable purposes.
CO 3: To develop understanding in the areas of Batteries, Fuels Mechanism of Corrosion of Metals and Corrosion Control Methods, Green Chemistry and Technology and Processes involving Green Chemistry and apply the knowledge for solving existing challenges faced in various engineering and societal areas.

**LEARNING OUTCOMES:**

LO 1: The students are able to apply the basic concepts and principles studied in Chemistry to the field of Engineering.
LO 2: The students are able to apply chemistry to different branches of engineering.
LO 3: The students are able to acquire the knowledge in the areas of Water Chemistry, Mechanism of Corrosion of Metals and Corrosion Control Methods, Batteries, Fuel Cells, Green Chemistry and Technology and Processes involving Green Chemistry and suggest innovative solutions for existing challenges in these areas.

**UNIT-I**

**Water Technology:** Sources of Water – Impurities – WHO Limits – Hardness and its Determination by EDTA method– Boiler Troubles– Water Softening Methods – Lime-Soda, Zeolite and Ion Exchange - Municipal Water Treatment-Break Point Chlorination – Desalination of Sea Water : Reverse Osmosis and Electrodialysis. Methods

**UNIT-II**

**Batteries:** Definition, types, Primary batteries: Zinc-carbon (Leclanche type), zinc alkaline (Duracell), zinc/air batteries; Lithium primary cells – . Secondary batteries: Lead acid and VRLA (valve regulated (sealed) lead acid), nickel-cadmium, nickel-metal hydride batteries, lithium ion batteries, ultrathin lithium polymer cells. Advanced Batteries for electric vehicles: requirements of the battery – sodium- beta and redox batteries.

**UNIT-III**

**Fuel Cells:** Definition, Description, working principle, advantages, disadvantages, $H_2$_$O_2$ fuel cells: alkaline fuel cells, phosphoric acid, solid oxide, molten carbonate, methanol fuel cells- Proton Membrane fuel cells.

**UNIT-IV**

**Corrosion:** Origin and Theory – Types of Corrosion: Chemical and Electrochemical; Pitting, Waterline, and Stress corrosion – Galvanic Series – Factors Effecting Corrosion. Corrosion Controlling Methods: Protective Coatings: Metallic Coatings, Electroplating and Electroless Plating.

**UNIT-V**

**Green Chemistry and Technology**: Introduction and significance of Green Chemistry, goals of green chemistry, 12 principles of green chemistry, toxicity of chemicals, material safety data sheet (MSDS), concept of zero pollution technologies: atom economy, functional toxicity Vs non-functional toxicity, functional group approaches to green chemistry, Elemination of toxic functional group, optimization of frame works for the design of the dgreener synthetic pathways, applications of green chemistry – Green solvents, green fuels and propellants, biocatalysis.

**Text Books:**

1. Engineering Chemistry – PC Jain and M. Jain – DhanpathRai and Sons, New Delhi.
2. A Text book of Engineering Chemistry – S. S. Dara – S. Chand & Co. New Delhi. 3. Dell, Ronald M Rand, David A J, ‘Understanding Batteries’, Royal Society of Chemistry, (2001).
3. Hand Book of Green Chemistry and Technology; by James Clarke and Duncan Macquarrie; Blakwell Publishing.

# GAYATRI VIDYA PARISHAD COLLEGE FOR DEGREE AND PG COURSES (AUTONOMOUS)
# ENGINEERING AND TECHNOLOGY PROGRAM
# RUSHIKONDA CAMPUS, VISAKHAPATNAM–530045.

| Course code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int. | Ext | |
| CS1103 | English | 3 | 0 | 0 | 30 | 70 | 3 |

**COURSE OUTCOMES**

**CO-1:**

- Addressing explicit and implicit meanings of a text on current topics.
- Understanding the context.
- Learning new words and phrases.
- Using words and phrases in different contexts.

**CO-2:**

- Using the basic structure of a sentence.
- Applying relevant writing formats to create paragraphs, essays, letters, emails, reports and presentations.
- Retaining a logical flow while writing.
- Planning and executing an assignment creatively.
- Participating in discussions and influencing them and communicating ideas effectively.
- Examining self-attributes and identifying areas that require improvement: self-diagnosis and self-motivation.

**CO-3:**

- Analyzing a topic of discussion and relating it to time management skills.
- Participating in discussions and influencing them.
  Communicating ideas effectively.
  Presenting ideas coherently within a stipulated time.

**CO-4:**

- Examining self-attributes and identifying areas that require improvement: self-diagnosis and self-motivation.
- Adapting to a given situation and developing a functional approach to finding solutions: adaptability and problem solving.
- Understanding the importance of helping others: community services and enthusiasm.

**CO-5:**

- The student will learn to avoid redundancy will learn common abbreviations useful for competitive exams and will acquire basic proficiency in English including reading, comprehension and writing skills.
- The student will be motivated with a sense of purpose throughout the course by learning life skills.

**DETAILED SYLLABUS**

**Topics:**

*On the conduct of life:* William Hazlitt

**Life skills: Values and Ethics**

*If:* Rudyard Kipling

*The Brook:* Alfred Tennyson

**Life skills: Self-Improvement**

*How I Became a Public Speaker*: George Bernard Shaw

*The Death Trap*: Saki

**Life skills: Time Management**

*On saving Time*: Seneca

*Chindu Yellama*

**Life skills: Innovation**

*Muhammad Yunus*

*Politics and the English Language*: George Orwell

**Life skills: Motivation**

*Dancer with a White Parasol*: Ranjana Dave

**Grammar:**

Prepositions – Articles – Noun-Pronoun Agreement, Subject-Verb Agreement – Misplaced Modifiers – Clichés, Redundancies, Punctuations.

**Vocabulary:**

Introduction to Word Formation – Root Words from other Languages – Prefixes and Suffixes – Synonyms, Antonyms – Common Abbreviations

**Writing:**

E-mail writing– Principles of Good Writing – Essay Writing –Paragraph Writing (with hints) Writing a Summary.

**Textbook:**

*Language and Life: A Skills Approach* Board of Editors, Orient Black swan Publishers, India. 2018.

**Reference Books:**

Practical English Usage. Michael Swan. OUP. 1995.
Remedial English Grammar. F.T. Wood. Macmillan.2007
On Writing Well. William Zinsser. Harper Resource Book. 2001

1. Study Writing. Liz Hamp-Lyons and Ben Heasly. Cambridge University Press. 2006.
2. Communication Skills. Sanjay Kumar and PushpLata. Oxford University Press. 2011.
3. Exercises in Spoken English. Parts. I-III. CIEFL, Hyderabad. Oxford UniversityPress.
4. University Physics by Young & Freedman
5. Nonconventional Energy by Ashoke V. Desai

| Course code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int. | Ext | |
| CS1104 | Computer Programming using 'C' | 3 | 0 | 0 | 30 | 70 | 3 |

**COURSE OBJECTIVES**

1. The course is designed to provide complete knowledge of C language.
2. To provide students with understanding of code organization and functional hierarchical decomposition with using complex data types.
3. To provide knowledge to the students to develop logics which will help them to create programs, applications in C.
4. This course aims to identify tasks in which the numerical techniques learned are applicable and apply them to write programs, and hence use computers effectively to solve the task.
5. This course provides the fundamental knowledge which is useful in understanding the other programming languages.

**COURSE OUTCOMES**

At the end of the course students will be able to:
CO1. Identify basic elements of C programming structures like data types, operators, constants, input- output statements and decision-making statements.
CO2. Apply Various Operations on derived data types like arrays & strings.
CO3. Design and Implementation of Modular Programming using functions.
CO4. Design and Implementation of Modular Programming using Pointers.
CO5. Develop C programs using user defined data types like structures and unions.

**UNIT-I**

**Introduction to C and Decision Making,:** Basic structure of C program, Constants, Variables and data types, Operators and Expressions, Arithmetic Precedence and associativity, Type Conversions. Managing Input and Output Operations Formatted Input, Formatted Output, Decision making with if statement, Simple if statement, The if…else statement, Nesting of if…else statement, the else.. if ladder, switch statement.

**UNIT-II**

**Branching, Looping, Arrays & Strings:**, the GOTO statement., The while statement, the do statement, The for statement, Jumps in Loops, One, Two-dimensional Arrays, Character Arrays. Declaration and

Initialization of Strings, reading and writing of strings, String handling functions, Table of strings.

**UNIT-III**

**Functions:** Definition of Functions, Return Values and their Types, Function Calls, Function Declaration, Category of Functions: No Arguments and no Return Values, Arguments but no Return Values, Arguments with Return Values, No Argument but Returns a Value, Functions that Return Multiple Values. Nesting of functions, recursion, passing arrays to functions, passing strings to functions, the scope, visibility and lifetime of variables.

**UNIT-IV**

**Pointers:** Accessing the address of a variable, declaring pointer variables, initializing of pointer variables, accessing variables using pointers, chain of pointers, pointer expressions, pointers and arrays, pointers and character strings, array of pointes, pointers as function arguments, functions returning pointers, pointers to functions, pointers to structures-Program Applications.

**UNIT-V**

**Structure, Unions and File handling: :** Defining a structure, declaring structure variables, accessing structure members, structure initialization, copying and comparing structure variables, arrays of structures, arrays within structures, structures within structures, structures and functions and unions, size of structures and bit-fields- Defining and opening a file, closinga file, Input/ Output operations on files, Error handling during I/O operations, random access to files and Command Line Arguments-Program Applications.

**Text Books**

1. Programming in ANSI C, E Balagurusamy, 6th Edition. McGraw Hill Education (India)Private Limited.

**Reference Books**

1. Let Us C ,Yashwant Kanetkar, BPB Publications, 5th Edition.
2. Computer Science, A structured programming approach using C", B.A.Forouzan and R.F.Gilberg, " 3rd Edition, Thomson, 2007.
3. The C –Programming Language' B.W. Kernighan, Dennis M. Ritchie, PHI.
4. Scientific Programming: C-Language, Algorithms and Models in Science, Luciano M. Barone (Author), EnzoMarinari (Author), Giovanni Organtini, World Scientific.

| Course code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| CS1105 | IT Essentials | L | T | P | Int. | Ext | 3 |
| | | 3 | 0 | 0 | 30 | 70 | |

**COURSE OBJECTIVES**

1. Select the appropriate computer components to build, repair, or upgrade personalcomputers.
2. Explain how to correctly use tools and safely work in a lab.
3. Install components to build, repair, or upgrade personal computers.
4. Configure computers to communicate on a network
5. Configure devices to connect to the Internet and Cloud services
6. Explain how to use, configure, and manage laptops and mobile devices

**COURSE OUTCOMES**

At the end of the course student will be able to

CO1: Understands the roles and responsibilities of the IT professional
CO2: Troubleshoot advanced hardware and software problems
CO3: Understand the basics of Operating Systems
CO4: Learn the Safe Lab Procedures and Tools CO5:
Understand Networks and Security Concepts.
.

**UNIT- I**

**Introduction to the Personal Computer** Describe a Computer System, Identify the Names, Purposes, and Characteristics of Cases and Power Supplies, Identify the Names, Purposes, and Characteristics of Internal Components, Identify the Names, Purposes, and Characteristics of Ports and Cables, Identify the Names, Purposes, and Characteristics of Input Devices, Identify the Names, Purposes, and Characteristics of Output Devices, Explain System Resources and Their Purposes.

**UNIT- II**

**Computer Assembly** Attach the Components to the Motherboard and Install the Motherboard, Install Internal Drives, Install Drives in External Bays, Install Adapter Cards, Connect the Power Cables Reattach the Side Panels to the Case, Boot the Computer for the First Time. **Basics of Preventive**

**Maintenance and Troubleshooting** Explain the Purpose of Preventive Maintenance, Identify the Steps of the Troubleshooting Process.

**UNIT- III**

**Fundamental Operating Systems** Explain the Purpose of an Operating System, Describe and Compare Operating Systems to Include Purpose, Limitations, and Compatibilities, Determine Operating System Based on Customer Needs, Install an Operating System, Identify and Apply Common Preventive Maintenance Techniques for Operating Systems, Troubleshoot Operating Systems.

**UNIT- IV**

**Safe Lab Procedures and Tool Use** Explain the Purpose of Safe Working Conditions and Procedures, Identify Tools and Software Used with Personal Computer Components and Their Purposes, Implement Proper Tool Use. **Fundamental Laptops and Portable Devices** Identify Common Preventive Maintenance Techniques for Laptops and Portable Devices, Describe How to Troubleshoot Laptops and Portable Devices.

**UNIT-V**

**Fundamental Networks** Explain the Principles of Networking, Describe Types of Networks, Describe Basic Networking Concepts and Technologies, Describe the Physical Components of a Network, Describe LAN Topologies and Architectures. **Fundamental Security:** Explain Why Security Is Important, Describe Security Threats, Identify Security Procedures, Identify Common Preventive Maintenance Techniques for Security, Troubleshoot Security.

**Text Books:**

1. IT Essentials: PC Hardware and Software Companion Guide Fourth Edition, CiscoNetworking Academy.

**Reference Books:**

1. Network security essentials application and standrads, by William stallings, 4th edition, prentice hall.
2. Mike Meyers' CompTIA A+ Guide to Managing and Troubleshooting PCs, Sixth Edition 6th Edition

| Course code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| CS1106 | Communication Skills LAB | L | T | P | Int. | Ext | 1.5 |
| | | 0 | 0 | 3 | 50 | 50 | |

**COURSE OBJECTIVES**

1. To make students recognize the sounds of English through Audio-Visual aids;
2. To help students build their confidence and help them to overcome their inhibitions and self-consciousness while speaking in English;
3. To familiarize the students with stress and intonation and enable them to speak English effectively; and
4. To give learners exposure to and practice in speaking in both formal and informal contexts.

**COURSE OUTCOMES**

CO1: Students will be sensitized towards recognition of English sound patterns and the fluency in their speech will be enhanced;
CO2: Students will be able to participate in group activities like roleplays, group discussions and debates; and
CO3: Students will be able to express themselves fluently and accurately in social as well professional context.

**Topics:**

**Introduction to Phonetics:** The Sounds of English (Speech sound – vowels and consonants) - Stress and Intonation - Accent and Rhythm.

**Listening Skills:** Listening for gist and specific information - listening for Note taking, summarizing and for opinions - Listening to the speeches of eminent personalities.

**Speaking Skills**: Self-introduction - Conversation Skills (Introducing and taking leave) - Giving and asking for information - Role Play - Just A Minute (JAM) session - Telephone etiquette.

**Reading and Writing skills:** Reading Comprehension – Précis Writing - E-Mail writing - Punctuation.

**Presentation skills**: Verbal and non-verbal communication - Body Language - Making a Presentation.

**DISTRIBUTION AND WEIGHTAGE OF MARKS**

- The practical examinations for the English Language Lab shall be conducted as per the University norms prescribed for the core Engineering practical sessions.
- For the Language lab sessions, there shall be a continuous evaluation during the semester for 50 sessional marks and 50 semester-end Examination marks.
- For the 50 sessional (Internal) marks, 30 marks shall be awarded for day-to-day performance and for completing activities in the lab manual, 20 marks to be awarded by conducting Internal Lab Test(s).
- For the 50 Semester- end (External) marks, 30 marks shall be awarded for written examination (dialogues, the sounds of English and stress) and 20 marks for External Examiner viva-voce.

**Reference Books:**


- Ashraf Rizvi. Effective Technical Communication. Tata McGraw Hill Education Private Limited, New Delhi.
- Speak Well. Orient Blackswan Publishers, Hyderabad.
- Allan Pease. Body Language. Manjul Publishing House, New Delhi.

| Course code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int. | Ext | |
| CS1107 | Green Chemistry LAB | 0 | 0 | 3 | 50 | 50 | 1.5 |

**COURSE OUTCOMES**

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

Course outcomes: At the end of the course the student shall be able to CO
1: Determine the quality of the ground water sample.
CO 2: Determine the metal ions using titrimetry.
CO 3: Explain the functioning of the instruments like pH metry, Conductometry and Potentiometry .
CO 4: Use spectrophotometry to determine the metal ions.

List of Laboratory Experiments
(Any 10 experiments to be performed during the semester)

1. Determination of sodium hydroxide with HCl .
2. Determination of Fe (II) by potassium dichromate.
3. Determination of Fe(II) by permanganometry.
4. Determination of chromium (VI) by hypo.
5. Determination of Zinc by EDTA method.
6. Determination of hardness of water sample by EDTA method.
7. Determination of available chlorine in water .
8. Determination of sulphuric acid in lead-acid storage cell.
9. Determination of carbonate and bicarbonate in a mixer.
10. Determination of strength of an acid by pH metric method.
11. Determination of citric acid in a citrus fruit by conductometric method.
12. Determination of Fe(II) in Mohr's salt by potentiometric method.
13. Construction of Galvanic cell.
14. Determination of Fe(III) by spectrophotometry.
15. Optimization of structure of the compound using Gaussian software.
16. Preparation of Biodiesel from vegetable oils.

**Reference Books:**

1. Vogel's Quantitative Chemical Analysis – V Edition – Longman

| Course code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int. | Ext | |
| CS1108 | **Computer Programming using 'C' LAB** | **0** | **0** | **3** | **50** | **50** | 1.5 |

## COURSE OBJECTIVES

1. To provide complete knowledge of C language.
2. To develop logics which will help them to create programs, applications in C.
3. To learn the basic programming constructs they can easily switch over to any other language in future.

## COURSE Outcomes

CO1: Ability to implement the programs using control structures & arrays. CO2:
Ability to implement the programs using strings & functions.
CO3: Ability to implement the programs using user defined data types.
CO4: Ability to implement the programs using pointers and operations on files.

## LIST OF EXPERIMENTS

1. **Basic Programs**
   A. C program to scan all data type variables as input and print it as output.
   B. C program to perform arithmetic operations like +,-,*,/,% on two input variables.
   C. C program to perform temperature conversions from Centigrade to Fahrenheit and vice versa.

2. **Programs on Operators**
   A. C program to perform all bit wise operations.
   B. C program to extract the last two digits of a given integer n, where the number of digits should be greater than 2.
   C. C program to display the greatest of three numbers using a conditional operator.
   D. C program to swap two numbers without using a third variable.

3. **Programs on Conditional Statements**
   A. C program to check whether a given input integer is in between two values x and y.
   B. C program to check whether a given character is a vowel or a consonant or a digit or a special symbol.
   C. C program to display the nature of roots of a quadratic equation.
   D. C program to perform arithmetic operations using switch statement.
   E. C program to convert upper case character to lowercase and vice versa.

**4. Programs on Loop Statements**

A. C program to print odd numbers between specified ranges.

B. C program to display the factors of a given number and check whether it is a prime or not.

C. C program to display the sum of individual digits of a given integer raised to the power of n. Also check whether the given integer is Armstrong or not.

D. C Program to demonstrate the usage of unconditional control statements

E. C Program to display Fibonacci series.

F. C Program to generates 100 random real numbers in the range of 10.0 to 20.0 and sort them in descending order.

G. C program to display the following pattern.

```
5 4 3 2 1
4 3 2 1
3 2 1
2 1
1
```

**5. Programs on Functions**

A. C program to demonstrate the various categories of functions with respect to return type and number of arguments.

B. C program to find the LCM of two numbers using functions.

C. Create a header file which contains the following prototype:

i. int factorial(int) ; // non-recursive function

ii. int factorial_rec(int); //Recursive function

iii. int prime(int);

Use the above functions in a C program by including the above header file.

D. C program to display Pascal's triangle using functions.

**6. Programs on Arrays**

A. C program to read n integer values into an array and display them

B. C program to count and display the number of positive, negative, even and odd numbers in a given array of integers and also display their sum.

C. C program to find the smallest and largest numbers in an array of integers.

D. C program to perform addition, multiplication, transpose of given matrices using functions.

E. C program to check whether a given integer exists in a list of numbers and print its index value if it is present, otherwise print "No".

**7. Programs on Character Array**

A. C program to convert upper case character to lowercase and vice versa in a given string.

B. C program to delete all vowels in a given string and display the remaining string.

C. C program to check whether a given string is palindrome or not.

**8. Programs using String handling functions**

A. C program to demonstrate the usage of at least 10 predefined string handling functions.
B. C Program to Sort the given n strings in ascending order.
C. C program that implements the following user defined string handling functions
   i. To find the length of the given string
   ii. To copy the contents of one string to another
   iii. To reverse the contents of a string
   iv. To compare two strings
   v. To concatenate two strings

**9. Programs on Pointers and Dynamic Memory Allocation**

A. C program to demonstrate the usage of pointers.
B. C program that uses dynamic memory allocation functions to add n elements and display their average.
C. C program that performs pointer arithmetic.
D. C program that implements call by reference.
E. C program to demonstrate the following
   i. Pointers to Pointers
   ii. Array of Pointers
   iii. Pointer to Array
   iv. Pointers to Functions

**10. Programs on Structures**

A. C program to read the data of four students, each students has a name (string), roll number (string), age (integer), use an array of structure. Later find the average age of the students.
B. C program using nested structures.
C. C program that demonstrates different ways to access the structure elements using pointers.

**11. Programs on Files**

A. C program to read the contents of a file and display on the output screen.
B. C program to copy the contents of one file to another.
C. C program to count and display the number of characters, words and lines in a file.

**12. Programs on Command Line Arguments**

A. C program to perform arithmetic operations using command line arguments.
B. C program to pass file name as argument and display the contents of file on screen.

# B. Tech Computer Science and Engineering (AI and ML)
# (R-22 Regulation)

### I Year – II Semester

| Course code | Category | Course Title | Hours per week | | | Internal Marks | External Marks | Total Marks | Credits |
|---|---|---|---|---|---|---|---|---|---|
| | | | L | T | P | | | | |
| CSM1201 | BS | Engineering Mathematics-II (Matrix Algebra, Ordinary Differential Equations and Laplace Transforms) | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM1202 | BS | Engineering Physics | 3 | 1 | 0 | 30 | 70 | 100 | 3 |
| CSM1203 | ES | Data Structures Using 'C' | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM1204 | ES | Engineering Graphics | 1 | 0 | 4 | 30 | 70 | 100 | 3 |
| CSM1205 | ES | Discrete Mathematical Structures | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM1206 | ES | IT Workshop Lab | 0 | 0 | 3 | 50 | 50 | 100 | 1.5 |
| CSM1207 | BS | Engineering Physics Lab | 0 | 0 | 3 | 50 | 50 | 100 | 1.5 |
| CSM1208 | ES | Data Structures Lab | 0 | 0 | 3 | 50 | 50 | 100 | 1.5 |
| **Total Credits** | | | | | | | | | **19.5** |

# GAYATRI VIDYA PARISHAD COLLEGE FOR DEGREE AND PG COURSES (AUTONOMOUS)
# ENGINEERING AND TECHNOLOGY PROGRAM
# RUSHIKONDA CAMPUS, VISAKHAPATNAM–530045.

| Course code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int. | Ext | |
| CS1201 | **Engineering Mathematics-II (Matrix Algebra, Ordinary Differential Equations and Laplace Transforms)** | 3 | 0 | 0 | 30 | 70 | 3 |

## COURSE OBJECTIVES

- The way of obtaining rank, Eigen values and Eigen vectors of a matrix.
- To know the importance of Cayley-Hamilton theorem and getting canonical form from a given quadratic form.
- To solve the system of equations by using direct and indirect methods.
- To solve first order and higher order differential equations by various methods.
- To obtain the Laplace transforms and inverse Laplace transforms for a given functions and their applications.

## COURSE OUTCOMES

CO1: Find rank, Eigen values and Eigen vectors of a matrix and understand the importance of Cayley-Hamilton theorem.

CO2: Reduce quadratic form to canonical forms and solving linear systems by direct and indirect methods.

CO3: Demonstrate solutions to first order differential equations by various methods and solve basic applications problems related to electrical circuits, orthogonal trajectories and Newton's law of cooling

CO4: Discriminate among the structure and procedure of solving higher order differential equations with constant and variable coefficients.

CO5: Understand Laplace transforms and its properties and finding the solution of ordinary differential equations.

### UNIT-I

**Matrix Algebra:** Rank of a matrix- Echelon form, Normal Form - Solution of Linear System of Equations - Consistency of Linear System of Equations - Gauss elimination method, LU Factorization method, Complex Matrices: Hermitian, Skew-Hermitian and Unitary Matrices and their Properties.

### UNIT-II

**Eigen Values and Eigen Vectors:** Eigen Values and Eigen Vectors of a Matrix - Cayley-Hamilton theorem - Inverse and Powers of a Matrix using Cayley-Hamilton's theorem and its applications.

Diagonalization of a Matrix - Quadratic Forms - Reduction of Quadratic Form to Canonical Form - Nature of a Quadratic Form.

**UNIT-III**

**Ordinary Differential Equations of First Order and its Applications:** Formation of ordinary differential equations (ODEs) - Solution of an ordinary differential equation - Equations of the first order and first degree - Linear differential equation - Bernoulli's equation - Exact differential equations - Equations reducible to exact equations - Orthogonal Trajectories - Simple Electric (LR & CR) Circuits - Newton's Law of Cooling - Law of Natural growth and decay.

**UNIT-IV**

**Differential Equations of Higher Order:** Solutions of Linear Ordinary Differential Equations with Constant Coefficients - Rules for finding the complimentary function - Rules for finding the particular integral - Method of variation of parameters - Cauchy's linear equation - Legendre's linear equation - Simultaneous linear differential equations.

**UNIT-V**

**Laplace Transforms:** Introduction - Existence Conditions - Transforms of Elementary Functions - Properties of Laplace Transforms - Transforms of Derivatives - Transforms of Integrals - Multiplication by $t^n$ - Division by t – Evaluation of integrals by Laplace Transforms - Inverse Laplace Transform - Applications of Laplace Transforms to Ordinary Differential Equations - Simultaneous Linear Differential Equations with Constant Coefficients - Second Shifting Theorem - Laplace Transforms of Unit Step Function, Unit Impulse Function and Laplace Transforms of Periodic Functions.

**Text Book:**

Scope and Treatment as in "Higher Engineering Mathematics", by Dr. B.S. Grewal, 43rd Edition, Khanna publishers.

**Reference Books:**

1. Graduate Engineering Mathematics by V B Kumar Vatti., I.K. International publishing house Pvt. Ltd.
2. Advanced Engineering Mathematics by Erwin Kreyszig.
3. A text book of Engineering Mathematics, by N.P. Bali and Dr. Manish Goyal. Lakshmi Publications.
4. Advanced Engineering Mathematics by H.K. Dass. S. Chand Company.
5. Higher Engineering Mathematics by B.V. Ramana, Tata Mc Graw Hill Company.

# GAYATRI VIDYA PARISHAD COLLEGE FOR DEGREE AND PG COURSES (AUTONOMOUS)
# ENGINEERING AND TECHNOLOGY PROGRAM
# RUSHIKONDA CAMPUS, VISAKHAPATNAM–530045.

| Course code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| CS1202 | Engineering Physics | L | T | P | Int. | Ext | 3 |
| | | 3 | 0 | 0 | 30 | 70 | |

## COURSE OBJECTIVES

The fundamentals of sciences are essential to learn as the application of science in solving problems is technology. The engineering physics curriculum is designed in such a way that all branches of engineering will study the basic fundamentals of technology from where it is originated. The course objectives are

1. To impart knowledge in basic concept of physics of Thermodynamics relevant to engineering applications.
2. To grasp the concepts of physics for electromagnetism and its application to engineering. Learn production of Ultrasonics and their applications in engineering.
3. To Develop understanding of interference, diffraction and polarization: connect it to a few engineering applications.
4. To learn basics of lasers and optical fibers and their use in some applications.
5. To understand concepts and principles in quantum mechanics and Nano phase Materials. Relate them to some applications.

## COURSE OUTCOMES

By the end of this course, student would have

CO.1: Understand the fundamentals of Thermodynamics and Laws of thermodynamics. Understand the working of Carnot cycle and concept of entropy.
CO.2: Gain Knowledge on the basic concepts of electric and magnetic fields. Understand the concept of the nature of magnetic materials. Gain knowledge on electromagnetic induction and its applications.
CO.3: Understand the Theory of Superposition of waves. Understand the formation of Newton's rings and the working of Michelson's interferometer. Remember the basics of diffraction, Evaluate the path difference. Analysis of Fraunhofer Diffraction due to a single slit.
CO.4: Understand the interaction of matter with radiation, Characteristics of Lasers, Principle, working schemes of Laser and Principle of Optical Fiber. Realize their role in optical fiber communication.
CO.5: Understand the intuitive ideas of the Quantum physics and understand dual nature of matter. Compute Eigen values, Eigen functions, momentum of Atomic and subatomic particles using Time independent one Dimensional Schrodinger's wave equation. Understand the fundamentals and synthesis processes of Nanophase materials.

## UNIT-I

### THERMODYNAMICS (CO1)

**Part-I**

Introduction, Heat and Work, First law of thermodynamics and its applications, Reversible and Irreversible process, Carnot cycle and Efficiency (Problems based on efficiency), Carnot's Theorem.

**Part-II**

Second law of thermodynamics (Kelvins and Clausius statement only) , Entropy - Physical Significance, Change of entropy in reversible and irreversible process, Second law in terms of entropy, Entropy and disorder, Third law of thermodynamics (statement only).

## UNIT - II

### ELECTROMAGNETISM (CO2)

Gauss's law (Statement and Proof, without applications), Ampere's law (Statement and Proof, without applications), Faraday's law of induction, Lenz's law, Induced magnetic fields, Displacement current, Maxwell's equations (no derivation), propagation of electromagnetic waves in free space (Theory only). Ultrasonics: Properties of ultrasonic waves, production of ultrasonic waves by magnetostriction and piezoelectric methods, acoustic grating, applications of ultrasonics, acoustic grating.

## UNIT - III

### OPTICS (CO3)

**Interference:** Principle of superposition – Young's Experiment – Coherence - Interference in thin films (reflected light), Newton's Rings (Problems based on finding radius of curvature or wavelength), Michelson Interferometer and its applications (Theory only).

**Diffraction:** Introduction, Differences between interference and diffraction, Fresnel and Fraunhofer diffraction, Fraunhofer diffraction at a single slit **(**Qualitative and quantitative treatment**)**(Theory only).

**Polarisation:** Polarisation by reflection, refraction and double refraction in uniaxial crystals, Nicol prism, Quarter and Half wave plate (problems based on thickness), production and detection of plane, circular and elliptical polarization (Theory only).

## UNIT - IV

### LASERS AND FIBRE OPTICS (CO 4)

Introduction, characteristics of a laser beam, spontaneous and stimulated emission of radiation, population inversion, Ruby laser, He-Ne laser, Semiconductor laser, applications of lasers (Theory only). Introduction to optical fibers, principle of propagation of light in optical fibers, Acceptance Angle and cone of a fibre, Numerical aperture (Problems based on acceptance angle and numerical aperture),

Modes of propagations, classification of fibers, Fibre optics in communications, Application of optical fibers, fiber optic sensors.

**MODERN PHYSICS (CO. 5)**

**UNIT - V**

Introduction, De Broglie concept of matter waves, Heisenberg uncertainty principle, Schrodinger time dependent and independent wave equations, physical significance of wave function and its properties, application for particle in a one dimensional well – energy eigen values and eigen functions of the particles (No problems).Energy band theory of crystals, classification of conductors, semiconductors and insulators. (Theory only). **Nanophase Materials :**Introduction, properties, Top-down and bottom up approaches, Synthesis - Ball milling, Chemical vapour deposition method, sol-gel methods, Applications of nanomaterials. (Theory only).

**Text Books:**

1. Physics by David Halliday and Robert Resnick – Part I and Part II - Wiley.
2. A textbook of Engineering Physics, Dr. M. N. Avadhanulu, Dr. P.G. Kshirsagar - S. Chand
3. Engineering Physics by R.K. Gaur and S.L. Gupta –DhanpatRai

**Reference Books:**

1. Modern Engineering Physics by A.S. Vadudeva
2. University Physics by Young and Freedman

# GAYATRI VIDYA PARISHAD COLLEGE FOR DEGREE AND PG COURSES (AUTONOMOUS)
# ENGINEERING AND TECHNOLOGY PROGRAM
# RUSHIKONDA CAMPUS, VISAKHAPATNAM–530045.

| Course code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int. | Ext | |
| CS1203 | Data Structures using 'C' | 3 | 0 | 0 | 30 | 70 | 3 |

**COURSE OBJECTIVES:**

1. To understand recursive algorithms and basic concepts of data structures.
2. To learn linear data structures such as Stacks, Queues and Linked lists.
3. To learn Nonlinear data structures such as Trees and Graphs.
4. To understand and solve searching and sorting techniques.
5. To solve problems using data structures such as stacks, queues, linear lists, trees and graphs.

**COURSE OUTCOMES:**

At the end of the course student will be able to

1. Understand the implementation of Stacks and Queues using Arrays and their applications.
2. Describe various types of linked lists and their implementation.
3. Construct various types of trees and their traversal techniques.
4. Discuss the computational efficiency of the principal algorithms for sorting and searching.
5. Describe how graphs are represented in memory and solve real time application problems using concepts of graphs.

## UNIT – I

**Introduction to Data Structures:** Abstract Data Types, Meaning and Definition of Data Structures. **Stacks:** Stack as an Abstract Data Type, Primitive Operations, Implementing Stack Operations using Arrays. Infix to Postfix, Infix to Prefix Conversions, Postfix Evaluation and Recursion. **Queues:** Queue as an Abstract Data Type, Sequential Representation, Types of Queues, Operations, Implementation using Arrays.

## UNIT – II

**Linked List:** Operations, Implementation of Stacks, Queues and priority Queues using Linked Lists, Circular Lists: Insertion, Deletion and Concatenation Operations, Stacks and Queues as Circular Lists, Doubly Linked Lists.

## UNIT- III

**Trees**: Binary Trees - Definitions and Operations, Binary Tree Representation: Node Representation, Implicit array Representation, Binary Tree Traversal, Threaded Binary Trees and their Traversal, Trees and their Applications; Heterogeneous binary trees, Tree Searching Insertion and Deletion of a node from a Binary Search Tree, Efficiency of Binary Search Tree operations.

## UNIT –IV

**Searching:** Basic Searching Techniques: Dictionary as an Abstract Data Type, Algorithmic Notation, Sequential Searching and its Efficiency, Binary Search, Interpolation Search.
**Sorting:** General Background: Efficiency, Asymptotic Notations, Efficiency of Sorting, Bubble Sort and Quick Sort and their Efficiency, Selection Sorting, Binary Tree Sort, Heap Sort, Insertion Sorts, Shell Sort, Address calculation Sort, Merge and Radix Sorts.

## UNIT – V

**Graphs and Their Application:** Definition of Graphs, Representation of Graphs, Transitive closure, Linked Representation of Graphs, Topological Ordering of nodes, Graph Traversal and Spanning Forests, Undirected Graphs and their Traversals Applications of Graphs, Minimal Spanning Trees.

**Text Books:**

1. Data Structures Using C and C++ Yaddish Langsam, Moshe J .Augenstein and Aaron M. Tanenbaum,Prentice Hall Of India(2ndEdition)
2. Data Structure and Algorithm, Prof. Maria Rukadikar S

**Reference Book:**

1. Data Structures, Algorithms and Applications with C++, Sahani Mc-Graw Hill.

# GAYATRI VIDYA PARISHAD COLLEGE FOR DEGREE AND PG COURSES (AUTONOMOUS)
# ENGINEERING AND TECHNOLOGY PROGRAM
# RUSHIKONDA CAMPUS, VISAKHAPATNAM–530045.

| Course code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int. | Ext | |
| CS1204 | Engineering Graphics | 1 | 0 | 4 | 30 | 70 | 3 |

## COURSE OBJECTIVES

1. The course is aimed at developing Basic Graphic skills.2:
2. Develop Skills in Preparation of Basic Drawings
3. Skills in Reading and Interpretation of Engineering Drawings

## COURSE OUTCOMES

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

CO 1: Graphically construct and understand, the importance of mathematical curves in Engineering applications

CO 2: Graphically visualize and construct orthographic projection of points and lines.

CO 3: Visualize and construct different views of planes and solids in different orientations

CO 4: Construct and develop the sectioned surfaces of geometrical solids

CO 5: Interpret and draw the Orthographic and Isometric views of different solids.

## UNIT-I

**Introduction**: Lines, Lettering and Dimensioning, Geometrical Constructions.

**Curves**: Construction of Conic sections by using general method - Normal and tangent to the curves.

## UNIT – II

**Projections of Points and Straight Lines**: Principal or Reference Planes - Projections of a point lying in any one of the four quadrants. Projections of straight lines parallel to both reference planes - perpendicular to one reference plane and parallel to other reference plane - inclined to one planeand parallel to the other.

## UNIT – III

**Projections of Planes**: Projection of Perpendicular planes: Perpendicular to both reference planes, perpendicular to one reference plane and parallel to other - perpendicular to one reference plane and inclined to other plane.

## UNIT – IV

**Projections of Solids**: Types of solids: Polyhedral and Solids of revolution. Projections of solids in simple positions: Axis perpendicular to horizontal plane - vertical plane -parallel to both the reference planes - Projection of Solids with axis inclined to one reference plane and parallel to other.

## UNIT – V

**Isometric Views**: Isometric projection - Isometric scale and Isometric view. Isometric view of Prisms, Pyramids, cylinder, cone, sphere .

**Text Books**:

Elementary Engineering Drawing by N.D. Bhatt, Charotar Publishing House.

**Reference Books:**

Engineering Graphics by K.L. Narayana and P. Kannaiah, Tata Mc-Graw Hill.

| Course code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int. | Ext | |
| CS1205 | Discrete Mathematical Structures | 3 | 0 | 0 | 30 | 70 | 3 |

**COURSE OBJECTIVES:**

1. To understand mathematical arguments using logical connectives and quantifiers and verify the validity of logical flow of arguments using propositional, predicate logic and truth tables.
2. To understand about permutations and combinations.
3. To understand various types of relations and discuss various properties of the relations.
4. To study the graphs, graph isomorphism and spanning trees.

**COURSE OUTCOMES:**

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

CO1: Solve the basic principles of Logics and proofs.
CO2: Solve different kinds of problems related to Relations and set theory.
CO3: Analyze the fundamental algorithms and construct simple mathematical proofs.
CO4: Acquire knowledge to solve network problems using graph theory.
CO5: Solve problems related to counting and advanced counting techniques.

**UNIT-I**

**The Foundations-Logic and Proofs:** Propositional Logic, Propositional Equivalences, Predicates and Quantifiers, Nested Quantifiers Rules of Inference, Introduction to Proofs, Proof Methods andStrategy.

**UNIT-II**

**Basic Structures-Sets, Functions, Sequences and Sums:** Sets, Set Operations, Functions, Sequences and Summations. Relations: Relations and their properties, n-ary relations, applications,Representation, closure equivalence relations, Partial orderings.

**UNIT-III**

**The Fundamentals-Algorithms, the Integers and Matrices:** Algorithms, The Growth of Functions, Complexity of Algorithms, The Integers and Division, Primes and Greatest Common Devisors, Integers and Algorithms, Applications of Number Theory, Matrices. Induction And Recursion: Mathematical Induction, Strong Induction and Well- Ordering, Recursive Definitions and Structural Induction, Recursive Algorithms, Program Correctness.

## UNIT-IV

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

## UNIT-V

**Counting:** The Basics of Counting, the Pigeonhole Principle, Permutations and Combinations. Advanced Counting Techniques: Recurrence Relations, Solving Linear Recurrence Relations, Divide-and-Conquer Algorithms and Recursion Relations, Generating Functions, Inclusion- Exclusion, and Applications of Inclusion-Exclusion.

**Text Book:**

1. Discrete Mathematics & Its Applications with Combinatorics and Graph Theory by Kenneth H Rosen, Tata McGraw-Hill Publishing Company Ltd., New Delhi.

**Reference Books:**

1. Discrete Mathematics for Computer Scientists & Mathematicians by Joe L. Mott, Abraham Kandel, Theodore P. Baker, Prentice-Hall, India
2. Discrete Mathematics by Richard Johnson Baug, Pearson Education, New Delhi.
3. Discrete and Combinatorial Mathematics by Ralph. G. Grimaldi, Pearson Education, NewDelhi.

| Course code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int. | Ext | |
| CS1206 | IT Workshop LAB | 0 | 0 | 3 | 50 | 50 | 1.5 |

## COURSE OBJECTIVES

1. Explain the internal parts of a computer, peripherals, I/O ports, connecting cables.
2. Demonstrate basic command line interface commands on LINUX.
3. Teach the usage of Internet for productivity and self-paced lifelong learning.
4. Describe about Compression, Multimedia and Antivirus tools.
5. Demonstrate Office Tools such as Word processors, Spreadsheets and Presentation tools.

## COURSE OUTCOMES

CO1: Assemble and disassemble components of a PC
CO2: Construct a fully functional virtual machine, Summarize various LINUX operating system commands.
CO3: Able to Troubleshoot hardware and software problems.

### Module I – Hardware Concepts

1. Every student should identify the peripherals of a computer, components in a CPU and its functions. Draw the block diagram of the CPU along with the configuration of each peripheral and submit to your instructor. Every student should disassemble and assemble the PC back to working condition.

### Module II – Software Installations

1. Every student should individually install operating system like LINUX or MS windows on the personal computer. The system should be configured as dual boot with both windows and LINUX.
2. Hardware Troubleshooting: Students have to be given a PC which does not boot due to improper assembly or defective peripherals. They should identify the problem and fix it to get the computer back to working condition.
3. Software Troubleshooting: Students have to be given a malfunctioning CPU due to system software problems. They should identify the problem and fix it to get the computer back to working condition.
4. Cyber Hygiene: Students should learn about viruses on the internet and install antivirus software. Student should learn to customize the browsers to block pop ups, block active x downloads to avoid viruses and/or worms.
5. Web Browsers, Surfing the Web: Students customize their web browsers with the LAN proxy

settings, bookmarks, search toolbars and popup blockers.

**Module III – MS-Office**

1. MS Word - Features to be covered: Formatting Fonts in word, Drop Cap in word, Applying Text effects, Using Character Spacing, Borders and Colours, Inserting Header and Footer, Using Date.
2. Creating project abstract Features to be covered: Formatting Styles, inserting table, Bullets and Numbering, Changing Text Direction, Cell alignment, Footnote, Hyperlink, Symbols, Spell Check, Track Changes.
3. Creating a Newsletter: Features to be covered: Table of Content, Newspaper columns, Images from files and clipart, Drawing toolbar and Word Art, Formatting Images, Textboxes, Paragraphs in word.
4. Spreadsheet Orientation: Accessing, overview of toolbars, saving spreadsheet files, Using help and resources. Creating a Scheduler: Gridlines, Format Cells, Summation, auto fill, Formatting Text.
5. Calculating GPA - Features to be covered: Cell Referencing, Formulae in spreadsheet – average, std. deviation, Charts, Renaming and Inserting worksheets, Hyper linking, Count function, Sorting, Conditional formatting.
6. Creating Power Point: Student should work on basic power point utilities and tools in Latex and MS-Office/equivalent (FOSS) which help them create basic power point presentation. PPT Orientation, Slide Layouts, Inserting Text, Formatting Text, Bullets and Numbering, Auto Shapes, Lines and Arrows, Hyperlinks, Inserting Images, Tables and charts.

| Course code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| CS1207 | Engineering Physics Lab | L | T | P | Int. | Ext | 1.5 |
| | | 0 | 0 | 3 | 50 | 50 | |

**COURSE OBJECTIVES**

To train the student in acquiring skills, techniques of using instruments to observe the physical phenomena, to measure certain physical quantities and constants.

**COURSE OUTCOMES**

By the end of the course

CO1: Experiment and evaluate basic principles of physics by observing and analyzing the data, plotting graphs and interpreting the results.

**List of Laboratory Experiments:**

1. Determination of Radius of Curvature of a given Convex Lens By forming Newton‘s Rings.
2. Determination of Wavelength of Spectral Lines in the Mercury Spectrum by Normal Incidence method.
3. Study the Intensity Variation of the Magnetic Field along axis of Current Carrying Circular Coil.
4. Determination of Cauchy‘s Constants of a Given Material of the Prism using Spectrometer.
5. Determination of Refractive Index of Ordinary ray μo and Extraordinary μe ray.
6. Determination of Thickness Given Paper Strip by Wedge Method.
7. Calibration of Low Range Voltmeter.
8. Calibration of Low Range Ammeter.
9. Determination of Magnetic Moment and Horizontal Component of Earth‘s Magnetic Field.
10. Lees Method - Coefficient of thermal Conductivity of a Bad Conductor.
11. Carey Foster‘s Bridge – Verification of laws of Resistance and Determination of Specific Resistance.
12. Melde‘s Apparatus – Frequency of electrically maintained Tuning Fork.
13. Photoelectric cell-Characteristics.
14. Planks Constants.
15. Determination of energy band gap of a given semiconductor.

| Course code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int. | Ext | |
| CS1208 | Data Structures LAB | 0 | 0 | 3 | 50 | 50 | 1.5 |

**COURSE OBJECTIVES:**

1. To implement stacks and queues using arrays and linked lists.
2. To develop programs for searching and sorting algorithms.
3. To write programs using concepts of various trees.
4. To implement programs using graphs.

**COURSE OUTCOMES:**

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

Implement programs on stacks, queues and various types of linked list.

1. Develop programs using various graph algorithms
2. Implement program on Binary search tree traversals
3. Write programs using various searching and sorting techniques.

**List of Programs:**

1. Write a C program to implement the operations on stacks.
2. Write a C program to implement the operations on circular queues.
3. Write a C program for evaluating a given postfix expression using stack.
4. Write a C program for converting a given infix expression to postfix form using stack.
5. Write a C program for implementing the operations of a dequeue.
6. Write a C program for the representation of polynomials using circular linked list and for the addition of two such polynomials.
7. Write a C program to create a binary search tree and for implementing the in order, preorder, post order traversal using recursion
8. a) Write a C program for finding the transitive closure of a digraph
   b) Write a C program for finding the shortest path from a given source to any vertex in a digraph using Dijkstra's algorithm
9. a) Write a C program for finding the Depth First Search of a graph.
   b) Write a C program for finding the Breadth First Search of a graph.
10. Write a C program for sorting a list using Bubble sort and then apply binary search.
11. Write a C program for quick sort
12. Write a C program for Merge sort.
13. Write a C program for Heap sort.

**Reference Books:**

1. Data Structures Using C and C++ Yaddish Langsam, Moshe J .Augenstein and Aaron M.Tanenbaum, Prentice Hall Of India(2ndEdition)
2. Data Structure and Algorithm, Prof. Maria RukadikarS
3. Data Structures, Algorithms and Applications with C++, Sahani Mc-Graw Hill.

# B. Tech Computer Science and Engineering (AI and ML)
# (R-22 Regulation)

### II Year – I Semester

| Course code | Category | Course Title | Hours per week | | | Internal Marks | External Marks | Total Marks | Credits |
|---|---|---|---|---|---|---|---|---|---|
| | | | L | T | P | | | | |
| CSM2101 | ES | Digital Logic Design & Computer Architecture | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM2102 | PC | Artificial Intelligence – I | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM2103 | HSS | Managerial Economics & Financial Accounting | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM2104 | PC | Operating Systems | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM2105 | PC | Object Oriented Programming Through Java | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM2106 | PC | Artificial Intelligence Lab | 0 | 0 | 3 | 50 | 50 | 100 | 1.5 |
| CSM2107 | PC | Object Oriented Programming Through Java Lab | 0 | 0 | 3 | 50 | 50 | 100 | 1.5 |
| CSM2108 | PC | Operating Systems Lab | 0 | 0 | 3 | 50 | 50 | 100 | 1.5 |
| CSM2109 | SC | Skill Course – 1 Python Programming | 1 | 0 | 2 | 50 | 50 | 100 | 2 |
| CSM2110 | MC | Environmental Science | 3 | 0 | 0 | 30 | 70 | 100 | 0 |
| **Total Credits** | | | | | | | | | **21.5** |

# GAYATRI VIDYA PARISHAD COLLEGE FOR DEGREE AND PG COURSES (AUTONOMOUS)
# ENGINEERING AND TECHNOLOGY PROGRAM
# RUSHIKONDA CAMPUS, VISAKHAPATNAM–530045.

| Course Code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int. | Ext | |
| CSM 2101 | **Digital Logic Design and Computer Architecture** | **3** | **0** | **0** | **30** | **70** | 3 |

**COURSE OBJECTIVES:**

1. To learn the basic principles for design of combinational circuits and sequential circuits.
2. Understanding the hierarchical organization of a computer system which consists of an instruction set of commands.
3. Learn about the architecture of a computer from a programming view.

**COURSE OUTCOMES:**

At the end of the course student will be able to
CO1: Understand Digital Circuits and Boolean Algebra and Minimize the Boolean expression using Boolean algebra and design it using logic gates.
CO2: Realize and simplify Boolean Algebraic functions using K-Maps and design combinational circuits.
CO3: Design and develop sequential circuits
CO4: Learn Computer Organization and Design & Micro-programmed Control
CO5: Understand Central Processing Design and able to Discuss the design concepts of pipeline and vector processing

## UNIT-I

**Introduction:** Digital Signals, Digital Circuits, AND, OR , NAND, NOT, NOR and Exclusive-OR Operations, Boolean Algebra and Examples of IC Gates, Number Systems: Binary, Signed Binary, Octal, Hexadecimal Number, Binary Arithmetic, one's and two's complements arithmetic, codes, error detecting and correcting codes.

## UNIT-II

**Combinational Digital Circuits** Standard Representation for logic functions, K-Map Representation, simplification of logic functions using K-Map, Minimization of Logical Functions, Don't Care Conditions, Multiplexer, Demultiplexer/Decoders, Adders, Subtractors, BCD Arithmetic, Carry look

ahead adder, serial adder, ALU, elementary ALU design, parity checker/generator.

**UNIT-III**

**Sequential Circuits and Systems** 1-bit memory, the circuit properties of Bistable latch, Clocked S-R Flip-flop, J-K, T and D-types Flip Flops, applications of flip flops, shift registers, applications of shift registers, ripple (Asynchronous) counters, synchronous counters, Counters design using Flip Flops.

**UNIT-IV**

**Computer Organization and Design:** Instruction Codes, Computer Registers, Computer Instructions, Timing and Control, Instruction Cycle, Memory-Reference Instructions, Input Output and Interrupt, Complete Computer Description, Design of Basic Computer, Design of Accumulator Logic. **Micro programmed Control:** Control Memory, Address Sequencing, Microprogram Example, Design of Control Unit.

**UNIT-V:**

**Central Processing Unit** Introduction, General Register Organization, Stack Organization, Instruction Formats, Addressing Modes, Data Transfer and Manipulation, Program Control, Reduced Instruction Set Computer (RISC), Architecture and Programming of 8085Microprocessor**. Pipeline and Vector Processing:** Parallel Processing, Pipelining, Arithmetic Pipeline, Instruction Pipeline, RISK Pipeline, Vector Processing, Array Processors.

**TEXT BOOKS:**

1. Digital Logic Design and Microprocessors, Dr. Narendra S. Jadhav , Dr. (Mrs.) Alpana P. Adsul, ISBN: 9789354517242.
2. Digital Design, 3rdEdition, M.Morris Mano, Pearson Education
3. Computer System Architecture, M. Morris Mano, Prentice Hall of India Pvt. Ltd. Third Edition Sept. 2008.
4. Microprocessor Architecture, Programming and Applications with the 8085 by Ramesh S Gaonkar

.

**REFERENCE BOOKS:**

1. Digital Logic and Microprocessor Design with VHDL, by Enoch Hwang
2. Computer Architecture and Organization, William Stallings, PHI Pvt. Ltd., Eastern Economy Edition, Sixth Edition, 2003.
3. Computer Organization and Architecture, Linda Null, Julia Lobur, Narosa Publications ISBN81-7319-609-5

# GAYATRI VIDYA PARISHAD COLLEGE FOR DEGREE AND PG COURSES (AUTONOMOUS)
# ENGINEERING AND TECHNOLOGY PROGRAM
# RUSHIKONDA CAMPUS, VISAKHAPATNAM–530045.

| Course Code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| CSM2102 | **Artificial Intelligence - I** | L | T | P | Int. | Ext | 3 |
| | | **3** | **0** | **0** | **30** | **70** | |

**Course Objectives :**

To learn about AI applications, intelligent agents and Heuristic Functions.

1. To understand the importance of search and the corresponding search strategies for solving AI problems.
2. To introduce to Knowledge based agents, knowledge representation and planning.

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

CO1: Demonstrate various AI applications, languages and Intelligent Agents. CO2:
Solve problems using uninformed and informed search strategies
CO3: Make use of local and backtrack search techniques in constraint satisfaction problems
CO4: Apply propositional logic techniques for knowledge representation.
CO5: Utilize the algorithms and their heuristics in the planning problems

## UNIT-I

Introduction to Artificial Intelligence, Foundations of Artificial Intelligence, History of Artificial Intelligence, Intelligent Agents : Agents and Environments, concept of Rationality, Nature of Environments, Structure of Agents.

## UNIT-II

Problem-solving: Problem-Solving Agents, Example Problems, Search Algorithms, Uninformed Search Strategies, Informed Search Strategies: greedy best first search, A* search, Heuristic functions.

## UNIT-III

Adversarial Search and Games: Game Theory, Optimal Decisions in Games, Heuristic Alpha– Beta Tree Search, Stochastic Games, Limitations of Game Search Algorithms, Constraint Satisfaction Problems: Defining Constraint Satisfaction Problems, Constraint Propagation: Inference in CSPs, Backtracking Search for CSPs, Local Search for CSPs, The Structure of Problems.

## UNIT-IV

Knowledge-Based Agents, The Wumpus World, Logic , Propositional Logic: A Very Simple Logic, Propositional Theorem Proving, First-Order Logic : Syntax and Semantics of First-Order Logic, Using First-Order Logic , Inference in First-Order Logic : Propositional vs. First-Order Inference, Unification and First- Order Inference, Forward Chaining, Backward Chaining, Resolution

## UNIT-V

Knowledge Representation: Ontological Engineering, Categories and Objects, Events, Mental Objects and Modal Logic, Reasoning Systems for Categories, Automated Planning: Definition of Classical Planning, Algorithms for Classical Planning, Heuristics for Planning, Hierarchical Planning, Planning and Acting in Nondeterministic Domains.

**Text Books**

1. Stuart J. Russell and Peter Norvig, *Artificial Intelligence A Modern Approach*, FourthEdition, Pearson, 2020

**Reference Books**

1. Dr.Nilakshi Jain, *Artificial Intelligence: Making a System Intelligent*, Wiley Publications,1st Edition,2019.
2. Elaine Rich, Kevin Knight and Shivashankar B. Nair, *Artificial Intelligence*, ThirdEdition, McGrawHill, 2017

# GAYATRI VIDYA PARISHAD COLLEGE FOR DEGREE AND PG COURSES (AUTONOMOUS)
# ENGINEERING AND TECHNOLOGY PROGRAM
# RUSHIKONDA CAMPUS, VISAKHAPATNAM–530045.

| Course Code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int. | Ext | |
| CSM2103 | **Managerial Economics & Financial Accounting** | **3** | **0** | **0** | **30** | **70** | 3 |

**COURSE OBJECTIVES:**

**The objective of this course is to:**

1. To understand the concepts of managerial economics and familiar with demand concepts, types of methodsor techniques of demand those are used by the entrepreneur or producer.
2. To have a thorough knowledge on the production theories and cost while dealing with the production and factors of production. To introduce the concepts of cost and significance, limitation of Break even analysis.
3. To understand how to start a business by using different forms of business organizations.
4. To have a knowledge about how to record business transactions and books by using Accounting conceptsand conventions, journal, ledger and other accounting records.
5. To assess the sources of funds and the financial position of the business by using common and comparativebalance sheets.

**COURSE OUTCOMES:**

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

CO1: Adopt the Managerial Economic concepts for decision making and forward planning. Also know law ofdemand and its exceptions, to use different forecasting methods for predicting demand for various products and services.

CO2: To assess the functional relationship between Production and factors of production and list out various costs associated with production and able to compute breakeven point to illustrate the various uses of breakeven analysis.

CO3: To outline the different types of business organizations and their registration process. CO4:

To adopt the principles of accounting to record, classify and summarize the accounts.

CO5: To plan about the sources of funds for business and the implementation of common size and comparative balance sheets for assessing the financial position of the business.

**UNIT-I**

INTRODUCTION TO MANAGERIAL ECONOMICS & DEMAND Definition, Nature and Scope of Managerial Economics. Demand Analysis: Definition-types of demand – Demand Determinants, Law of Demand and its exceptions. Elasticity of Demand: Definition, Types, Significance of Elasticity of Demand. Demand Forecasting: definition, methods of demand forecasting (survey methods, statistical methods, expert opinion method, test marketing, controlled experiments, judgmental approach to demand forecasting)

**UNIT-II**

THEORY OF PRODUCTION AND COST ANALYSIS Production Function – Law of Variable Proportion, Isoquants and Isocosts, MRTS, Cobb-Douglas Productionfunction, Laws of Returns. Cost Analysis: Types of Cost, Break-even Analysis (BEA)- Determination of Break-Even Point (Simple numerical problems) - Managerial Significance and limitations of BEA.

**UNIT-III**

BUSINESS ENVIRONMENT- Features of Business Organization, Sole Proprietorship, Partnership and Joint Stock Company, Steps for formation and Registration of the company.

**UNIT-IV**

INTRODUCTION TO FINANCIAL ACCOUNTING- Introduction to Accounting: Accounting Principles, Concepts & conventions, Double-Entry Book Keeping, Journal, Ledger and Trial Balance.

**UNIT-V**

PREPARATION AND ANALYSIS OF FINANCIAL STATEMENTS- Introduction to Sources of Finance: Equity shares, Preference shares, debentures, long term loans and Retained Earnings: Financial statement Analysis: advantages, Comparative and Common Size Balance Sheets Statements

**TEXT BOOKS:**

1. A R Arya sri, “Managerial Economics and Financial Analysis”, 4th Edition, TMH Publication, 2012.
2. S A Siddiqui & A. S. Siddiqui “Managerial Economics and Financial Analysis”,1st Edition, New Age Publishers, 2012.

**REFERENCE BOOKS:**

1. Dominick Salvatore, "*Managerial Economics: Principles and Worldwide Applications*", 7th edition, Oxford University Press, 2012.
2. N Ramachandran, Ram Kumar Kakani, "*Financial Accounting for Management*", 2nd Edition, Pearson Education, 2007.
3. D N Dwivedi, "*Managerial Economics*", 8th Edition, PHI Publication, 2010.

| Course Code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int | Ext | |
| CSM2104 | **Operating Systems** | **3** | **0** | **0** | **30** | **70** | 3 |

**COURSE OBJECTIVES:**

1. To learn about operating system structure, services, operations and design principles.
2. To understand how processes are scheduled and synchronized by Operating System.
3. To learn different OS approaches to memory management and deadlocks.
4. To learn design and implementation of OS subsystems such as File Systems, I/O Systems.

**COURSE OUTCOMES:**

At the end of the course student will be able to

CO1: Understand Operating System structure, classify OS services and analyze scheduling algorithms.
CO2: Identify solutions to overcome synchronization problems and deadlocks in modern operating system design.
CO3: Explain about memory management functions and compare various page replacement algorithms.
CO4: Understand how File Systems and I/O Systems are organized, implemented and managed by operating system.
CO5: Understand how OS recognize and protects system from unauthorized access.

**UNIT-I**

**Introduction to Operating Systems**: Over View of Operating Systems, Types of Operating Systems, Operating System Structures, Operating System Services, System Calls, Virtual Machines, Operating System Design and Implementation. **Process Management**: Process, Process Control Block, Process States, Operations on Processes, Cooperating Processes, Threads, Inter Process Communication, Process Scheduling, Scheduling Algorithms.

**UNIT-II**

**Process Synchronization:** The Critical Section Problem, Peterson ‘s Solution, Synchronization Hardware, Semaphores, Classical Problems of Synchronization, Critical Regions, Monitors. **Deadlocks**: System Model, Deadlock Characterization, Methods for Handling Deadlocks,

Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, Recovery from Deadlocks.

**UNIT-III**

**Memory Management:** Logical versus Physical Address, Swapping, contiguous memory allocation, paging, structure of the page table, segmentation. **Virtual Memory**: Demand Paging, Page Replacement, Allocation of Frames, Thrashing.

**UNIT-IV**

**File Systems: Implementation and Secondary-Storage Structure:** Concept of a file, Access Methods, Directory Structure, Protection, File System Structure, Allocation Methods, Free Space Management.

**UNIT-V**

**I/O systems**: Overview of Mass-storage structure, Disk structure, Disk attachment, Disk scheduling. **System Protection:** Goals of Protection, Principles of Protection, Domain of Protection, Access Matrix, Implementation of Access Matrix, Access Control, Revocation of Access Rights.

**TEXT BOOKS**:

1. Operating Systems, Abraham Silber schatz, Peter Baer Galvin, and Greg Gagne, John Wiley Publ., 9th Edition.

**REFERENCE BOOKS:**

1. Modern Operating Systems, Andrew S. Tanenbaum, 4th edition, 2016, Pearson.
2. Operating Systems, William Stallings 5th Edition –PHI.
   Operating Systems: A Design-Oriented Approach ‘, Charles Crowley, Tata Hill Co., 1998 Edition.

| Course Code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int. | Ext | |
| CSM2105 | **Object-Orientated Programming through Java** | **3** | **0** | **0** | **30** | **70** | 3 |

**COURSE OBJECTIVES:**

1. To learn the basics of object oriented programming concepts and Java programming.
2. To learn different types of Constructors and inheritance, Super Keyword, Method overriding and dynamic method dispatch.
3. To understand the concepts of packages and Interfaces.
4. To understand the concepts of Exception Handling and Multithreading.
5. To design concepts of real time problems using Graphical User Interface.

**COURSE OUTCOMES:**

At the end of the Course the student shall be able to

CO1: Implement object orientated programming strategies and Contrast classes and objects.
CO2: Analyze Constructors, Inheritance and Dynamic Method Dispatch.
CO3: Demonstrate various classes in different packages, design own packages and implementing interfaces.
CO4: Manage Exceptions and Apply Threads.
CO5: Create GUI screens along with event handling and write network programs.

**UNIT-I**

**Introduction to Objects & Classes:** OOP Principles, Java Buzz Words, The Byte Code, A First Simple Program. Class Fundamentals with Variables and Methods, Declaring objects for accessing variables and methods. Data Types and Variables, Operators and Expressions, Control Statements, Type Conversion and casting, Arrays: Single Dimension, Multi Dimension, command line arguments.

.

**UNIT-II**

**Constructors:** Default and Parameterized, this keyword and Garbage Collection, Overloading Methods, Overloading Constructors, Using objects as Parameters, Returning objects, String Handling.

**Inheritance:** Inheritance Basics, Types of Inheritance, Using Super keyword for constructors, Super to call variables and methods, Method Overriding, Dynamic Method Dispatch.

## UNIT-III

**Packages and Interfaces**: Defining a Package, importing a package, Package Example, Access Modifiers, Abstract class. **Interfaces:** Defining and Implementing Interfaces. **Exploring java.lang**: Wrapper classes, Object, Math, **Exploring java.util**: The collection framework: Array List, HashSet and Hash Map, String Tokenizer, Calendar, Random, Scanner. **Exploring java.io**: File class, Byte Streams, Character Streams, File Input Stream, File Output Stream, File Reader and File Writer classes.

## UNIT-IV

**Exception Handling**: Exception Handling Fundamentals, Exception Types, throw, throws and finally, Creating your own exceptions. **Multithreaded Programming**: Two ways of Creating a Thread, Creating Multiple Threads, isAlive(), join(), Synchronization.

## UNIT-V

**Introducing GUI Programming With Swings**: Swing Features, MVC Connection, Components and Containers, Panes, Simple Swing Application, Simple Swing Applet, Layout Managers: Flow, Border, Card, Grid, Grid Bag, Working with Color, Working with Fonts, Painting in Swing, Exploring Swing Components. **Delegation Event Mode**l: Event Classes, Sources and Listeners.

**TEXT BOOKS:**

1. Herbert Schildt, Java The complete reference, 11th Edition, Mc Graw Hill, 2019
2. Timothy budd, An introduction to object-oriented programming, 3rd Edition, Pearson, 2009.

**REFERENCE BOOKS:**

1. Cay S. Horstmann, Core Java Volume I–Fundamentals, 11th Edition, Pearson 2019
2. Y. Daniel Liang Introduction to Java Programming Comprehensive Version, 10thEdition, Pearson, 2015.
3. Bruce Eckel, Thinking in Java, 4th Edition, Prentice Hall, 2006

| Course code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| CSM2106 | **Artificial Intelligence Lab** | L | T | P | Int. | Ext | 1.5 |
| | | **0** | **0** | **3** | **50** | **50** | |

**COURSE OBJECTIVES:**

1) Understand the Programming Concepts of Prolog Software / C language
2) Describe the Programming features of LISP / C Language

**COURSE OUTCOMES:**

CO1: Analyze AI Programs in PROLOG/C

CO2: Analyze AI Programs in LISP / C

**List of Programs**

1. Study of Prolog.
2. Write simple fact for the statements using PROLOG.
3. Write predicates one converts centigrade temperatures to Fahrenheit, the other checks if a temperature is below freezing using Prolog.
4. Write a program to solve the Monkey Banana problem using Prolog.
5. Write a program in prolog for medical diagnosis and show the advantage and disadvantage of green and red cuts.
6. Write a program to implement factorial, Fibonacci of a given number.
7. Write a program to solve 4-Queen problem.
8. Write a program to solve traveling salesman problem.
9. Write a program to solve water jug problem using LISP.
10. Write a Lisp program to solve best first search traversal.

| Course Code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int. | Ext | |
| CSM2107 | **Object - Oriented Programming through Java lab** | **0** | **0** | **3** | **50** | **50** | 1.5 |

**COURSE OUTCOMES:**

At the end of the Course the student shall be able to

CO1**:** Write basic Java applications and use arrays

CO2**:** Demonstrate classes, objects and apply Inheritance

CO3**:** Implement Packages and build applications using default packages CO4**:** Illustrate Exceptions and develop multithreaded applications

CO5**:** Outline GUI applications which are event based and write network programs

**LIST OF PROGRAMS:**

**(Any 12 programs from the following to be performed)**

1) a) Implement the following programs using command line arguments and Scanner class

b) Accept two strings from the user and print it on console with concatenation of "and" in the middle of the strings.

c) To find the perimeter and area of a circle given a value of radius.

b) Write a program using classes and objects in java?

2) a) Write a program to call default constructor first and then any other constructor in the class?

b) Write a program that accepts an array of integers and print those which are both odd and prime. If no such element in that array print "Not found".

c) Write a program to accept contents into an Integer Array and print the frequency of each number in the order of their number of occurrences.

d) Write a program that accepts an „m x n" double dimension array, where „m" represents financial years and „n" represents Ids of the items sold. Each element in the array represents number of items sold in a particular year. Identify the year and id of the item which has more demand.

3) a) Create a class Box that uses a parameterized constructor to initialize the dimensions of a box. The dimensions of the Box are width, height, depth. The class should have a method that can return the volume of the box. Create an object of the Box class and test the functionalities.

b) Create a new class called Calculator with the following methods:

A static method called powerInt(int num1,int num2) This method should return num1 to the power num2.A static method called powerDouble(double num1,double num2). This method should return num1 to the power num2. Invoke both the methods and test the functionality. Also count the number of objects created.

4) a) Accept a String and a number „n" from user. Divide the given string into substring search of size „n" and sort them lexicographically.

b) Accept an array of strings and display the number of vowels and consonants occurred in each string.

c) Accept two strings from the user and determine if the strings are anagrams or not.

5) a) Create a multilevel inheritance for classes vehicle, brand and cost. The vehicle class determines the type of vehicle which is inherited by the class brand which determines the brand of the vehicle. Brand class is inherited by cost class, which tells about the cost of the vehicle. Create another class which calls the constructor of cost class and method that displays the total vehicle information from the attributes available in the super classes.

b) Create an inheritance hierarchy of Figure_3D, Cylinder, Cone, Sphere etc. In the base class provides methods that are common to all Figure_3Ds and override these in the derived classes to perform different behaviors, depending on the specific type of Figure_3D. Create an array of Figure_3D, fill it with different specific types of Figure_3Ds and call your base class methods.

6) a) Design a package to contain the class Student that contains data members such as name, roll number and another package contains the interface Sports which contains some sports information. Import these two packages in a package called Report which process both Student and Sport and give the report.

b) Write a program that accepts values of different data types and convert them to corresponding wrapper classes and display using the vector

7) a) Write a program to generate a set of random numbers between two numbers x1 and x2, and x1>0.

b) Write a program to implement a new Array List class. It should contain add(), get(), remove(), size() methods. Use dynamic array logic.

c) Create an employee class containing at least 3 details along with Id, setters, and getters. Insert the employee objects dynamically key as employee id and value as its corresponding object into a Hash Map. Perform Id based search operation on the Hash Map.

8) a) Write a program that reads file name from the user then displays information about that file, also read the contents from the file in byte stream to count the number of alphabets, numeric values, and special symbols. Write these statistics into another file using byte streams. Write a program that reads a CSV file containing a super market data containing product ID, Name, Cost and Quantity of sales and calculate the total revenue of the supermarket also sort the products in the order of their

demand.

b) Write a program that reads a text file containing some technical content and identify the technical terms and sort them alphabetically.

Note: use a file containing stop words (general English and Grammar terms as many as possible)

9) a) Write a program that reads two numbers from the user to perform integer division into Num1 and Num2 variables. The division of Num1 and Num2 is displayed if they are integers. If Num1 or Num2 were not an integer, the program would throw a Number Format Exception. If Num2 were Zero, the program would throw an Arithmetic Exception.

b) Create a user defined exception.

10) a) Write a program that creates 3 threads by extending the 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).

b) Write a program to illustrate Thread synchronization.

11) a) Create a JApplet that displays a message which is scrolling from left to right. b) Write a program that displays a sample registration page using Swing controls use appropriate layout managers.

b) Write a program for handling mouse events with adapter classes.

12) a) Create an interface containing 3 radio buttons named line, rectangle, and oval. Based on the radio button selected, allow user to draw lines, rectangles, or ovalsas per the locations selected by the user.

b) Write a program to create a Table inside a JFrame.

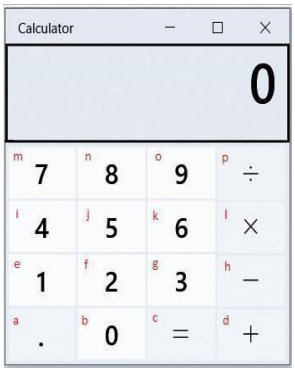
c) Create an interface that illustrates JFileChooser class and read CSV file containing employee data of various departments and display the records department wise on the interface.

13) a)Check all the fields filled or not, display success dialogue if all fields are filled with the help of Action Listener for program.

b). Display respective error dialogue if afield is empty.

14) Write a program to create three JSliders where each represents colors RED, GREEN and BLUE. Each slider has a value from 0 to 255. The background color of the applet is set based on the values retrieved from each slider to form a color using the color class constructor. On sliding any slider, the background color of the applet changes.

15) Complete the code to develop an ADVANCED CALCULATOR that emulates all the functions of the GUI Calculator as shown in the image.




**TEXT BOOKS:**

1. Herbert Schildt, *Java The complete reference*, 11th Edition, Mc Graw Hill, 2019
2. Timothy budd, *An introduction to object-oriented programming*, 3rdEdition, Pearson Education, 2009.

**REFERENCE BOOKS:**

1. Cay S. Horst mann, Core Java Volume I–Fundamentals, 11th Edition, Pearson 2019.
2. Y. Daniel Liang Introduction to Java Programming Comprehensive Version, 10th Edition, Pearson, 2015.
3. Bruce Eckel, Thinking in Java, 4th Edition, Prentice Hall, 2006.

**WEB REFERENCES:**

1. https://onlinecourses.nptel.ac.in/noc19_cs84/preview

| Course code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int. | Ext | |
| CSM2108 | **Operating Systems Lab** | **0** | **0** | **3** | **50** | **50** | 1.5 |

**COURSE OBJECTIVES:**

1. To learn about UNIX/LINUX operating system and system calls.
2. To understand and simulate the principles of resource management.
3. To understand UNIX/LINUX shell and its programming and vi editor.
4. To identify the data structures used for solving the problems related to synchronization, deadlocks and file allocation methods.

**COURSE OUTCOMES:**

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

CO1: Examine different Unix commands and Experiment programs using system calls.

CO2: Develop shell programs using vi editor

CO3: Employ various data structures to implement OS functions.

**LIST OF EXPERIMENTS**

**MODULE-I**

1. OS lab familiarization, Home Assignment on Unix commands, vi editor.
2. Simple C programs using command line arguments, system calls, library function calls.
3. C programs using fork system call to create processes and study parent, child process mechanism.
4. C programs to create process chaining, spawning.
5. C programs to handle errors using errno, perror() function.
6. C programs to use pipe system call for inter process communication.

**MODULE-II**

1. Familiarization of UNIX shell programming.
2. Simple shell programming exercises.
3. Shell programming using decision making constructs.
4. Shell programming using loop constructs.
5. Shell programming for file and directorymanipulation.

**MODULE-III**

1.C programs to study process scheduling implementing FCFS, Shortest Job First, and Round Robin algorithms.

2. C programs to study page replacement implementing FIFO, Optimal, and LRU page replacement algorithms.

3. C programs to study deadlock avoidance and detection.

4. Implement the Producer – Consumer problem using Semaphores.

**REFERENCE BOOKS:**

1. Unix concepts and applications by Sumitabha Das, TMH Publications.
2. Unix programming by Stevens, Pearson Education.
3. Shell programming by Yashwanth Kanetkar.
4. Operating System, Concepts by Silber schatz, and Peter Galvin.

| Course Code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int. | Ext | |
| CSM2109 | **Python Programming (Skill Course – 1)** | **1** | **0** | **2** | **50** | **50** | 2 |

**COURSE OBJECTIVES:**

1. Learn basic programming of Python.
2. To develop programs using Python packages.

**COURSE OUTCOMES:**

At the end of the course student will be able to

1: Develop the Python programs using operators, conditional and looping statements and strings.
2: Implement programs using functions and different types of Data structures.3: Develop the programs using Python Packages, OOP concepts.

**MODULE-I**

**Week 1: Introduction:**
History of Python, Need of Python Programming, Python Installation, Python basics.

**Week 2: Operators in python, conditional statements**

1. Accept two numbers from the user and calculate Addition, Subtraction, multiplication and Division.
2. Write a Program for checking whether the given number is an even number or not.
3. Given a two integer numbers return their product and if the product is greater than 1000, then return their sum.
4. A student will not be allowed to sit in exam if his/her attendance is less than 75%. Take following input from user - Number of classes held, Number of classes attended, and print percentage of class attended Is student is allowed to sit in exam or not.

**Week 3: Iterations, continue and break statements.**

1. Print the following pattern

```
1
1 2
1 2 3
1 2 3 4
```

2. Accept number from user and calculate the sum of all number between 1 and given number
3. Given a number count the total number of digits in a number

**Week 4: Strings, string functions, string slicing**

1. Given 2 strings, s1 and s2, create a new string by appending s2 in the middle of s1.
2. Given a string input Count all lower case, upper case, digits, and special symbols.
3. Given an input string, count occurrences of all characters within a string.

**MODULE-II**

**Week 5: Lists and Tuples**

1. Write a Python program to get the largest number and smallest number from a list.
2. Write a Python program to remove duplicates from a list.
3. Write a Python program to find the length of a tuple.
4. Write a Python program to convert a list to a tuple.

**Week 6: Sets and Dictionaries**

1. Dictionaries and dictionary methods, Sets and set methods.
2. Write a Python script to merge two Python dictionaries
3. Write a Python program to sort a dictionary by key
4. Return a set of identical items from a given two Python set

**Week 7: Functions:**

(**Defining** Functions, Calling Functions, Passing Arguments, Anonymous Functions, Fruitful Functions (Function Returning Values)

1. Write a Python program to reverse a string using functions
2. Write a Python function to check whether a number is perfect or not
3. Write a function unique to find all the unique elements of a list.

**Week 8: Recursion**

1. Write a Python program to get the factorial of a non-negative integer using Recursion
2. Write a Python program to solve the Fibon3a7cci sequence using recursion.

**Week 9: Regular expressions:**

Meta characters, Special Sequences, Sets, RegEx Function. File handling: modes, reading files, writing and closing files, Iterators, Generators, Filters and Lambda.

1. Write a Python program to find the substrings within a string
2. Write a Python program to Email id validation
3. Write a Python program to write a list to a file
4. Write a Python program to copy the contents of a file to another file

**MODULE-III**

**Week 10: Modules:**

1. Creating modules, import statement, from. Import statement, name spacing.
2. Python packages: Introduction to PIP, Installing Packages via PIP, Using Python Packages.
3. Install packages requests, flask and explore them. using (pip)
4. Write a script that imports requests and fetch content from the page. Eg. (Wiki)
5. Write a simple script that serves a simple HTTP Response and a simple HTMLPage

**Week 11: Basics of NumPy and Pandas packages, Basics of Matplotlib library.**

1. Add the following two NumPy arrays and modify a result array by calculating the square of each element.
2. Write a Python program to convert a dictionary to a Pandas series

**Week 12: OOP**

a) Class variables and instance variable
i) Robot
ii) ATM Machine.

**REFERENCE BOOKS:**

1. Head-First Python: A Brain-Friendly Guide (2nd Edition).
2. Python Programming: An Introduction to Computer Science (3rd Edition)
3. Fluent Python: Clear, Concise, and Effective Programming (1st Edition)
4. Programming Python: Powerful Object-Oriented Programming (4th Edition)

# GAYATRI VIDYA PARISHAD COLLEGE FOR DEGREE AND PG COURSES (AUTONOMOUS)
# ENGINEERING AND TECHNOLOGY PROGRAM
# RUSHIKONDA CAMPUS, VISAKHAPATNAM–530045.

| Course Code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int. | Ext | |
| CSM2110 | **Environmental Science** | **3** | **0** | **0** | **30** | **70** | 0 |

**COURSE OBJECTIVES:**

The objectives of the Environmental Science course are to

1. Familiarize the fundamental aspects of environment and the environmental management'
2. Make realize the importance of natural resources management for the sustenance of the life and the society.
3. Apprise the impact of pollution getting generated through the anthropogenic activities on the environment
4. Provide the concept of Sustainable Development, energy and environment.
5. Impart knowledge on the new generation waste like e-waste and plastic waste

**COURSE OUTCOMES:**

CO1: In this unit the students learn about the scope and importance of Environmental studies. The students understand about different kinds of ecosystems.
CO2: The students learn about biodiversity and its conservation. They also learn about types of biodiversity, values of biodiversity and threats to biodiversity.
CO3: The students understand about the types of natural resources and problems associated with them.
CO4: In this unit the students gain knowledge about different types of environmental pollution- causes, effects and control measures.
CO5: In this unit the students gain knowledge about characteristics of human population growth and its impact on environment. The students develop deep understanding about the environmental legislation.

**UNIT-I**

**Introduction to Environmental studies and Ecosystems**: Definition, Scope and importance of environmental studies. Concept of an Eco system, Biotic and Abiotic components of ecosystem, structure and function of an ecosystem. Food Chains, Food web and Ecological Pyramids. Forest ecosystem, Grassland ecosystem, Desert ecosystem, Pond ecosystem and Marine ecosystem.

**UNIT-II**

**Bio-Diversity and its Conservation:** Introduction – Definition and types of biodiversity – value of biodiversity - India as mega diversity nation – Hot 3S9pots of biodiversity – Threats to biodiversity – Conservation methods of biodiversity – In-situ & Ex – situ methods of conservation - Concept of sustainable development.

## UNIT-III

**Environment and Natural Resources Management:** Soil erosion and desertification, Effects of modern agriculture, fertilizer-pesticide problems, Forest Resources: Use and over-exploitation, Mining and dams – their effects on forest and tribal people, Water resources: Use and over-utilization of surface and ground water, Floods, droughts, Water logging and salinity, Dams – benefits and costs, Conflicts over water, Energy Resources: Energy needs, Renewable and non-renewable energy sources.

## UNIT-IV

**Environmental Pollution – climate change and environmental problems:** Definition, causes, effects and control measures of (a) air pollution (b) water pollution (c) soil pollution (d) noise pollution. Global Warming – Acid Rain – Ozone depletion – Photochemical smog. Drinking water, Sanitation and public health, Effect of Human activities of the quality of environment-Urbanization, transportation, Industrialization. Water scarcity and ground water depletion, Controversies on major dams – resettlement and rehabilitation of people problems and concerns. Concept of plastic waste and e-waste.

## UNIT-V

**Human Population and Environmental legislations:** Population Explosion – characteristics of population explosion. Impact of population growth on Environment – Role of Information technology in Environment and Human Health. Environmental Ethics. Environmental acts: Water (Prevention and control of pollution) act, air (Prevention and control of pollution) act, Environmental Protection Act, Wild life protection act, Forest conservation act.

## TEXT BOOKS:

1. Bharucha, Erach (2004). Textbook for Environmental Studies for Undergraduate Courses of all Branches of Higher Education, University Grants Commission, New Delhi.
2. Basu, M., Xavier, S. (2016). Fundamentals of Environmental Studies, Cambridge University Press, India
3. Masters, G. M., &Ela, W. P. (1991). Introduction to environmental engineering and science. Englewood Cliffs, NJ: Prentice Hall.
4. Enger, E. and Smith, B., Environmental Science: A Study of Inter relationships, Publisher: McGraw-Hill Higher Education; 12th edition, 2010.
5. Anubha Kaushik and C.P.Kaushik. Environmental Science by New age International Publishers.

# B. Tech Computer Science and Engineering (AI and ML)
# (R-22 Regulation)

### II Year – II Semester

| Course code | Category | Course Title | Hours per week | | | Internal Marks | External Marks | Total Marks | Credits |
|---|---|---|---|---|---|---|---|---|---|
| | | | L | T | P | | | | |
| CSM2201 | ES | Mathematical Foundations for Machine Learning | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM2202 | PC | Artificial Intelligence – II | 3 | 1 | 0 | 30 | 70 | 100 | 3 |
| CSM2203 | PC | Database Management Systems | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM2204 | PC | Design and Analysis of Algorithms | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM2205 | BS | Probability & Statistical Methods | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM2206 | PC | Neural Networks Lab | 0 | 0 | 3 | 50 | 50 | 100 | 1.5 |
| CSM2207 | PC | Database Management Systems Lab | 0 | 0 | 3 | 50 | 50 | 100 | 1.5 |
| CSM2208 | SC | Skill Course – 2 Design Thinking and Innovation | 1 | 0 | 2 | 50 | 50 | 100 | 2 |
| CSM2209 | MOOCS | Massive Open Online Courses (MOOCs) (NPTEL/COURSERA/ UDEMY) | 0 | 0 | 3 | 50 | 50 | 100 | 1.5 |
| CSM2210 | MC | NCC/NSS | 0 | 0 | 2 | - | - | - | 0 |
| **Total Credits** | | | | | | | | | **21.5** |
| **Summer Internship – I (Evaluation will be done in 3-1)** | | | | | | | | | |
| **Honors & Minors** | | | | | | | | | **4** |

# GAYATRI VIDYA PARISHAD COLLEGE FOR DEGREE AND PG COURSES (AUTONOMOUS)
# ENGINEERING AND TECHNOLOGY PROGRAM
# RUSHIKONDA CAMPUS, VISAKHAPATNAM–530045.

| Course code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int. | Ext | |
| CSM2201 | **Mathematical Foundations for Machine Learning** | **3** | **0** | **0** | **30** | **70** | **3** |

**Course Objectives:**

1. To provide students with a solid mathematical foundation necessary for understanding and applying machine learning algorithms effectively.
2. To introduce key mathematical concepts and techniques that are commonly used in various aspects of machine learning, including linear algebra, calculus, and probability.
3. To develop students' ability to analyze and solve mathematical problems related to machine learning applications.
4. To equip students with the mathematical skills needed to critically evaluate and contribute to research in the field of machine learning.

**Course Outcomes:**

Upon completing this course, students will be able to:

1. Apply linear algebra concepts to model and manipulate data in machine learning, including matrix operations, vector spaces, and eigenvalue/eigenvector analysis.
2. Utilize calculus techniques for optimization and gradient-based learning algorithms in machine learning applications.
3. Understand and apply probability and statistics concepts, including probability distributions, hypothesis testing, and Bayesian inference, to model uncertainty in machine learning problems.
4. Analyze and interpret mathematical notations and equations commonly used in machine learning research papers.
5. Develop mathematical problem-solving skills and the ability to create mathematical models for various machine learning tasks.

**UNIT-1:**

**Linear Algebra:** Vector Spaces, Linear Independence, Basis and Rank, Linear Mappings. **Analytic Geometry:** Norms, Inner Products, Lengths and Distances, Angles and Orthogonality, Ortho normal Basis, Orthogonal Complement, Inner Product of Functions, Orthogonal Projections.

**UNIT-2:**

**Matrix Decompositions:** Cholesky Decomposition, Singular Value Decomposition, Matrix Approximation.

**Vector Calculus:** Gradients of vector valued functions, Gradients of Matrices, Useful identities for computing gradients, Back propagation and Automatic Differentiation.

**UNIT-3:**



**Probability and Distributions:** Construction of a Probability space, Discrete and Continuous probabilities, sum rule, product rule and Bayes Theorem, Summary statistics and Independence, Gaussian Distribution.

**Continuous Optimization:** Optimization using Gradient Descent, Constrained optimization and Lagrange Multipliers, Convex Optimization.

**UNIT-4:**

**Linear Regression:** Problem Formulation, Parameter Estimation, Bayesian Linear Regression, Maximum Likelihood as Orthogonal Projection.

**Dimensionality Reduction with Principal** Component Analysis: Problem setting, Maximum Variance Perspective, Projection Perspective, Eigenvector computation and Low Rank Approximations, PCA in High Dimensions, Latent Variable Perspective.

UNIT-5

Density Estimation with Gaussian Mixture Models: Gaussian Mixture Model, Parameter Learning via Maximum Likelihood, EM Algorithm, Latent-Variable Perspective.

Classification with Support Vector Machines: Separating Hyper planes, Primal Support Vector Machine, Dual Support Vector Machine, Kernels, Numerical Solution.

**Text Books:**

1. Mathematics For Machine Learning by Marc Peter Deisenroth, A. Aldo Faisal and Cheng Soon Ong published in 2020, Cambridge university press.
2. Linear Algebra for Data Science, with Examples in R. By Shaina Bennett published in 2021.
3. The Elements of Statistical Learning by Hastie, Tibshirani and Friedmam Published in 2016 from Stanford university.

**Reference Books:**

1. Probabilistic Machine Learning by Kevin Murphy, MIT Press (2022).
2. Interpretable Machine Learning by Christoph Molnar, Published in 2022.
3. Machine Learning by Andrew Ng, Stanford university in 2020.
4. Model based Machine Learning by John Winn published in 2022.
5. Introduction to Mathematical Statistics by Robert Hogg, Joseph McKean and Allen Craig, in 2019, Published by Pearson.

| Course Code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| CSM2202 | **Artificial Intelligence – II** | L | T | P | Int. | Ext | **3** |
| | | **3** | **0** | **0** | **30** | **70** | |

**COURSE OBJECTIVES:**

1. Understanding Reasoning under Uncertainty
2. Learn Probabilistic Reasoning
3. Understanding Expert Systems, Fuzzy Logic and Advanced Knowledge Representation.

**COURSE OUTCOMES:**

CO1: Apply probabilistic reasoning to solve uncertainty problems.
CO2: Examine simple, complex and multi agent decision making.
CO3: Outline various slot and filler structures.
CO4: Develop Neuro fuzzy systems and Genetic models in the real world.
CO5: Illustrate applications of expert systems and robotics.

**UNIT-I**

Quantifying Uncertainty: Acting under Uncertainty, Basic Probability Notation, Inference using Full Joint Distributions, Bayes' Rule and its use. Probabilistic Reasoning: Representing Knowledge in Uncertain Domain, Semantics of Bayesian Networks, Efficient Representation of Conditional Distributions, Exact Inference in Bayesian Networks.

**UNIT-II**

Probabilistic Reasoning over Time: Time and Uncertainty, Hidden Markov Models, Dynamic Bayesian Networks. Decisions with Multiple Agents: Game Theory. Multi-Agent Decision Making: Properties of Multiagent Environments, Non-Cooperative Game Theory, Cooperative Game Theory, Making Collective Decisions

**UNIT-III**

Advanced Knowledge Representation: Weak Slot-and-Filler Structures: Semantic Nets, Frames, Strong Slot-and-Filler Structures: Conceptual Dependency, Scripts, Knowledge Representation Summary: Syntactic Semantic Spectrum of Representation, Logic and Slot-and-Filler Structures, Other Representational Techniques.

## UNIT-IV

Fuzzy Logic: Introduction, Fuzzy Sets, Fuzzy Terminology, Fuzzy Logic Control, Fuzzy Inference Processing, Neuro Fuzzy Systems. Genetic Algorithms: Introduction to GA, Significance of Genetic Operators, Termination parameters, Evolving Neural Networks.

## UNIT-V

Expert Systems: Definition – Features of an expert system, Organization, Characteristics, Prospector, Knowledge Representation in expert systems, Expert system tools-MYCIN-EMYCIN.

Introduction to Robotics: Robot Hardware, Robot Perception, Planning to move, Planning Uncertain movements, Moving, Robotic Software Architectures, Application Domains.

### TEXT BOOKS:

1. Stuart J. Russell and Peter Norvig, Artificial Intelligence: A Modern Approach, Fourth Edition, Pearson, 2020.

2. Elaine Rich, Kevin Knight and Shiva Shankar B. Nair, Artificial Intelligence, Third Edition, Mcgraw Hill, 2017.

### REFERENCE BOOKS:

1. Dr.NilakshiJain, Artificial Intelligence: Making System Intelligent, Wiley Publications, 1st Edition, 2019.

| Course Code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int | Ext | |
| CSM2203 | **Database Management Systems** | **3** | **0** | **0** | **30** | **70** | 3 |

**COURSE OBJECTIVES:**

1. Focus the role of a database management system in an organization and construct ER
2. Diagrams Demonstrate basic database concepts, including the structure and operation of the relational data model and basic database queries using SQL.
3. Applying advanced database queries using Structured Query Language (SQL)
4. Evaluating logical database design principles and database normalization.
5. Demonstrate the concept of a database transaction, concurrency control, and data object locking and protocols.

**COURSE OUTCOMES:**

After successful completion of the course, the student will be able to:
CO1: Understand database design principles.
CO2: Apply data Modelling using E-R diagrams.
CO3: Create refined data models using normalization.
CO4: Build database queries using Structured Query Language.
CO5: Understand the transaction management and concurrency control.

**UNIT-I**

**Introduction to DBMS and Database Design:** File system vs DBMS, advantages of DBMS, storage data, queries, DBMS structure, Types of Databases – Hierarchical, Network, Relational, Key-Value, Object Oriented, XML DB Overview of File Structures in database, **Data base Design**: data models, the importance of data models. **E-R model:** Entities, attributes and entity sets, relationship and relationship set, mapping cardinalities, keys, features of ER model, conceptual database design with ER model.

**UNIT-II**

**Relational Model and Basic SQL Relational model:** Integrity constraints over relations and enforcement, querying relation data, logical database design, views, destroying/altering tables and views. **Basic SQL:** Introduction to SQL, Basic SQL Queries: DML, DDL, DCL, TCL.

## UNIT - III

**Structured Query Language (SQL):** Select Commands, Union, Intersection, Except, Nested Queries, Aggregate Operators, Null values, Relational set operators, SQL join operators. **Relational Algebra(RA):** Selection, Projection, Set operations, Joins Relational Calculus (TRC, DRC): Tuple Relational Calculus, Domain Relational Calculus PL/SQL, Assertions, Triggers.

## UNIT - IV

**Schema Refinement and Normal Forms:** Introduction to Schema Refinement, Functional Dependencies, Reasoning about Functional Dependencies. Normal Forms, Properties of Decomposition, Normalization, different types of dependencies.

## UNIT - V

**Introduction to Transaction Management:** ACID properties, Transactions and Schedules, Concurrent Execution of Transactions, Lock-Based Concurrency Control. **Concurrency Control:** 2PL, Serializability and Recoverability, Introduction to Lock Management, Lock Conversions, Dealing with Deadlocks, Concurrency control without locking. **Crash Recovery:** Aries, Recovering from a System Crash.

**TEXT BOOKS:**

1. Raghu Ramakrishnan and Johannes Gehrke, Database Management Systems, McGraw-Hill, 3e, 2014 .

2. H.F.Korth and A.silberschatz, Database System Concepts, McGraw-Hill, 6e, 2011.

**REFERENCE BOOKS:**

1. D. Ullman, Principles of Database and Knowledge – Base Systems, Vol 1,1/e, Computer Science Press,1990.

2. RamezElmasri, Shamkant B. Navathe, Fundamentals of Database Systems, Pearson Education, 7e, 2016.

| Course Code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| CSM2204 | **Design and Analysis of Algorithms** | L | T | P | Int. | Ext | **3** |
| | | **3** | **0** | **0** | **30** | **70** | |

**COURSE OBJECTIVES:**

1. To learn techniques for effective problem solving in computing.
2. To analyze the asymptotic performance of algorithms.
3. To explain familiarity with major algorithms and data structures.
4. To apply algorithm designing techniques such as greedy algorithms, dynamic programming, divide and conquer, backtracking, branch and bound etc. for common engineering design situations.

**COURSE OUTCOMES:**

At the end of the course student will be able to
1. Analyze the efficiency of algorithms using mathematical analysis and asymptotic notations.
2. Employ divide-and-conquer and decrease-and-conquer strategies for problem solving.
3. Apply transform-and-conquer and string-matching techniques appropriately when an algorithmic design situation calls for it.
4. Solve problems using algorithm design methods such as the greedy method, dynamic programming.
5. Understand P and NP, NP-complete and NP-hard problems.

**UNIT-I**

**Introduction** Fundamentals of algorithmic problem solving, important problem type. Fundamentals of analysis of algorithms and efficiency, Analysis framework, Asymptotic Notations and Basic Efficiency classes, Mathematical Analysis of Non-recursive Algorithms, Mathematical Analysis of recursive Algorithms. **Brute Force:** Selection Sort and Bubble sort, Sequential Search and Brute Force String Matching, Closest Pair and Convex Hull Problems by Brute Force–Exhaustive Search.

**UNIT-II**

Divide-and-Conquer – Merge sort, Quick sort, Binary Search, Binary Tree Traversals and Related Properties, Multiplication of large integers and Strassen's Matrix Multiplication, Closest- Pair Convex-Hull Problems by Divide-and-Conquer. Decrease–and–Conquer: Insertion Sort, Depth-First Search and Breadth-First Search, Topological Sorting, Algorithms for Generating Combinatorial Objects, Decrease-by-a-Constant-Factor Algorithms, Variable-Size-Decrease Algorithms.

## UNIT-III

Transform-and-Conquer – Pre-sorting, Gaussian Elimination, Balanced Search Trees, Heaps and Heapsort, Horner's Rule and Binary Exponentiation, Problem Reduction. Space and Time Tradeoffs: Sorting by Counting, Input Enhancement in string Matching, Hashing, B-Trees.

## UNIT-IV

Dynamic Programming–Computing a Binomial Coefficient, Warshall's and Floyd's Algorithm, Optimal Binary Search Trees, The Knapsack Problem and Memory Functions. Greedy Technique: Prim's Algorithm, Kruskal's Algorithm, Dijkstra's Algorithm, Huffman Trees.

## UNIT-V

Limitations of Algorithm Power - Lower-Bound Arguments, Decision Trees, P, NP and NP – complete problems, Challenges of Numerical Algorithms. Coping with the Limitations of Algorithms Power: Backtracking, Branch-and-Bound.

**TEXT BOOKS:**

1. Introduction to Design & Analysis of Algorithms by Anany Levitin, Pearson Education, New Delhi, 3rd Edition, 2017.
2. Fundamentals of Computer Algorithms, Horowitz and Sahni, Galgothia publications.

**REFERENCE BOOKS**:

1. Introduction to Algorithms by Thomas H.Corman, Charles E.Leiserson, Ronald R. Rivest & Clifford Stein, Prentice Hall of India, New Delhi, New Delhi

| Course Code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| CSM2205 | **Probability And Statistical Methods** | L | T | P | Int. | Ext | **3** |
| | | **3** | **1** | **0** | **30** | **70** | |

## COURSE OBJECTIVES

1. To discuss basics of probability and related theorems, Problems. To study about conditional probability and Bayes theorem.
2. To study about random variables and their properties. To examine, analyze and compare Probability distributions.
3. To discuss regression and estimation techniques.
4. To discuss various types of tests such as F-test, Chi-square test. To study the various queuing models.

## COURSE OUTCOMES

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

**CO 1**: Ability to solve various problems regarding probability and conditional probability.

**CO 2**: Ability to determine measures of central tendency and dispersion.
**CO 3**: Examine, analyze, and compare probability distributions.
**CO 4**: Prepare null and alternative hypotheses and test its validity based on random sample.
**CO 5**: Solve various types of regression problems.

## Unit-I: Probability

Definitions of Probability, Addition Theorem, Conditional Probability, Multiplication Theorem, Bayes' Theorem of Probability and Geometric Probability.

**Random Variables and their Properties**: Discrete Random Variable, Continuous Random Variable, Probability Distribution, Joint Probability Distributions, Their Properties, Transformation Variables, Mathematical Expectations, Probability Generating Functions.

## Unit-II: Measures of Central Tendency and Dispersion

Arithmetic mean, median, mode, geometric mean, harmonic mean, range, quartile deviation, mean deviation, standard deviation, variance, and coefficient of variation.

## Unit-III: Probability Distributions

**Discrete Distributions**: Binomial, Poisson, Negative Binomial Distributions and their Properties.
**Continuous Distributions**: Uniform, Normal, Exponential Distributions and their Properties.

## Unit-IV: Multivariate Analysis and Curve Fitting

Correlation, Correlation Coefficient, Rank Correlation, Regression Analysis, Multiple Regression, Principles of Least Squares, and Curve Fitting.

## Unit-V: Estimation and Testing of Hypothesis

Types of Errors, Sample, Populations, Statistic, Parameter, Sampling Distribution, Standard Error, Un-Biasedness, Efficiency, Maximum Likelihood Estimator, Notion & Interval Estimation.
**Sample Tests**: Large Sample Tests Based on Normal Distribution, Small Sample Tests: Testing Equality of Means, Testing Equality of Variances, Test of Correlation Coefficient, Test for Regression Coefficient, Coefficient of Association, 2-Test for Goodness of Fit, Test for Independence.

## Text Books


1. *Probability & Statistics for Engineers and Scientists* by Ronald E. Walpole, Raymond H. Myers, Sharon L. Myers, Keying Ye, Pearson Education.
2. *Probability, Statistics, and Random Processes* by T. Veerarajan, Tata McGraw Hill.
3. *Fundamentals of Mathematical Statistics* by S.C. Gupta and V.K. Kapoor, Sultan Chand & Sons.


## Reference Books


1. *Probability & Statistics with Reliability, Queuing and Computer Applications* by Kishor S. Trivedi, Prentice Hall of India, 1999.

| Course Code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int. | Ext | |
| CSM2206 | **Neural Network Lab** | **0** | **0** | **3** | **50** | **50** | 1.5 |

**COURSE OBJECTIVES:**

1) Understand Neural Network Concepts
2) Write NN Programs in Python

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

1) Design and Train Neural Networks for Binary Logic Gates using Python
2) Create Hopfield NN using Python

**Solve the following problems using Python Programming Language CYCLE -** I

1. To Design and Train a Perceptron for Logic Gates ( AND, OR, NOT, NAND, NOR, EX-OR)
2. To design and train a perceptron for identifying ODD and EVEN number.
3. Implementing ANN from Scratch
4. Implementing Backpropagation Algorithm in NN
5. To create a Bi-directional Associative Memory (BAM) for ID and telephone number.

**CYCLE – II**

6. To design and train the Hopfield network to map the input vector with the stored vector and correct them.
7. Program to calculate output in a multi-layer feed forward network
8. Program to train a neural network to classify two clusters in a 2-dimensional space
9. Make Predictions with k-nearest neighbors on the Iris Flowers Dataset.
10. Design of Convolutional Neural Network

**REFERENCE BOOKS:**

1) Python AI: How to Build a Neural Network & Make Predictions by Déborah Mesquita

| Course Code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int. | Ext | |
| CSM2207 | **Database Management Systems Lab** | **0** | **0** | **3** | **50** | **50** | 1.5 |

**COURSE BJECTIVES:**

1. To introduce to a commercial DBMS such as ORACLE.
2. To learn and practice SQL commands for schema creation, data manipulation.
3. To learn conceptual and physical database design based on a case study.
4. To apply database design stages by studying a case study.

**COURSE OUTCOMES:**

At the end of the course student will be able to

1. Understand and effectively explain the underlying concepts of database technologies
2. Explore to a commercial RDBMS environment to write SQL queries.
3. Normalize a database
4. Develop mini project using DBMS Concepts.

**CYCLE-I**

**Laboratory Exercises Should Include:**

1. Developing a sample ER model for the specified database.

2. Create a database and learn to set various constraints (can use Sailors example from textbook1, University example from textbook2).

3. Familiarization of SQL DDL commands-create, alter, drop, rename and truncate.

4. Use of DML commands-select, insert, update and delete.

5. Use of different of operators for nested sub-queries.

6. Use of Joins.

7. Use of grouping functions.

8. Creating Views.

9. PL/SQL programming environment.

10. Declaring triggers and use of cursors.

**CYCLE II**

**MINI PROJECT**:

Mini Project in oracle that includes i)System Requirements, E-R diagrams, ii) Database design along with Key Constraints and Normalization, iii)Execution of SQL Commands (DDL,DML, JOINS, GROUPING, AGGREGATE FUNCTIONS & PL/SQL) on the database. Some sample applications are given below:

1. Accounting Package for Shops,
2. Database Manager for Magazine Agency or Newspaper Agency,
3. Ticket Booking for Performances,
4. Preparing Greeting Cards & Birthday Cards
5. Personal Accounts - Insurance, Loans, Mortgage Payments, Etc.,
6. Doctor's Diary & Billing System
7. Personal Bank Account
8. Class Marks Management
9. Hostel Accounting
10. Video Tape Library,
11. History of Cricket Scores,
12. Cable TV Transmission Program Manager,
13. Personal Library.
14. Sailors Database
15. Suppliers and Parts Database

**REFERENCE BOOKS:**

1. Database Management Systems; Raghu Ramakrishnan, Johannes Gehrke 4th Edition,McGraw- Hill.
2. Database System Concepts; A. Silberschatz, H. Korth 5th Edition, McGraw-Hill.

# GAYATRI VIDYA PARISHAD COLLEGE FOR DEGREE AND PG COURSES (AUTONOMOUS)
# ENGINEERING AND TECHNOLOGY PROGRAM
# RUSHIKONDA CAMPUS, VISAKHAPATNAM–530045.

| Course Code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int | Ext | |
| CSM2208 | **Design Thinking and Innovation (Skill Course - 2)** | **1** | **0** | **2** | **50** | **50** | 2 |

**Course Outcomes:**

After successful completion of this activity the student will be able to:

C01: Outline a problem, apply methods of Empathy on user groups

CO2: Describe and Define the problem specific to the user group
CO3: Apply Ideation tools to generate Ideas to solve the problem

C04: Develop prototypes

CO5: Test the ideas and demonstrate Storytelling ability to present the Ideas

**Students shall form into groups and Identify a problem (preferably, societal problem with engineering orientation to solve) suitable for the design thinking and go through the process week-wise. At the end of each phase, brief documentation shall be submitted and a final report covering all phases has to be submitted at the end of the semester.**

**Weeks 1-3:**

**Introduction to Design Thinking:** A primer on design_ thinking - Traditional approach, The new design thinking approach. Stages in Design Thinking: Empathize, Define, Ideate, Prototype, Test. Mindset for design thinking, Design thinking for product and process innovation, Difference between engineering design and design thinking.

**Case Studies**: General, Engineering and Service applications.

Activities: Identify an Opportunity and Scope of the Project Explore the possibilities and Prepare design briefly.

**Weeks 4-6:**

**Methods and Tools for Empathize and Define phases:**

**Empathize -** Methods of Empathize Phase: Ask 5 Why / 5W—H questions, Stakeholder map,

Empathy Map, Peer observation, Trend analysis.

**Define -** Methods of Define Phase: Storytelling, Critical items diagram, Define success

**Activities:** Apply the methods of empathize and Define Phases Finalize the problem statement

**Weeks 7-8:**

Methods **and Tools for Ideate phase:**

**Ideate -** Brainstorming, 2X2 matrix, 6-3-5 method, NABC method;

**Activities:** Apply the methods of Ideate Phase: Generate lots of Ideas

**Weeks 9-11:**

**Methods and Tools for Prototype Phase:**

**Prototype -** Types of prototypes - Methods of prototyping - Focused experiments, Exploration map, Minimum Viable Product;

**Activities:** Apply the methods of Prototype Phase: Create prototypes for selected ideas

**Weeks 12-13:**

**Methods and Tools for Test Phase:**

**Test -** Methods of Testing: Feedback capture grid, ...k/B testing

**Activities:** Collect feedback; iterate and improve the ideas

**Weeks 14-15:**

**Solution Overview -** Create a Pitch - Plan for scaling rip - Road map for implementation

**Activities:** Present your solution using Storytelling method

**Week 16:**

**Project Submission:** Fine tuning and submission of project report

**Reference books:**

1. Tim Brown, *Change* by *Design: How Design Thinking Transforms Organizations and inspires Innovation,* HarperCollins e-books, 2009.
2. Michael Lewrick, Patrick Link, Larry Leifer, *The Design Thinking Toolbox,* John Wiley &. Sons, 2020.
3. Michael Lewrick, Patrick Link, Larry Leifer, *The Design Thinking Playbook,* John. Wiley & Sons, 2018.
4. Kristin Fontichiaro, *Design Thinking,* Cherry Lake Publishing, USA, 2015.
5. Walter Brenner, Falk Uebernickel, *Design Thinking for Innovation - Research and Practice,* Springer Series, 2016.
6. Gavin Ambrose. Paul Harris, *Design Thinking,* AVA Publishing, 2010.
7. Muhammad Mashhood Alain, *Transforming an Idea into Business with Design Thinking,* First Edition, Taylor and Francis Group, 2019.
8. S. Balaram, *Thinking Design,* Sage Publications. 2011.

| Course Code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int | Ext | |
| CSM2210 | **NCC/ NSS** | **0** | **0** | **2** | **0** | **0** | 0 |

**NCC:** NCC facility is provided for students to develop their leadership, character, comradeship, discipline, a secular outlook and spirit. The main function of NCC is to stand for the country in tough times or when in need to provide a suitable environment to motivate the youth to take up a career in the Armed forces. Students have to apply within the due date as soon as the academic session starts. They have to do at least 2 hours of service in a Week.

**NSS** : The core objective of N.S.S. (National Service Scheme) is to contribute towards national development and for the student's creative development. Various camps are organized from time to time where students learn life skills, leadership and teamwork along with serving society. Students have to apply within the due date as soon as the academic session starts. They have to do at least 2 hours of service in a Week.

# GAYATRI VIDYA PARISHAD COLLEGE FOR DEGREE AND PG COURSES (AUTONOMOUS)
# ENGINEERING AND TECHNOLOGY PROGRAM
# RUSHIKONDA CAMPUS, VISAKHAPATNAM–530045.

| Course code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int. | Ext | |
| Honors 2-2 | Artificial Neural Networks | 3 | 1 | 0 | 30 | 70 | 4 |

## COURSE OBJECTIVES:

1. To understand the biological neural network and to model equivalent neuron models.
2. To understand the architecture, learning algorithm and issues of various feed forward and feedback neural networks.

## COURSE OUTCOMES:

By completing this course the student will be able to:

**CO1.** Understand Neural Networks Basics and Architecture.

**CO2.** Learn Single layer and Multi-layer perceptron with back propagation algorithm.

**CO3.** Create Support Vector Machines for various pattern recognition problems.

**CO4.** Train Models using Self Organizing Maps.

**CO5.** Testing Neuro Dynamic Models.

## UNIT – I

**Introduction to Neural Netwoks**: A Neural Network, Human Brain, Models of a Neuron, Neural Networks viewed as Directed Graphs, Network Architectures, Knowledge Representation, Artificial Intelligence and Neural Networks.

## UNIT – II

**Single Layer Perceptrons**: Adaptive Filtering Problem, Unconstrained Organization Techniques, Linear Least Square Filters, Least Mean Square Algorithm, Learning Curves, Learning Rate Annealing Techniques, Perceptron –Convergence Theorem, Relation Between Perceptron and Bayes Classifier for a Gaussian Environment.

**Multilayer Perceptron**: Back Propagation Algorithm XOR Problem, Heuristics, Output Representation and Decision Rule, Computer Experiment, Feature Detection.

## UNIT – III

**Support Vector Machines:** Optimal Hyperplane for Linearly Separable pattern, Optimal Hyperplane for Non-Separable pattern, How to build a Support Vector Machine for Pattern Recognition, Example XOR Problem, Computer Experiment.

## UNIT – IV

**Self-Organization Maps (SOM):** Two Basic Feature Mapping Models, Self-Organization Map, SOM Algorithm, Properties of Feature Map, Computer Simulations, Learning Vector Quantization, Adaptive Pattern Classification.

## UNIT – V

**Neuro Dynamics:** Dynamical Systems, Stability of Equilibrium States, Attractors, Neuro Dynamical Models, Manipulation of Attractors as a Recurrent Network Paradigm. **Hopfield Models** – Hopfield Models, Computer Experiment.

## TEXT BOOKS:

1. Neural Networks a Comprehensive Foundations, Simon Haykin, PHI edition.

## REFERENCE BOOKS:

1. Artificial Neural Networks - B. Vegnanarayana Prentice Hall of India P Ltd 2005
2. Neural Networks in Computer Intelligence, Li Min Fu MC GRAW HILL EDUCATION 2003
3. Neural Networks -James A Freeman David M S Kapura Pearson Education 2004.
4. Introduction to Artificial Neural Systems Jacek M. Zurada, JAICO Publishing House Ed. 2006.

# B. Tech Computer Science and Engineering (AI and ML)
## (R-22 Regulation)
### III Year – I Semester

| Course code | Category | Course Title | Hours per week | | | Internal Marks | External Marks | Total Marks | Credits |
|---|---|---|---|---|---|---|---|---|---|
| | | | L | T | P | | | | |
| CSM3101 | PC/PCC | Data Warehousing and DataMining | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM3102 | PC/PCC | Object Oriented Software Engineering | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM3103 | PC/PCC | Computer Networks | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM3104 | OEC/JOE | Open Elective – I | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM3105 | PEC | Elective-I | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM3106 | PC/PCC | Computer Networks Lab | 0 | 0 | 3 | 50 | 50 | 100 | 1.5 |
| CSM3107 | PC/PCC | Software Engineering& Mini Project Lab | 0 | 0 | 3 | 50 | 50 | 100 | 1.5 |
| CSM3108 | SAC/SC | Skill Course – 3 Web Technologies | 1 | 0 | 2 | 50 | 50 | 100 | 2 |
| CSM3109 | MC | Technical Communication& Soft Skills | 2 | 0 | 0 | 100 | 0 | 100 | 0 |
| Summer Internship, Two months mandatory after 2nd year to be evaluated during 5th Semester | | | 0 | 0 | 3 | 0 | 100 | 100 | 1.5 |
| **Total Credits** | | | | | | | | | **21.5** |

| Title of the Program | L | T | P | Credits |
|---|---|---|---|---|
| **Honors/Minor Courses (The hours distribution can be 3-0-2 or 3-1-0 also)** | **3** | **1** | **0** | **4** |

**Elective-I**
Introduction to Data Science
**Expert Systems**
Pattern Recognition

**Open Elective – I**
Offered by Mechanical Engineering
Offered by Civil Engineering
Offered by ECE
Offered by CSE
Offered by CSE(AI&ML): Introduction to Artificial Intelligence

# GAYATRI VIDYA PARISHAD COLLEGE FOR DEGREE AND PG COURSES (AUTONOMOUS)
# ENGINEERING AND TECHNOLOGY PROGRAM
# RUSHIKONDA CAMPUS, VISAKHAPATNAM–530045.

| Course code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int. | Ext | |
| CSM3101 | Data warehousing and Data Mining | 3 | 0 | 0 | 30 | 70 | 3 |

**COURSE OBJECTIVES:**

1: Illustrate the evolution and importance of Data Mining and its applications various types of data.

2: Explain the evolution of data warehouses and data mining systems, pre-processing techniques, OLAP operations and concepts of data cube.

3: Experiment with various data mining algorithms with association and correlations of frequent Patterns mining.

4: Experiment the principles of statistics with classification and clustering methods for mining patterns.

5: Illustrate the basic concepts of Clustering, types of clustering and graph mining approaches.

**COURSE OUTCOMES:**

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

**CO1: Explain** the functionality of various data mining components and concepts of Data Pre-Processing.

**CO2: Explain** data ware house design and data cube technology for summarization and querying high dimensional data.

**CO3: Compare** and **contrast** the strengths and limitations of various data mining models.

**CO4: Apply** knowledge for various classification and prediction techniques for developing new Data Mining algorithms.

**CO5: Evaluate** various clustering analysis algorithms for designing new Data Mining algorithms.

## UNIT-I

**Introduction to Data Mining, Data pre-processing:** Evolution of IT into DBMS, Motivation and importance of Data Warehousing and Data Mining, Kinds of Patterns, Technologies, Applications, Major Issues in Data Mining, Data Objects and Attributes Types, Statistical Descriptions of Data, Data Visualization, Estimating Data Similarity and Dissimilarity, Quality data, Data Cleaning, Data Integration, Data Reduction, Data Transformation, Discretization and Concept Hierarchy Generation.

## UNIT-II

Data Warehouse, OLAP Technology and Data Cube Technology: Basic Concepts of Data warehouse, Data Modeling using Cubes and OLAP, DWH Design and usage, Implementation using Data Cubes and OLAPs, Data Generalization with AOI, Preliminary Concepts of Data Cube Computation, Data Cube Computation Methods: Multi-way Array Aggregation for Full Cube, Multi-dimensional Data Analysis in cube space.

## UNIT-III

**Mining Frequent Patterns Based on Associations and Correlations:** Basic Concepts, Frequent Item set Mining Methods: Apriori Algorithm, Association Rule Generation, Improvements to A Priori, FP-Growth Approach, Mining Closed and Max Patterns, Pattern Evaluation Methods, Association mining in multi-level, multi-dimensional space.

## UNIT-IV

**Classification & Prediction:** Basic Concepts, Decision Tree Induction, Bayes Classification, Rule-Based Classification, Model Evaluation and Selection, Techniques to Improve Classification Accuracy, Classification by Back Propagation, Associative Classification, K-nearest neighbor classifier.

## UNIT-V

**Cluster Analysis:** Basic Concepts and issues in clustering, Types of Data in Cluster Analysis, Partitioning Methods, Hierarchical Methods, Density Based Methods, Grid Based Methods, Evaluation of Clustering Solutions, Graph Mining Approaches.

### TEXT BOOKS

1. Data Mining- Concepts and Techniques by Jiawei Han, Micheline Kamber and Jian Pei – Morgan Kaufmann publishers --3rd edition

### REFERENCE BOOKS:

1. Introduction to Data Mining, Adriaan, Addison Wesley Publication
2. Data Mining Techniques, A.K.Pujari, University Press Data mining concepts by Tan, Steinbech, and Vipin Kumar - Pearson Edu publishers

# GAYATRI VIDYA PARISHAD COLLEGE FOR DEGREE AND PG COURSES (AUTONOMOUS)
# ENGINEERING AND TECHNOLOGY PROGRAM
# RUSHIKONDA CAMPUS, VISAKHAPATNAM–530045.

| Course code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int. | Ext | |
| CSM3102 | Object-Oriented Software Engineering | 3 | 0 | 0 | 30 | 70 | 3 |

## COURSE DESCRIPTION:

This Course will help the students to get familiar with various models for software development process and helps to understand security services which needs to be adopted at each phase of Software Development Life Cycle. The purpose of requirements management is to ensure that the organization validates and meets the needs of its customers and external and internal stakeholders. Requirements management provides a way to avoid errors by keeping track of changes in requirements and fostering communication with stakeholders from the start of a project throughout the engineering lifecycle. Students will be able to identify requirements of the software problem/application and prepare estimation in terms of cost and effort. The purpose of requirements management is to ensure that the organization validates and meets the needs of its customers and external and internal stakeholders. Requirements management provides a way to avoid errors by keeping track of changes in requirements and fostering communication with stakeholders from the start of a project throughout the engineering lifecycle. Student able to identify requirements of the software Problem/application and prepare estimation in terms of cost and effort.

## COURSE OBJECTIVES:

1. Understand the importance of Object-Oriented Software Engineering in Software Development.
2. Elicit, analyze, and specify security requirements to develop problem statement and requirements.
3. Design UML Diagrams.
4. Acquaint with various architectural models and design patterns.
5. Develop and apply testing strategies.

## COURSE OUTCOMES:

At the end of the course student will be able to

**CO1:** Understand the concepts related to development of Software Engineering.

**CO2:** Apply the knowledge of requirements elicitation process.

**CO3:** Design the UML Diagrams for improving communication between client and developer.

**CO4:** Analyze architecture models and design patterns.

**CO5:** Apply and develops various testing strategies on the developed products.

## UNIT-I

**Introduction to Object Oriented Software Engineering:** Nature of the Software, Types of Software, Software Engineering Projects, Software Engineering Activities, Software Quality, and Introduction to Object Orientation, Software Process Models-Waterfall Model, and Opportunistic Model, Phased Released Model, Spiral Model, Evolutionary Model, Concurrent Engineering Model.

## UNIT-II

**Requirements Engineering:** Domain Analysis, Problem Definition and Scope, Requirements Definition, Types of Requirements, Techniques for Gathering and Analyzing Requirements, Requirement Documents, Reviewing, Managing Change in Requirements.

## UNIT-III

**Unified Modeling Language**: Introduction to UML, Modeling Concepts: Systems, Models and Views Data Types, Abstract Data Types, Instances Classes, Abstract Classes, Objects, Event Classes, Events, and Messages Object-Oriented Modeling, Falsification and Prototyping. Types of UML Diagrams- **Structural diagrams**: Class Diagrams, Associations and Multiplicity, Labelling Associations, Validating associations, Reflexive Associations, Generalization, Component Diagrams, Deployment Diagrams, and Object Diagrams. **Behavioral Diagrams**: Use Case Diagrams, Activity Diagrams, State Machine Diagrams, Sequence Diagrams.

## UNIT-IV

**Software Design and Architecture:** Process of Design, Principles Leading to Good Design, Techniques for Making Good Design Decisions, Good Design Document; Pattern Introduction, Design Patterns: Abstraction-Occurrence Pattern, General Hierarchical Pattern, Play-Role Pattern, Singleton Pattern, Observer Pattern, Delegation Pattern, Adaptor Pattern, Façade Pattern, Immutable Pattern, Read-Only Interface Pattern and The Proxy Pattern; Software Architecture Contents of Architecture Model, Architectural Patterns: Multilayer, Client-Server, Broker, Transaction Processing, Pipe & Filter and MVC Architectural Patterns.

## UNIT-V

**Software Testing**: Overview, Testing Conventional Applications: **White-Box Testing**: Basis Path Testing: Flow Graph Notation, Independent Program Paths, Deriving Test Cases, Graph

Matrices, Control Structure Testing: Condition Testing, Data Flow Testing, Loop Testing, **Black Box Testing**: Graph-Based Testing Methods, Equivalence Partitioning, Boundary Value Analysis, Orthogonal Array Testing. **Testing Activities**: Component Inspection, Usability Testing, Unit Testing, Integration Testing, System Testing. **Managing Testing**: Planning Testing, Documenting Testing, Assigning Responsibilities, Regression Testing, Automating Testing, Model based testing. Software Quality, Quality Attributes and Criteria, Introduction to Software Project Management.

## CASE STUDY:

1. Simple Chat Instant Messaging System
2. GPS Based Automobile Navigation System
3. Waste Management Inspection Tracking System (WMITS)
4. Geographical Information System

## TEXT BOOKS:

1. Object-Oriented Software Engineering Practical software development using UML and Java by Timothy C. Lethbridge & Robert, Langaniere Mcgraw-Hill.
2. Object-Oriented Software Engineering: Using UML, Patterns and Java, Bernd Bruegge and Allen H. Dutoit, 2nd Edition, Pearson Education Asia.

## REFERENCE BOOKS:

1. Software Engineering: A Practitioner's Approach, Roger S Pressman.
2. A Practical Guide to Testing Object-Oriented Software, John D. McGregor; David A. Sykes, Addison-Wesley Professional.
3. Software Engineering, K.K. Agarwal, New Age Publications 2008.

# GAYATRI VIDYA PARISHAD COLLEGE FOR DEGREE AND PG COURSES (AUTONOMOUS)
## ENGINEERING AND TECHNOLOGY PROGRAM
## RUSHIKONDA CAMPUS, VISAKHAPATNAM–530045.

| Course Code | Title of the Course | Contact hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| **CSM3103** | **Computer Networks** | L | T | P | Internal | External | **3** |
| | | **3** | **0** | **0** | **30** | **70** | |

**COURSE OBJECTIVES:**

1. Provide a comprehensive understanding of data communication and networking principles.
2. Cover network components, topologies, and layered models, including the OSI and TCP/IP protocols.
3. Explore key topics like transmission media, routing algorithms, transport services, and congestion control.
4. Examine modern networking trends such as delay-tolerant networking and content delivery networks.
5. Equip students with the knowledge needed to design and manage efficient network systems.

**COURSE OUTCOMES:**

**CO 1:** Illustrate the fundamental concepts of data communication and networking principles.

**CO 2:** Identify and explain the components, topologies, and layered models including OSI and TCP/IP protocols.

**CO 3:** Demonstrate knowledge of transmission media, routing algorithms, transport services, and congestion control techniques.

**CO 4:** Understand and analyze modern networking trends such as delay-tolerant networking and content delivery networks.

**CO 5:** Apply networking principles to design and manage efficient network systems.

## UNIT I

**DATA COMMUNICATION:** Characteristics, Components, Data flow, Network criteria, Topologies, Network model, Layered tasks, ARPANET, OSI model, TCP/IP protocol suite, Addressing.
**PHYSICAL LAYER:** Transmission Media: Guided and unguided, Connecting devices: Hub, switch, bridge, router, Gateway.

**DATA LINK LAYER:** Design issues, Error detection and correction, Elementary data link protocols, Sliding window protocols.
**RANDOM ACCESS:** ALOHA, CSMA/CD, CSMA/CA, Controlled access, Channelization, Wired LAN: IEEE Standards, Standard Ethernet, Wireless LAN: IEEE802.11, ATM: architecture, Layers.

## UNIT III

**NETWORK LAYER:** Design issues, Routing algorithms, Internetworking, Network layer in the Internet**.**
**CONGESTION CONTROL:** Approaches to Congestion Control, Traffic-Aware Routing, Traffic Throttling, Load shedding, traffic shaping.
**IP Addressing:** IPv4 Addressing, Subnetting, IPv6 Addressing, Transition from IPv4 to IPv6.

## UNIT IV

**TRANSPORT LAYER:** Transport services, Elements of transport Protocols, TCP and UDP.
**DELAY-TOLERANT NETWORKING:** DTN Architecture, the Bundle Protocol.

## UNIT V

**APPLICATION LAYER:** Domain Name Space (DNS), SNMP, Electronic mail: MIME, SMTP, IMAP.
**CONTENT DELIVERY:** Content Delivery Networks, Peer-to-Peer Networks.

**TEXTBOOK:**

1. Andrew S. Tanenbaum, David J. Wetherall, Computer Networks, 5th Edition, Pearson New International Edition, 2016.
2. Behrouz A. Forouzan, Data Communication and Networking, 4th Edition, McGraw- Hill, 2017.

**REFERENCE BOOKS:**

1. William Stallings, Data and Computer Communication, 8th Edition, Pearson, PHI, 2013.

2. Douglas Comer, Internetworking with TCP/IP, 6th Edition, PHI, 2015.

**WEB REFERENCES:**

1. https://nptel.ac.in/courses/106105183/

# GAYATRI VIDYA PARISHAD COLLEGE FOR DEGREE AND PG COURSES (AUTONOMOUS)
# ENGINEERING AND TECHNOLOGY PROGRAM
# RUSHIKONDA CAMPUS, VISAKHAPATNAM–530045.

| Course code | Title of the Course | Contact Hours/week | | | Allotment Of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int. | Ext | |
| CSM3104 | Introduction to Artificial Intelligence | 3 | 0 | 0 | 30 | 70 | 3 |

**COURSE OBJECTIVES:**

1. To learn about AI problem, Production Systems and their characteristics.
2. To understand the importance of search and the corresponding search strategies for solving AI problems.
3. Analyze Natural Language Processing and Expert Systems.

**COURSE OUTCOMES:**

At the end of the course student will be able to

**CO1:** Solve AI Problems using the knowledge of State Space Search.
**CO2:** Apply several optimal search strategies and heuristic techniques to solve AI problems.
**CO3:** Learn relational, inferential, inheritable and procedural knowledge and the corresponding Knowledge representation approaches.
**CO4:** Apply the concepts of Reasoning under Uncertainty and solve the complex problems of AI.
**CO5:** Implement AI problem solving approaches to develop natural language processing and Expert systems.

**UNIT-I**

**Introduction to Artificial Intelligence:** Artificial Intelligence, AI Problems, AI Techniques, Defining the Problem as a State Space Search, Problem Characteristics, Production Systems.

**UNIT-II**

**Search Techniques:** Issues in The Design of Search Programs, Un-Informed Search, BFS, DFS; Heuristic Search Techniques: Generate-And- Test, Hill Climbing, Best- First Search, A* Algorithm, Problem Reduction, AO*Algorithm, Constraint Satisfaction, Means-Ends Analysis.

**UNIT-III**

**Knowledge Representation using Rules:** Procedural Vs Declarative Knowledge, Logic programming, Forward Vs Backward Reasoning.

**Symbolic Logic**: Propositional Logic, First Order Predicate Logic: Representing Instance and is-a Relationships, Computable Functions and Predicates, Unification & Resolution, Natural deduction.

## UNIT-IV

**Structured Representations of Knowledge**: Semantic Nets, Partitioned Semantic Nets, Frames, Conceptual Dependency, Conceptual Graphs, and Scripts.

**Reasoning under Uncertainty**: Introduction to Non-Monotonic Reasoning, Truth Maintenance Systems. **Fuzzy Logic:** Crisp Sets, Fuzzy Sets, Fuzzy Logic Control, Fuzzy Inferences &Fuzzy Systems.

## UNIT-V

**Natural Language Processing**: Steps in the Natural Language Processing, Syntactic processing and Augmented Transition Nets, Semantic Analysis, NLP Understanding Systems.

**Experts Systems:** Overview of an Expert System, Architecture of an Expert Systems, Different Types of Expert Systems- Rule Based, Frame Based, Decision Tree based, Case Based, Neural Network based.

.

**TEXT BOOKS:**

1. Artificial Intelligence, Elaine Rich and Kevin Knight, Tata Mcgraw-Hill Publications

**REFERENCE BOOKS:**

1. Artificial Intelligence, George F Luger, Pearson Education Publications

2. Artificial Intelligence : Amodern Approach, Russell and Norvig, Prentice Hall
3. Introduction To Artificial Intelligence & Expert Systems, Patterson, PHI publications

# GAYATRI VIDYA PARISHAD COLLEGE FOR DEGREE AND PG COURSES (AUTONOMOUS)
## ENGINEERING AND TECHNOLOGY PROGRAM
## RUSHIKONDA CAMPUS, VISAKHAPATNAM–530045.

| Course Code | Title of the Course | Contact hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Internal | External | |
| CSM3105 | Introduction to Data Science | 3 | 0 | 0 | 30 | 70 | 3 |

**COURSE OBJECTIVES:**

1. Students will able to learn the relationship of Data Science with the other allied sciences.
2. Students will able to grasp insights of data preprocessing.
3. Students will have proficiency with statistical analysis of data.
4. Students will develop the ability to build and assess data-based models.
5. Students will demonstrate skill in data management.

**COURSE OUTCOMES:**

**CO 1:** Illustrate the Data Science Methodology.
**CO 2:** Identify different sources of Data and Demonstrate different computing tools involved in data handling.
**CO 3:** Demonstrate various Techniques involved in Data analysis and Analytics
**CO 4:** Understanding of when to use supervised and unsupervised statistical learning methods on labeled and unlabeled datasets
**CO 5:** Apply domain expertise to solve real world problems using data science

### UNIT I

**Introduction:** What Is Data Science, Where Do We See Data Science, How Does Data Science Relate to Other Fields, The Relationship between Data Science and Information Science, Computational Thinking, Skills for Data Science, Tools for Data Science, Issues of Ethics, Bias, and Privacy in Data Science .

### UNIT II

**Data:** Introduction, Data Types, Structured Data, Unstructured Data, Challenges with Unstructured Data, Data Collection, Open Data, Social Media Data, Multimodal Data, Data Storage and Presentation, Data Pre-processing, Data Cleaning, Data Integration, Data Transformation, Data Reduction, Data Discretization.

### UNIT III

**Techniques:** Introduction , Data Analysis and Data Analytics, Descriptive Analysis, Variables, Frequency Distribution, Measures of Centrality, Dispersion of a Distribution, Diagnostic Analytics, Correlation, Predictive Analytics, Prescriptive Analytics, Exploratory Analysis, Mechanistic Analysis, Regression.

## UNIT IV

**Supervised Learning**: Introduction, Logistic Regression, Softmax Regression, Classification with KNN, Decision Tree, Decision Rule, Classification Rule, Association Rule, Random Forest, Naïve Bayes, Support Vector Machine (SVM)

**Unsupervised Learning:** Introduction, Agglomerative Clustering, Divisive Clustering, Expectation Maximization (EM)

## UNIT V

**Applications, Evaluations, and Methods:** Hands-On with Solving Data Problems: Introduction, Collecting and Analyzing Twitter Data, Collecting and Analyzing YouTube Data, Data Collection, Experimentation, and Evaluation: Introduction, Data Collection Methods, Surveys, Survey Question Types, Survey Audience, Survey Services, Analyzing Survey Data, Pros and Cons of Surveys, Interviews and Focus Groups, Why Do an Interview?, Why Focus Groups?, Interview or Focus Group Procedure, Analyzing Interview Data, Pros and Cons of Interviews and Focus Groups, Log and Diary Data, User Studies in Lab and Field Picking Data Collection and Analysis Methods,: Introduction to Qualitative Methods, Mixed Method Studies, Evaluation, Comparing Models, Training–Testing and A/B Testing, Cross-Validation.

**Text Book:**

1. Chirag Shah, 2020, A Hands-On Introduction to Data Science, Cambridge University Press

**Reference Books:**

1 .Jeffrey S. Saltz, Jeffrey M. Stanton, 2018, An Introduction to Data Science, SAGE Publications
2. Joel Grus, 2015, “Data Science from Scratch”.
3. Lillian Pierson, Jake Porway, “Data Science for Dummies”, 2nd Edition, For Dummies, 2017

| Course Code | Title of the Course | Contact hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| CSM3105 | Expert Systems | L | T | P | Internal | External | 3 |
| | | 3 | 0 | 0 | 30 | 70 | |

**COURSE OBJECTIVES:**

1. Provide a comprehensive understanding of the foundational concepts, architecture, and categories of Expert Systems.
2. Analyze various knowledge acquisition and representation methods used in Expert Systems.
3. Design and develop Expert Systems for specific applications, considering software engineering principles.
4. Evaluate the performance, limitations, and error potential of Expert Systems throughout their lifecycle.
5. Apply reasoning techniques, inference mechanisms, and problem-solving strategies in real-world scenarios.

**COURSE OUTCOMES:**

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

**CO 1:** Understand the basic concepts, architecture, and categories of Expert Systems.
**CO 2:** Analyze the Methods of Inference and Knowledge Representations.
**CO 3:** Design and develop an Expert System for a specific application, considering software engineering principles.
**CO 4:** Evaluate the performance of Expert Systems using CLIPS.
**CO 5:** Apply reasoning techniques, inference mechanisms, and problem-solving strategies in real-world scenarios.

## UNIT I

**Introduction:** Introduction to Expert System, Definitions, Importance of Expert System, Characteristic features of Expert System, Applications of Expert System, Different categories of Expert Systems, Rule Based System Architecture, Neural Network Architecture.

## UNIT II

**Methods of Inference:** Introduction, Trees, Lattices, and Graphs, State and Problem Spaces, And-Or Trees and Goals, Deductive Logic and Syllogisms, Rules of Inference, Limitations of Propositional Logic, First Order Predicate Logic, Logic Systems, Resolution, Resolution Systems and Deduction, Shallow and Causal Reasoning, Resolution and First Order Predicate Logic, Forward and Backward Chaining, Meta knowledge.

## UNIT III

**Design of Expert System:**

Introduction, Selecting the appropriate Problem, Stages in the Developing Expert System, Errors in Development stages, Software Engineering and Expert Systems, The Expert SystemLife Cycle, Expert System Design Examples- Certainty factors, Decision trees.

## UNIT IV

**Introduction to CLIPS**

Introduction, CLIPS, Notation, Fields, Entering and Exiting CLIPS, Facts, Adding and Removing Facts, Modifying and Duplicating Facts, The Watch Command, The Deffacts Construct, The Components of a Rule, The Agenda and Execution, Commands for Manipulating Constructs, The Printout Command, Using Multiple Rules, The Set-Break Command, Loading and Saving Constructs, Commenting Constructs, Summary.

## UNIT V

**Expert System Design Examples:**

Introduction, Certainty Factors, Decision Trees, Backward Chaining, a Monitoring Problem. **Case Study:** MYCIN, DENDRAL using CLIPS.

**Textbooks**

1. Expert Systems: Principles and Programming by Joseph C. Giarratano and Gary D. Riley, Cengage Learning, 4th Edition, 2004.
2. Introduction to Expert Systems by Peter Jackson, Addison-Wesley, 3rd Edition, 1998.

**Reference Books**

1. Artificial Intelligence: A Modern Approach by Stuart Russell and Peter Norvig, Pearson, 3rd Edition, 2009.
2. Principles of Artificial Intelligence by Nils J. Nilsson, Springer, 1982.
3. Expert Systems: Concepts and Examples by Tom Addis and John R. Gooding, Chapman & Hall, 1989.
4. Building Expert Systems by Frederick Hayes-Roth, Donald A. Waterman, and Douglas B. Lenat, Addison-Wesley, 1983.
5. The Engineering of Knowledge-Based Systems: Theory and Practice by A. T. Schreiber, Wielinga B. J., and J. A. Breuker, Springer, 1993.

# GAYATRI VIDYA PARISHAD COLLEGE FOR DEGREE AND PG COURSES (AUTONOMOUS)
## ENGINEERING AND TECHNOLOGY PROGRAM
## RUSHIKONDA CAMPUS, VISAKHAPATNAM–530045.

| Course Code | Title of the Course | Contact hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| CSM3105 | Pattern Recognition | L | T | P | Internal | External | 3 |
| | | 3 | 0 | 0 | 30 | 70 | |

**COURSE OBJECTIVES**

1. Provide a comprehensive understanding of the fundamental concepts and techniques of pattern recognition.
2. Teach various methods for feature extraction and data pre-processing.
3. Explore different classification and clustering algorithms.
4. Apply pattern recognition techniques to real-world problems.
5. Evaluate and compare the performance of different pattern recognition systems.

**COURSE OUTCOMES**

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

**CO 1:** Explain the basic principles and applications of pattern recognition.
**CO 2:** Implement various feature extraction and data pre-processing techniques.
**CO 3:** Utilize classification and clustering algorithms for pattern recognition tasks.
**CO 4:** Develop and test pattern recognition systems for specific applications.
**CO 5:** Analyze the performance of pattern recognition systems and suggest improvements.

## UNIT I: Introduction

Overview of Pattern Recognition- Relations of PR with other Systems, PR Applications, Different Approaches to Pattern Recognition- Statistical Approach to PR, Syntactic Approach to PR, Neural Approach to PR, Examples of PR Approaches. Other Approaches to PR.

## UNIT II: Structure of PR System:

Abstract Representation of PR Mappings, Structure of PR System, Patterns and Feature s, Feature Extraction Examples, Object Description and Classification, Figure Recognition, Numerical Results and Analysis. Feature Vector and Feature Space, training and Learning in PR System.

**Statistical Pattern Recognition:** Introduction, Gaussian Case and Class Dependency, Discriminate Function, Examples, Classifier Performance.

## UNIT III: Syntactic Pattern Recognition:

Overview of Syntactic Pattern Recognition, Grammar Based Approaches and Applications, Examples of String Generation as Pattern Description, 2-D line Drawing Description Grammar,

Character Description using PDL, Object Description using Projected Cylinder Models, Block World Description Models, Heuristic Generation of Grammars.

## UNIT IV: Recognition of Syntactic Description:

Recognition by Matching, Recognition by Parsing, CYK Parsing Algorithm, Augmented Transition Nets in Parsing, Graph Based structure representation, Structured Strategy to Compare Attributed Graphs.

## UNIT V: Neural Pattern Recognition:

Introduction to Neural Networks, Neural Network Structure for PR Applications, Physical Neural Networks, ANN Model, NN Based PR Association, Matrix Approaches and Examples.

## Text Book:

**1.** Pattern Recognition- Statistical, Structural and Neural Approaches, Rober.J. Shelkoff, John Wiley & Sons, NY1992, ISBN0-471-52974-5.

## Reference Book:

1. Neural Networks for pattern recognition, Christopher M.Bishop Oxford University Press.

2. Pattern Classification, Richard O.Duda , Wiley IndiaEdition

# GAYATRI VIDYA PARISHAD COLLEGE FOR DEGREE AND PG COURSES (AUTONOMOUS)
# ENGINEERING AND TECHNOLOGY PROGRAM
# RUSHIKONDA CAMPUS, VISAKHAPATNAM–530045.

| Course Code | Title of the Course | Contact hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Internal | External | |
| CSM3106 | Computer Networks Lab | 0 | 0 | 3 | 50 | 50 | 1.5 |

**COURSE OBJECTIVES:**

1. Provide hands-on experience with various networking tools, devices, and protocols.
2. Teach students to create network cables and analyze network packets using tools like Wireshark.
3. Explore network simulators such as NS2 and CISCO Packet Tracer.
4. Cover different network topologies, error detection and correction techniques.
5. Develop practical skills in socket programming for TCP and UDP protocols, as well as configuring, analyzing, and troubleshooting network systems.

**COURSE OUTCOMES:**

At the end of the course student will be able to

**CO 1:** Demonstrate the ability to create and implement different types of network cables, and connect computers in a Local Area Network.
**CO 2:** Illustrate the use of Wire Shark Packet Analyzer Tool to capture and analyze network packets in peer-to-peer mode.
**CO 3:** Utilize network simulators like NS2 and CISCO Packet Tracer to study and configure various network topologies and protocols.

**CYCLE-1**

1. Study of different types of network cables and practically implement the cross-wired cable and straight through cable using clamping tool.
2. Study of network devices in detail
3. Connect the computer in Local Area Networks
4. Study of Network Simulator (NS 2) and CISCO Packet Tracer.
5. Demonstrate the packets captured traces using Wire shark Packet Analyzer Tool for peer to peer mode.
6. Network Topology: Bus Topology, RING Topology, and STAR Topology.
7. Write a Program with following four options to transfer
   a) Characters separated by space
   b) One Strings at a time
   c) One Sentence at a time

   (To demonstrate Framing, Flow control, Error control).

**CYCLE-2**

8. Write a program for error detection and correction for 7/8 bits ASCII codes using HammingCodes or CRC.
9. Write a program to simulate Go Back N and Selective Repeat Modes of Sliding WindowProtocol in peer to peer mode
10. Write a program using TCP socket for wired network for following
    **a)** Say Hello to Each other
    **b)** File transfer
11. Write a program using UDP Sockets to enable file transfer (Script, Text, Audio and Videoone file each) between two machines.
12. Development of applications such as E – mail/ Multi – user Chat.

**REFERENCE BOOKS:**

1. Internet and Web Technologies by Raj Kamal, TataMcGraw-Hill
2. Programming the World Wide Web by Robert W. Sebesta, PearsonEducation.

# GAYATRI VIDYA PARISHAD COLLEGE FOR DEGREE AND PG COURSES (AUTONOMOUS)
# ENGINEERING AND TECHNOLOGY PROGRAM
# RUSHIKONDA CAMPUS, VISAKHAPATNAM–530045.

| Course code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int. | Ext | |
| CSM3107 | Software Engineering & Mini Project LAB | 0 | 0 | 3 | 50 | 50 | 1.5 |

**Course Description:**

The purpose of this course is to impart knowledge on the basic principles of software engineering and enabling the learner to understand software lifecycle stages. Systematic development of software products or solutions is emphasized throughout the course to enable the student ensure quality of development activities. The purpose of the Software Engineering Lab course is to familiarize the students with modern software engineering methods and tools, **Rational Products/Visual Paradigm**. The course is realized as a project-like assignment that can, in principle, by a team of three/four students working full time. Typically the assignments have been completed during the semester by each project team. The goal of the Software Engineering Project is to have a walk through from the requirements, design to implementing and testing. An emphasis is put on proper documentation. Term projects are projects that a group student might take through from initial specification to implementation by giving equal importance to both design and implementation

**COURSE OBJECTIVES:**

1. Provide Introduction to Software Engineering and process of Software production along with UML diagrams.
2. Facilitate Analysis of requirements for software solution development Learn to develop a Mini-Project.
3. Summarize architecture, design, and implementation considerations of software solution.

**COURSE OUTCOMES:**

At the end of the course student will be able to

**CO1:** Design different Structural UML Diagrams for a project using Rationale Architect Software Designer.

**CO2:** To test the software system thoroughly for all scenarios.

**CO3:** Develop the contents of Mini-Project for a given problem.

# GAYATRI VIDYA PARISHAD COLLEGE FOR DEGREE AND PG COURSES (AUTONOMOUS)
# ENGINEERING AND TECHNOLOGY PROGRAM
# RUSHIKONDA CAMPUS, VISAKHAPATNAM–530045.

**Cycle I: Practicing UML diagrams using IBM Rational Rose/Visual Paradigm**

Before developing a mini-project, in this cycle, the student is acquainted with different UML diagrams using Rational Rose. The experiments should include drawing UML diagrams listed below for two demo/example applications assigned by the lab Instructor. The input for the following experiments is problem statement for any two demo projects supplied by the instructor.

1. Introduction to Rational Rose and practicing the following diagrams
   a. Activity diagrams for the overall business process of the projects
   b. Use-case diagram for the demo projects along with Use-case descriptions and sub-diagrams for Use-cases.

2. Class diagram- Class diagrams including the features like classes, relationships, attributes and methods along with their visibilities.

3. Interaction diagrams- Sequence diagrams and Collaboration diagrams for different scenarios of the systems with all features like actors, objects and interactions.

4. Activity diagrams, State chart and other diagrams - Activity diagrams including the features like fork join and swim lanes. State diagrams including composite states and transitions. Component diagrams, Package diagrams and Deployment diagrams.

5. Forward and Reverser Engineering- Forward Engineering Class diagrams to classes in C++ and java and persistent classes to a database. Reverse Engineering C++ code, java code and a database.

6. Documentation using Rational Rose clear quest.

**Cycle II: Mini-Project**

The project deliverables include

- Problem statement
- Requirements Analysis
- Design
- A Software Design Description and a System Design.
- A test specification.
- Implementation
- Implement the assigned project with one of the following web technologies

**Front end:** Java technologies/PHP/MS.NET Technologies
**Backend:** Oracle/My-SQL/SQL-ServerTesting

**REFERENCE BOOKS:**

1. Project-based software engineering: An Object-oriented approach, EvelynStiller, Cathie LeBlanc, Pearson Education
2. Visual Modeling with Rational Rose 2002 and UML, Terry Quatrini, Pearson Education.

| Course Code | Title of the Course | Contact hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Internal | External | |
| CSM3108 | Web Technologies | 1 | 0 | 2 | 50 | 50 | 2 |

**COURSE OBJECTIVES:**

1. Understand client-server architecture and its role in web application development.
2. Develop web applications using HTML and PHP technologies.
3. Gain project-based experience to prepare for a career in web application development.

**COURSE OUTCOMES:**

At the end of the Course student will be able to
**CO 1: Design** static and dynamic web pages using HTML, CSS and JavaScript.
**CO 2: Develop** Server Side Scripts for insert, update, modify, and retrieve data from the server using PHP for an Application.

## LIST OF EXPERIMENTS

**Week 1 & Week 2:**
**HTML Tags:**

1. Design static web pages for home page that comprises of 3 frames. Top frame consists of Logo and title of the web page. Left frame comprises of links to different web pages and Right frame is used to display the content of web pages.
2. Left frame has links to Registration page, Login page, Contact us etc..
3. Login page has username and password fields along with submit button, forgot password and sign up hyperlinks.
4. Registration page has username, password, confirm password, email-id, Mobile Number, Date of birth, Address, Gender fields, submit button etc.

**Week 3:**
**CSS:**

5. Apply styles to web pages using inline.
6. Apply styles to web pages using embedded.
7. Apply styles to web pages using external style sheets.

**Week 4 and Week 5:**
**JAVA SCRIPT:**

8. Create a form similar to the one in previous experiment. Put validation checks on values entered by the user using JavaScript (such as age should be a value between 1 and 150).
9. Write a JavaScript program to display information box as soon as page loads.
10. Write a JavaScript program to change background color after 5 seconds of page load.

11. Write a JavaScript program to dynamically bold, italic and underline words and phrases based on user actions.
12. Write a JavaScript program to display a hidden div (e.g. showing stats of a player when user clicks on his name).
13. MySql Queries.

**Week 6 and Week 7:**
**PHP:**

14. Basic PHP Programs.

**Week 8 and Week 9:**
**PHP and MySQL:**

15. Perform insert, update, and retrieval and delete a record from database using Php and HTML.

**Week 10:**

16. Application

**Text Book:**

1. Programming the World Wide Web, 8th Edition, Robert W. Sebesta, Pearson.
2. Introduction to JavaScript Object Notation by Lindsay Bassett, O'Reilly Media, 2015.
3. Learning PHP, MySql, Robin Nixon.

**Reference Books:**

1. Web Programming, building internet applications, 2nd Ed., Chris Bates, Wiley Dreamtech.
2. Programming PHP, Kevin Tatroe, Peter MacIntyre & Rasmus Lerdorf foreword by Michael Bourque.

# GAYATRI VIDYA PARISHAD COLLEGE FOR DEGREE AND PG COURSES (AUTONOMOUS)
# ENGINEERING AND TECHNOLOGY PROGRAM
# RUSHIKONDA CAMPUS, VISAKHAPATNAM–530045.

| Course Code | Title of the Course | Contact hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Internal | External | |
| CSM3109 | Soft Skills And Technical Communication | 2 | 0 | 0 | 100 | 00 | 0 |

| | | |
|---|---|---|
| **Prerequisite (s)** | Knowledge of communication skills, work ethic, leadership, personal responsibility, empathy, leadership, sense of responsibility, integrity, self-esteem, self-management, motivation, flexibility, sociability, time management and making decisions. | **Ext. Exam Time 3 Hrs.** |

## COURSE OBJECTIVES

- By the end of the soft skills training program, the students should be able to:
- Develop effective communication skills (spoken and written).
- Develop effective presentation skills.
- Conduct effective business correspondence and prepare business reports which produce results

## COURSE OUTCOMES:

**CO 1: Develop** effective communication skills (spoken and written)..
**CO 2: Develop** effective presentation skills.
**CO 3: Conduct** effective business correspondence and prepare business reports which produce results

## UNIT-I

**Introduction to Soft Skills**: Communication – Verbal and Non-Verbal Communication - Personal grooming (Etiquette, Attitude, Body Language), Posture, Gestures, Facial Expressions, Eye Contact, Space Distancing, Presentation Skills, Public Speaking, Just a Minute (JAM) sessions, Adaptability.

## UNIT-II

**Goal Setting and Time Management:** Immediate, Short term, Long term, Smart Goals, Strategiesto Achieve goals, Types of Time, Identifying Time Wasters, Time Management Skills, Stress Busters.

## UNIT-III

**Leadership and Team Management:** Qualities of a Good Leader, Team Dynamics, Leadership Styles, Decision Making, Problem Solving, Negotiation Skills.

## UNIT-IV

**Group Discussions:** Purpose (Intellectual ability, Creativity, Approach to a problem, Tolerance), Group Behaviour, Analysing Performance.

## UNIT-V

**Job Interviews:** Identifying job openings, Covering Letter and CVs / Resumes, Interview (Opening, Body-Answer Q, Close-Ask Q), Telephone Interviews, Types of Questions.

## REFERENCE BOOKS:

1. English for Careers (ISBN: 9788131768846), Rs 150
2. Communication Skills and Soft Skills (ISBN: 9788131734537), Rs 160
3. Communicative English for Engineers and Professionals (ISBN: 9788131732045), Rs 190
4. Effective Communication and Soft Skills (ISBN: 9788131760345), Rs 245

# GAYATRI VIDYA PARISHAD COLLEGE FOR DEGREE AND PG COURSES (AUTONOMOUS)
# ENGINEERING AND TECHNOLOGY PROGRAM
# RUSHIKONDA CAMPUS, VISAKHAPATNAM–530045.

| Course code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int. | Ext | |
| Honors 3-1 | Recommender Systems | 3 | 1 | 0 | 30 | 70 | 4 |

## COURSE OBJECTIVES:

1. To understand the fundamental concepts and techniques used in building recommender systems.
2. To analyze and evaluate different recommendation algorithms for their effectiveness and applicability.
3. To develop the ability to implement and optimize various types of recommender systems, including collaborative filtering, content-based, and hybrid methods.
4. To explore advanced topics in recommender systems, such as handling the cold-start problem, privacy concerns, and incorporating contextual information.

## COURSE OUTCOMES:

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

**CO1.** Demonstrate a thorough understanding of the various models and approaches used in recommender systems.

**CO2.** Critically assess the strengths and weaknesses of different recommender system algorithms.

**CO3.** Design and implement a recommender system using appropriate tools and techniques.

**CO4.** Solve practical problems related to recommendation, such as data sparsity and scalability issues.

**CO5.** Apply advanced methods to enhance the performance of recommender systems in specific domains like social networks or e-commerce.

### UNIT 1: Introduction to Recommender Systems

Goals and applications of recommender systems, Basic models: Collaborative filtering, content-based, and knowledge-based systems, Hybrid and ensemble-based recommender systems, Evaluation metrics and methodologies, Challenges in recommender systems: Context, time-sensitivity, location, and social factors.

**UNIT 2: Neighborhood-Based Collaborative Filtering**

User-based and item-based neighborhood models, Similarity functions and prediction techniques, Clustering and dimensionality reduction, Graph models for neighborhood-based methods, Regression modeling and sparse linear models.

**UNIT 3: Model-Based Collaborative Filtering**

Decision trees, rule-based, and naive Bayes methods, Latent factor models and matrix factorization, Singular value decomposition (SVD) and non-negative matrix factorization, Integrating factorization with neighborhood models, Advanced techniques: Regularization, handling implicit feedback, and optimization methods.

**UNIT 4: Content-Based Recommender Systems**

Feature extraction and representation, Supervised feature selection and weighting Learning user profiles and preference modeling, Example applications: Product, web page, and music recommendations, Addressing challenges: Cold-start problem and overfitting.

**UNIT 5: Advanced Topics and Applications**

Privacy and security in recommender systems, Attack-resistant and trustworthy recommendations, Group recommender systems and multi-criteria recommendations, Active learning and real-time recommendations, Case studies: E-commerce, social networks, and personalized content delivery.

## Textbooks

1. "Recommender Systems: The Textbook" by Charu C. Aggarwal, Springer, 2016.
2. "Introduction to Recommender Systems: Algorithms and Technologies" by Dietmar Jannach, Markus Zanker, Alexander Felfernig, Gerhard Friedrich, Cambridge University Press, 2010.

## Reference Books

1. "Matrix Factorization Techniques for Recommender Systems" by S. Rendle, Morgan & Claypool Publishers, 2021.
2. "Personalized Recommender Systems: An Algorithmic Perspective" by Mouzhi Ge, Springer, 2022.
3. "Practical Recommender Systems" by Kim Falk, Manning Publications, 2019.

# B. Tech Computer Science and Engineering (AI and ML)
## (R-22 Regulation)
### III Year – II Semester

| Course code | Category | Course Title | Hours per week | | | Internal Marks | External Marks | Total Marks | Credits |
|---|---|---|---|---|---|---|---|---|---|
| | | | L | T | P | | | | |
| CSM3201 | PC/PCC | Machine Learning | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM3202 | PC/PCC | Soft Computing | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM3203 | PC/PCC | Automata Theory & Compiler Design | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM3204 | OEC/JOE | Open Elective - II | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM3205 | PEC | Elective - II | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM3206 | PC/PCC | Machine Learning Lab | 0 | 0 | 3 | 50 | 50 | 100 | 1.5 |
| CSM3207 | PC/PCC | Soft Computing Lab | 0 | 0 | 3 | 50 | 50 | 100 | 1.5 |
| CSM3208 | PEC | Elective – II Lab | 0 | 0 | 3 | 50 | 50 | 100 | 1.5 |
| CSM3209 | SAC/SC | Skill Course – 4 Android Programming | 1 | 0 | 2 | 50 | 50 | 100 | 2 |
| CSM3210 | MC | Essence of Indian Traditional Knowledge | 3 | 0 | 0 | 30 | 70 | 100 | 0 |
| | | | | | | | **Total Credits** | | **21.5** |
| **Summer Industrial Research Internship (2 months) Mandatory** | | | | | | | | | |

| Title of the Program | L | | P | Credits |
|---|---|---|---|---|
| **Honors/Minor Courses (The hours distribution can be 3-0-2 or 3-1-0 also)** | **3** | **1** | **0** | **4** |

**Elective-II**

**Introduction to Arduino and Raspberry PI**
Big Data Analytics
Cryptography & Network Security

**Open Elective – II**

Offered by Mechanical Engineering
Offered by Civil Engineering
Offered by ECE
Offered by CSE
Offered by CSE(AI&ML): Introduction to Machine Learning

# GAYATRI VIDYA PARISHAD COLLEGE FOR DEGREE AND PG COURSES (AUTONOMOUS)
# ENGINEERING AND TECHNOLOGY PROGRAM
# RUSHIKONDA CAMPUS, VISAKHAPATNAM–530045.

| Course Code | Title of the Course | Contact hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| CSM 3201 | MACHINE LEARNING | L | T | P | Internal | External | 3 |
| | | 3 | 0 | 0 | 30 | 70 | |

## COURSE OBJECTIVES:

- Introduce The Fundamental Concepts And Types Of Machine Learning.
- Develop Skills To Create And Evaluate Machine Learning Models.
- Understand And Apply Classification And Regression Algorithms.
- Explore And Analyze Clustering Techniques In Unsupervised Learning.
- Gain Foundational Knowledge Of Neural Networks And Deep Learning.

## COURSE OUTCOMES:

At the end of the course the student is able to

**CO1:** Understand the concepts of Machine Learning and its types
**CO2:** Develop Machine Learning Models
**CO3:** Apply Classification Algorithms
**CO4:** Analyze Clustering Techniques
**CO5:** Create Neural Networks

### UNIT – I - Introduction to Machine Learning 8 Hours

What is Human Learning? – Types of Human Learning – What is Machine Learning? – Types of Machine Learning – Problems not to be solved using Machine Learning – Applications of Machine Learning – Tools in Machine Learning – Issues in Machine Learning.

### UNIT – II – Preparing to Model & Modelling and Evaluation 12 Hours

Introduction - Machine Learning Activities – Basic Types of data in Machine Learning – Exploring Structure of Data – Data Quality and Remediation – Data Pre-Processing.

Selecting a Model – Training a Model (for Supervised Learning) – Model Representation and Interpretability – Evaluating Performance of a Model – Improving performance of a model.

### UNIT – III – Supervised Learning: Classification & Regression 10 Hours

Introduction – Example of Supervised Learning – Classification Model – Classification Learning steps – Common Classification Algorithms - Example of Regression – Common Regression Algorithms

### UNIT – IV – Unsupervised Learning 8 Hours

Introduction – Unsupervised vs Supervised Learning – Applications of Unsupervised Learning – Clustering – Types of Clustering - Finding Pattern using Association Rule.

**UNIT – V – Basics of Neural Network** **12 Hours**

Introduction – Understanding the Biological Neuron – Exploring the Artificial Neuron – Types of Activation Functions – Early Implementation of ANN – Architectures of Neural Network – Learning Process in ANN – Gradient Descent - Back Propagation – Deep Learning.

**TEXT BOOKS:**

1. Machine Learning by Saikat Dutt, Subramanian Chandramouli & Amit Kumar Das, Pearson.

**REFERENCE BOOKS:**

1. Introduction to Machine Learning - Alex Smola and S.V.N. Vishwanathan, Cambridge University Press.
2. Introduction to Machine Learning, 3rd Edition by Ethem Alpaydin, PHI
3. Machine Learning – Tom M Mitechell, Tata Mc Graw hill

# GAYATRI VIDYA PARISHAD COLLEGE FOR DEGREE AND PG COURSES (AUTONOMOUS)
## ENGINEERING AND TECHNOLOGY PROGRAM
## RUSHIKONDA CAMPUS, VISAKHAPATNAM–530045.

| Course Code | Title of the Course | Contact hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Internal | External | |
| CSM 3202 | SOFT COMPUTING | 3 | 0 | 0 | 30 | 70 | 3 |

**COURSE OBJECTIVES:**

- Understand the fundamental concepts and differences between hard computing and soft computing.
- Explore fuzzy logic, fuzzy inference systems, and their applications in decision-making processes.
- Learn the architecture and design principles of multilayer feed-forward neural networks and radial basis function neural networks.
- Gain knowledge of evolutionary computing techniques, including genetic algorithms and their applications.
- Explore advanced machine learning techniques, including support vector machines, Bayesian networks, and Markov models.

**COURSE OUTCOMES:** At the end of the course the student is able to

**CO1:** Understand the concepts of Soft Computing
**CO2:** Develop Fuzzy Inference Models
**CO3:** Apply Back Propagation algorithm
**CO4:** Analyze Evolutionary Computing models
**CO5:** Create Markov Models

**UNIT – I - Introduction to Soft Computing** **8 Hours**

Introduction – Hard Computing vs Soft Computing – Constituents of Soft Computing – Crisp and Fuzzy Sets – Classical or Crisp Set Theory – Crisp Relations and Operations – Fuzzy Sets Theory.

**UNIT – II – Fuzzy Logic and Inference Rules** **10 Hours**

Introduction – Classical Logic – Multi-valued Logic – Fuzzy Logic – Fuzzy Propositions – Inference Rules for Fuzzy Propositions – Fuzzy Inference System – Types of Fuzzy Inference Engines – Implementation of Fuzzy Inference Engine – Neuro – Fuzzy System

**UNIT – III – Multilayer Feed- forward Neural Network** **10 Hours**

Multilayer FFNN Architecture – Learning Methods – Back propagation Method - Design Issues of ANN – Applications of FFNN – Radial Basis Function Neural Network – Architecture of RBFNN – Learning of RBFNN – XOR Problem in RBFNN – Comparison of RBFNN with FFNN.

**UNIT – IV – Introduction to Evolutionary Computing** **12 Hours**

Evolutionary Algorithm – Swarm Intelligence – Biological Evolutionary Process- Paradigms of Evolutionary Computing – Evolutionary Strategies – Evolutionary Programming – Advantages of Evolutionary Computation – Applications of Evolutionary Algorithms – Genetic Algorithm – Selection of parents – Encoding of Genetic Operators – Classification of Genetic Algorithm – GA Example implementation – Applications, Advantages and Disadvantages of GA.

**UNIT – V – Advanced Machine Learning Techniques** **10 Hours**

Introduction – Support Vector Machines – Advantages and Disadvantages of SVM –Bayes' Theorem, Bayesian Belief Network – Dempster – Shafer Theory – Certainty Factor Model – Markov Models – DMM - HMM

**TEXT BOOKS:**

1. Soft Computing Fundamentals, Techniques and Applications by Saroj Kaushik, Sunita Tiwari –Mc Graw Hill

**REFERENCE BOOKS:**

1. Neural Networks, Fuzzy Logic, and Genetic Algorithms: Synthesis and Applications by S. Rajasekaran and G. A. Vijayalakshmi Pai, Prentice Hall of India, 2003.
2. Principles of Soft Computing by S. N. Sivanandam and S. N. Deepa, Wiley India Pvt Ltd, 2011.
3. Soft Computing and Intelligent Systems Design: Theory, Tools, and Applications by Fakhreddine O. Karray and Clarence De Silva, Pearson, 2004.

# GAYATRI VIDYA PARISHAD COLLEGE FOR DEGREE AND PG COURSES (AUTONOMOUS)
## ENGINEERING AND TECHNOLOGY PROGRAM
## RUSHIKONDA CAMPUS, VISAKHAPATNAM–530045.

| Course Code | Title of the Course | Contact hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Internal | External | |
| CSM 3203 | Automata Theory and Compiler Design | 3 | 0 | 0 | 30 | 70 | 3 |

**COURSE OBJECTIVES:**

1. Understand automata theory, finite state machines, and their optimization, along with compiler design principles, including language processors, programming language basics, and the tools and structure for effective compilation.
2. Learn to define, manipulate, and optimize regular expressions and regular grammars, convert them to and from finite state machines using techniques like Arden's theorem, and understand lexical analysis, including token specification, recognition, and input buffering with tools like LEX.
3. Comprehend the definition and design of context-free grammars, including derivations, parse trees, ambiguity, and minimization of CFGs, as well as the role of parsers and top-down parsing in the syntax analysis phase of compilers.
4. Understand Pushdown Automata including their definition, construction, conversion with Context-Free Grammars, and types, as well as to explore syntax analysis in compilers, focusing on bottom-up and LR parsing methods.
5. Understand Turing Machines and their types, along with Universal Turing Machines, and learn about intermediate code generation and optimization, including syntax trees, three-address code, and optimization techniques.

**COURSE OUTCOMES:**

**CO 1: Construct** Optimized Finite State Machine for any given Finite State Machine and **Analyze** the working of various phases of Compilers for a given high level language instruction.
**CO 2: Design** a Finite State Machine for the given Regular Expression and vice-versa and **Analyze** the working of Lexical Analyzer for a given high level language instruction.
**CO 3: Construct** a Minimized Context Free Grammar for a given Context Free Grammar and **apply** the working of a Top-Down parser for a given Context Free Grammar.
**CO 4: Design** the basic Push down Automata for a given Context Free Language and **Apply** the role of a Bottom – up parser for a given Context Free Grammar.
**CO 5: Demonstrate** the working of Push down Automata and **Determine** the object code for the given three address code.

**UNIT I**

**Introduction Finite Automata**: Basic concepts of Automata Theory, grammar, types of grammar, types of machines, Finite State Machines, Types of finite state machines, representations in mathematical diagram, tabular etc., id of finite state machines, indefinite statemachine to definite state machine, elimination of ϵ-transitions, optimization of finite state machine.

**Introduction to Compiler Design:** Introduction to Compilers and Language processors, Programming Language basics, Structure & Different Phases of a Compiler, Review of Compiler Structure, Structure of Optimizing Compilation, Compiler construction tools.

## UNIT II

**Regular Expressions and Regular Grammar:** Definition of regular expression, regular algebra, minimization of regular expressions, closure properties, construction of regular expression from the given description, regular expression to finite state machine, finite state machine to regular expression, construction of regular expression for the given finite state machine- a systematic way using Arden's theorem

**Lexical Analysis:** Role of Lexical Analyzer, Input Buffering, Specification of Token, Recognition of Token, LEX Program

## UNIT III

**Context Free Grammars:** Definition and designing CFGs, Derivations Using a Grammar, Parse Trees, Ambiguity and elimination of Ambiguity, Minimization of CFGs.

**Syntax Analysis Phase of Compilers: Part-1:** Role of Parser, Top-Down Parsing, Top-Down Parsers.

## UNIT IV

**Push Down Automata:** Definition of the Pushdown Automata, Construction of PDAs, Push down Automata to CFG, CFG to PDA, and Types of Push down Automata's.

**Syntax Analysis Phase of Compilers:** Part-2: Bottom-up Parsing, Introduction to LR Parsing: SLR, More Powerful LR parsers

## UNIT V

**Introduction to Turing Machine:** Definition of the Turing Machine, Types of Turing Machine, Universal Turing Machine.

**Intermediate Code Generation and Code Optimization**: Intermediate Codes, Syntax Trees, Intermediate Code Generation, and Three Address Code-Translation of Expressions, Machine Dependent Optimization and Independent Code Optimization.

**Text Book:**

1. Introduction to automata theory, languages and computation, John.E.H.P croft/ Rajeev Motwani & JD Ullman—pearson education- III edition
2. Principles of Compiler Design by Aho,D. Ullman, Lam and Ravi Sethi, Pearson Education Second Edition

**Reference Books:**

1. Theory of computation, formal languages and automata theory, G P Saradhi Varma, B.Thirupathi Rao –Sci Tech publications
2. Compiler Design, A.A. Pentambekar, Technical Publications

# GAYATRI VIDYA PARISHAD COLLEGE FOR DEGREE AND PG COURSES (AUTONOMOUS)
## ENGINEERING AND TECHNOLOGY PROGRAM
## RUSHIKONDA CAMPUS, VISAKHAPATNAM–530045.

| Course Code | Title of the Course | Contact hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int | Ext | |
| CSM 3204 | Introduction to Machine Learning | 3 | 0 | 0 | 30 | 70 | 3 |

**COURSE OBJECTIVES**

1. Introduce students to the basic concepts, types, applications, and tools of machine learning.
2. Equip students with skills in data preparation, model selection, and performance evaluation.
3. Teach principles and techniques of supervised learning, including classification and regression.
4. Introduce unsupervised learning methods, including clustering and association rules.
5. Provide understanding of advanced learning techniques like ensemble and reinforcement learning and their impact on model performance.

**COURSE OUTCOMES:**

At the end of the course the student is able to

**CO1:** Explain the fundamentals of machine learning, including its types, applications, and tools.
**CO2:** Prepare and pre-process data for machine learning models. Techniques
**CO3:** Develop and evaluate supervised learning models using classification and regression algorithms.
**CO4:** Implement unsupervised learning techniques such as clustering and association rules to find patterns in data.
**CO5:** Apply advanced learning techniques such as ensemble learning and reinforcement learning to improve model performance.

**UNIT – I: Introduction to Machine Learning**

What is Human Learning, Types of Human Learning, What is Machine Learning, Types of Machine Learning, Problems not to be solved using Machine Learning, Applications of Machine Learning, and Tools in Machine Learning, Issues in Machine Learning.

**UNIT – II: Preparing to Model & Modelling and Evaluation**

Introduction, Machine Learning Activities, Basic Types of data in Machine Learning, Exploring Structure of Data, Data Quality and Remediation, Data Pre-Processing, Selecting a Model, Training a Model (for Supervised Learning), Model Representation and Interpretability, Evaluating Performance of a Model, Improving performance of a model.

**UNIT – III: Supervised Learning: Classification & Regression**

Introduction, Example of Supervised Learning, Classification Model, Classification Learning steps, Common Classification Algorithms, Example of Regression, and Common Regression Algorithms.

**UNIT – IV: Unsupervised Learning**

Introduction, Unsupervised vs. Supervised Learning, Applications of Unsupervised Learning, Clustering, Types of Clustering, Finding Pattern using Association Rule.

**UNIT – V: Other Types of Learning**

Ensemble Learning- Bagging, Boosting, Stacking and its impact on bias and variance, Ada Boost, Gradient Boosting Machines, XG Boost. Reinforcement Learning - Introduction, Q Learning

**TEXT BOOKS:**

1. Machine Learning by Saikat Dutt, Subramanian Chandramouli & Amit Kumar Das, Pearson.

**REFERENCE BOOKS:**

1. Introduction to Machine Learning - Alex Smola and S.V.N. Vishwanathan, Cambridge University Press.
2. Introduction to Machine Learning, 3rd Edition by Ethem Alpaydin, PHI
3. Machine Learning – Tom M Mitechell, Tata Mc Graw hill

| Course Code | Title of the Course | Contact hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| CSM3205 | Introduction to Arduino and Raspberry PI | L | T | P | Internal | External | 3 |
| | | 3 | 0 | 0 | 30 | 70 | |

## COURSE OBJECTIVES:

1. To train students to be equipped with a solid theoretical foundation, systematic professional knowledge and strong practical skills in the Raspberry Pi and Arduino.
2. To focus on higher-level operating systems, advanced networking, user interfaces, multimedia and uses more computing intensive IoT applications as examples using Raspberry Pi running Linux as the platform of choice.
3. To expose the student to other comparable platforms like Arduino.
4. To design and deploy multiple IoT devices that could connect to the gateway.

## COURSE OBJECTIVES:

CO1: Understanding basic concepts of Arduino and Arduino IDE.
CO2: Familiar with Arduino libraries and Arduino programming.
CO3: Examine Raspberry Pi board features and familiar with basic concepts of Raspbian Linux.
CO4: Acquiring basic foundations of Python Programming and libraries for Raspberry Pi.
CO5: Develop IoT applications using Raspberry Pi board.

## UNIT- I

The Arduino Environment: Introduction to the Arduino environment, the Arduino board, the Arduino IDE, and the Arduino- compatible shields together with their libraries. Arduino board main components, inputs, and outputs. Arduino Integrated Development Environment (IDE), Compiling Code, Arduino Shields and Libraries.

## UNIT-II

Basics of C programming, composition of an Arduino programs, Arduino tool chain, Arduino IDE, basic structure of a sketch, including the use of the setup() and loop() functions. Accessing the pins from a sketch for input and output, introduction on debugging embedded software on an Arduino, UART communication protocol, Synchronization, parity and stop, the use of the Serial library to communicate with the Arduino through the serial monitor.

## UNIT-III

Getting Started with Raspberry Pi Basic functionality of Raspberry Pi B+ board, setting up the board, configuration and use, implications of an operating system on the behaviour of the Raspberry Pi as an IoT device, booting Raspberry Pi 3, Downloading an Operating System, format an SD card and 15 booting the OS Basics of Linux and its use, main features including navigating the file system and managing processes, text-based user interface through the shell

## UNIT- IV

Interfacing Hardware with the Raspberry Pi, Raspberry Pi Remote Access, operate the Raspberry Pi in "headless mode", Bash Command line, operating Raspberry Pi without needing a GUI interface, Basics of the Python programming language, programming on the Raspberry Pi. Python on Raspberry Pi, Python Programming Environment, Python Expressions, Strings, Functions and Function arguments, Lists, List Methods, Control Flow.

## UNIT-V

Communication with devices through the pins of the Raspberry Pi, RPi.GPIO library, Python Functions, setting up the pins, General purpose IO Pins, Protocol Pins, GPIO Access, applying digital voltages, and generating Pulse Width Modulated signals, Tkinter Python library, accessing pins through a graphic user interface.

### TEXT BOOKS:

1. McGraw Hill Professional Massimo Banzi, "Getting Started with Arduino", First Edition, February 2009, O'Reilly Media, Inc.
2. Simon Monk, "Programming the Raspberry Pi: Getting Started with Python", January 2012.

### REFERENCE BOOKS:

1. Eben Upton and Gareth Halfacree, "Raspberry Pi User Guide", August 2016, 4th edition, JohnWiley & Sons.
2. Alex Bradbury and Ben Everard, "Learning Python with Raspberry Pi", Feb 2014, John Wiley & Sons.
3. Michael Margolis, "Arduino Cookbook", First Edition, March 2011, O'Reilly Media, Inc

# GAYATRI VIDYA PARISHAD COLLEGE FOR DEGREE AND PG COURSES (AUTONOMOUS)
## ENGINEERING AND TECHNOLOGY PROGRAM
## RUSHIKONDA CAMPUS, VISAKHAPATNAM–530045.

| Course Code | Title of the Course | Contact hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| CSM3205 | BIG DATA ANALYTICS | L | T | P | Internal | External | 3 |
| | | 3 | 0 | 0 | 30 | 70 | |

**COURSE OBJECTIVES:**

1. Understand the Big Data Platform and its Use cases
2. Provide an overview of Apache Hadoop
3. Provide HDFS Concepts and Interfacing with HDFS
4. Understand Map Reduce Jobs

**COURSE OUTCOMES:**

At the end of the course student will be able to

CO1: Understand concept of Big Data and Hadoop Eco System
CO2: Configure various Hadoop services in distributed environment
CO3: Analyze unstructured data using Map Reduce
CO4: Understand various advanced Map Reduce tasks for analyzing the data
CO5: Solve various real times problems using Hadoop and HBase

### UNIT-I

**Introduction to Big Data**: Big Data-definition, Characteristics of Big Data (Volume, Variety, Velocity, Veracity, Validity), Importance of Big Data, Data in the Warehouse and Data in Hadoop, Algorithms using map reduce, Matrix-Vector Multiplication by Map Reduce. Introduction to Hadoop: Hadoop- definition, understanding distributed systems and Hadoop, Comparing SQL databases and Hadoop

### UNIT-II

**Hadoop Architecture**: History of Hadoop, building blocks of Hadoop, NameNode, DataNode, Secondary NameNode, JobTracker and Task Tracker, YARN. Understanding MapReduce, Word count program using traditional method and conventional methods Components of Hadoop - Working with files in HDFS, Reading and writing the Hadoop Distributed File system –The Design of HDFS, HDFS Concepts, The Command-Line Interface, Hadoop commands , Hadoop Filesystem

### UNIT-III

**MapReduce:** Hadoop EcoSystem – Moving Data in and out of Hadoop – Understanding inputs and outputs of MapReduce . Anatomy of a MapReduce program, A Weather Dataset,

Analyzing the Data with UnixTools, Analyzing the Data with Hadoop, Scaling Out, Hadoop Streaming, Hadoop Pipes, Hadoop Archives, Getting the patent data set, constructing the basic template of a Map Reduce program

**UNIT-IV**

**MapReduce Advanced Programming**: Advanced MapReduce - Chaining Map Reduce jobs, joining data from different sources, creating a Bloom filter, passing job-specific parameters to your tasks, probing for task-specific information, partitioning into multiple output files, inputting from and outputting to a database, keeping all output in sorted order.

**UNIT-V**

**Graph Representation in Map Reduce**: Modeling data and solving problems with graphs, Shortest Path Algorithm, Friends-of-Friends Algorithm, PageRank Algorithm, Bloom Filter, Zookeeper – how it helps in monitoring a cluster, HBase uses Zookeeper and how to Build Applications with Zookeeper.

**TEXT BOOKS**:

1. Dirk deRoos, Chris Eaton, George Lapis, Paul Zikopoulos, Tom Deutsch "Understanding Big Data Analytics for Enterprise Class Hadoop and Streaming Data", 1st Edition, TMH,2012.
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 in Practice by Alex Holmes, MANNING Publishers
3. Mining of massive datasets, AnandRajaraman, Jeffrey D Ullman, Wiley Publications.
4. Boris lublinsky, Kevin t. Smith, Alexey Yakubovich, "Professional Hadoop Solutions",Wiley,ISBN:9788126551071,2015.
5. Big Data Black Book ( Covers Hadoop 2, Map Reduce, Hive, Yarn, Pig & amp; Data Visualization ) - Dream Tech Publications
6. Chris Eaton, Dirk deroos et al. , "Understanding Big data ", McGraw Hill, 2012.
7. Tom White, "HADOOP: The definitive Guide" , O Reilly 2012.
8. Vignesh Prajapati, "Big Data Analytics with R and Haoop", Packet Publishing 2013.
9. Tom Plunkett, Brian Macdonald et al, "Oracle Big Data Handbook", Oracle Press, 2014.
10. Jy Liebowitz, "Big Data and Business analytics", CRC press, 2013.

# GAYATRI VIDYA PARISHAD COLLEGE FOR DEGREE AND PG COURSES (AUTONOMOUS)
# ENGINEERING AND TECHNOLOGY PROGRAM
# RUSHIKONDA CAMPUS, VISAKHAPATNAM–530045.

| COURSE CODE | TITLE OF THE COURSE | CONTACT HOURS/WEEK | | | ALLOTMENT OF MARKS | | CREDITS |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int. | Ext | |
| CSM3205 | Cryptography & Network Security | 3 | 0 | 0 | 30 | 70 | 3 |

**COURSE OBJECTIVES:**

1. To introduce several issues in network security- its need and importance, taxonomy and terminology.
2. To learn various cryptographic techniques.
3. To understand Internet security protocols and standards.
4. To design security applications in the field of Information technology.

**COURSE OUTCOMES:**

At the end of the course student will be able to

CO1: Classify network security attacks, services, mechanisms and classical encryption techniques.
CO2: Apply symmetric/asymmetric key cryptographic techniques to ensure privacy of data in transit.
CO3: Describe symmetric keys distribution techniques and public key Infrastructure (PKI).
CO4: Design new cryptographic protocols for different security applications.
CO5: Discuss intrusion detection techniques, Firewalls and malicious software.

## UNIT-I

**Overview**: Computer Security Concepts, the OSI Security Architecture, Security Attacks, Security Services, Security Mechanisms, a Model for Network Security, Basics of Buffer Overflow**,** Software Security Issues**. Classical Encryption Techniques**: Symmetric Cipher Models, Substitution Techniques, Transposition techniques, Introduction to Steganography.

## UNIT-II

**Block Ciphers and the Data Encryption Standard**: Stream Ciphers and Block Ciphers, the Data Encryption Standard (DES), A DES Example, the Strength of DES. **Advanced Encryption Standard**: AES Structure, AES Transformation Functions, AES Key Expansion. **Block Cipher Operations**: Electronic Code Book, Cipher Block Chaining Mode, Cipher Feedback Mode, Output Feedback Mode, Counter Mode. **Public-Key Cryptography and RSA**: Principles of Public Key Cryptosystems, the RSA Algorithm. **Other Public-Key Cryptosystems:** Diffie-Hellman Key Exchange, Elliptic curve Cryptography.

## UNIT-III

**Cryptographic Hash Functions**: Applications of Cryptographic Hash Functions, Secure Hash Algorithm (SHA-512). **Digital Signatures:** Digital Signatures, NIST Digital Signature Algorithm. **Key Management and Distribution:** Symmetric Key Distribution using Symmetric Key Encryption, Symmetric Key Distribution using Asymmetric Key Encryption, Distribution of public Keys, X.509 Certificates, Public-Key Infrastructure.

## UNIT-IV

**Transport-Level Security:** Web Security Considerations, Secure Socket Layer and Transport Layer Security, Transport Layer Security. **Electronic Mail Security**: Pretty Good Privacy, S/MIME. **IP Security**: Overview, IP Security Policy, Encapsulating Security Payload, Combining Security Associations.

## UNIT-V

**Malicious Software:** Types of Viruses, Virus Countermeasures, Worms, Distributed Denial Of Service Attacks. **Intruders:** Intruders, Intrusion Detection, Password Management. **Firewalls:** Need of Firewalls, Firewall Characteristics, Types of Firewalls, Configurations.

### TEXT BOOKS:

Cryptography and Network Security Principles and Practice, William Stallings, Seventh Edition, Pearson Education.

### REFERENCE BOOKS:

1. Computer Security - Principles and Practice, 4th Edition by William Stallings, Pearson Education.
2. Cryptography and Network Security, Atul Kahate, 4th Edition, Tata McGraw Hill Publications.
3. Cryptography and Network Security Behrouz A Frorouzan, Second Edition, Tata McGraw Hill Pub Company Ltd, New Delhi.

# GAYATRI VIDYA PARISHAD COLLEGE FOR DEGREE AND PG COURSES (AUTONOMOUS)
# ENGINEERING AND TECHNOLOGY PROGRAM
# RUSHIKONDA CAMPUS, VISAKHAPATNAM–530045.

| COURSE CODE | TITLE OF THE COURSE | CONTACT HOURS/ WEEK | | | ALLOTMENT OF MARKS | | CREDITS |
|---|---|---|---|---|---|---|---|
| CSM3206 | MACHINE LEARNING LAB | L | T | P | Int. | | 1.5 |
| | | 0 | 0 | 3 | 50 | | |

**COURSE OBJECTIVES:**

This LAB course will enable students to

1. Make use of Data sets in implementing the machine learning algorithms.
2. Implement the machine learning concepts and algorithms in any suitable language of choice.

**COURSE OUTCOMES:**

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

1. Understand the implementation procedures for the machine learning algorithms.
2. Design Java/Python programs for various Learning algorithms.
3. Apply appropriate data sets to the Machine Learning algorithms.
4. Identify and apply Machine Learning algorithms to solve real world problems.

Implement the experiments using Python

**List of Lab Experiments:**

1. 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.

2. Build an Artificial Neural Network by implementing the **Back-propagation algorithm** and test the same using appropriate data sets.

3. Write a program to implement the **naïve Bayesian classifier** for a sample training data set stored as a .CSV file. Compute the accuracy of the classifier, considering few test data sets.

4. Assuming a set of documents that need to be classified, use the **naïve Bayesian Classifier** model to perform this task. Built-in classes/API can be used to write the program. Calculate the accuracy, precision, and recall for your data set.

5. Write a 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. You can use Python ML library classes/API.

6. Apply **EM algorithm** to cluster a set of data stored in a .CSV file. 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 Python ML library classes/API in the program.

7. Write a program to implement ***k*-Nearest Neighbor algorithm** to classify the iris data set. Print both correct and wrong predictions. Python ML library classes can be used for this problem.

8. Implement the non-parametric **Locally Weighted Regression algorithm** in order to fit data points. Select appropriate data set for your experiment and draw graphs.

| COURSE CODE | TITLE OF THE COURSE | CONTACT HOURS/ WEEK | | | ALLOTMENT OF MARKS | | CREDITS |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int. | Ext | |
| CSM3207 | SOFT COMPUTING LAB | 0 | 0 | 3 | 50 | 50 | 1.5 |

## COURSE OBJECTIVES

1. To learn to implement soft computing methods.

2. To learn to solve the real world problem through program of Python

3. To learn to solve and optimize the real world problem using Python

## COURSE OUTCOMES

After completion of the course, the students would be able to:

1. Understand the concept and techniques of designing and implementing of soft computing methods in real world problems

2. Acquire the knowledge of the fuzzy Neural network and Genetic Language

3. Analyze and optimized the problem of real-life applications

**List of Programs**

1. Python programming fundamental.

2. Python introduction to numerical calculation programming (scitific python, Numerical python, Image processing).

3. Python programming to simulate a single layer neural network designs.

4. Python programming to simulate multiple layer neural network designs.

5. Python programming to observe the perceptron learning algorithm performances for a single layer network. In this experiment consider the XOR dataset.

6. Implementation of Simple Genetic Algorithm in python for solving optimization problem.

7. Write a python program to implement the different Fuzzy Membership functions.

8. Write a python program to implement Clustering K –Means and its properties.

9. Write a python program to implement Hierarchical Clustering.

10. Write a python program to implement Clustering Fuzzy C-Means.

**REFERENCE BOOKS:**

1. Principles of Soft Computing Using Python Programming: Learn How to Deploy Soft Computing Models in Real World Applications, Gypsy Nandi, November 2023 Wiley-IEEE Press.

# GAYATRI VIDYA PARISHAD COLLEGE FOR DEGREE AND PG COURSES (AUTONOMOUS)
# ENGINEERING AND TECHNOLOGY PROGRAM
# RUSHIKONDA CAMPUS, VISAKHAPATNAM–530045.

| COURSE CODE | TITLE OF THE COURSE | CONTACT HOURS/ WEEK | | | ALLOTMENT OF MARKS | | CREDITS |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int. | Ext | |
| CSM3208 | Introduction to Arduino and Raspberry PI Lab | 0 | 0 | 3 | 50 | 50 | 1.5 |

## COURSE OBJECTIVES:

1. To learn the basics concept Internet of Things.
2. To study design principles.
3. To understand different sensors in IoT environment
4. To study basic building blocks of IoT devices.

## COURSE OUTCOMES:

At the end of the Course, the Student will be able to:

**CO1:** Build basic prototypes using Arduino Uno.

**CO2:**Use different types of sensors, actuators for Arduino Uno.

**CO3:** Demonstrate the setup and Installation procedure of Raspberry Pi.

**CO4:** Build prototypes using Raspberry pi with different communicationprotocols.

**CO5:** Design an interface using Tkinter to control the IoT devices.

**LIST OF EXPERIMENTS: (Any 12 Experiments from the following to be performed)**

1. Install the Arduino and write a program using the Arduino IDE to blink LED.
2. Interface LED and buzzer with Arduino to buzz for a period of time.
3. Interface RGB LED with Aurdino to obtain different colours and brightness using PWM.
4. a) Control a servo motor using Arduino with an input given through a push button (e.g: When the push button is pressed the servo motor has to rotate by 15 degrees).

   b) Rotate Stepper motor either clockwise or anti clockwise at ‘n’ number of steps using Arduino.
5. Write a program to read the data from the RFID tag and display the information on the display board using Arduino and control LED (e.g: if it is a valid card then the LED should be ON otherwise OFF).
6. Control any two actuators connected to the Arduino using Bluetooth/Wifi.
7. Interface analog/digital sensors with Arduino and analyze the corresponding readings. (Sensors like temperature, alcohol, humidity, pressure, gas, sound pollution, level, weight, flow, proximity, LDR, PIR, pulse, vibration, sound etc..)
8. Demonstration of setup & working of Raspberry Pi. (Students have to prepare the report for the same).

9. Interface RGB LED with Raspberry Pi to obtain different colors and brightness using PWM.
10. a) Interface an ultrasonic sensor with Raspberry pi to print distance readings on the monitor when the sensor changes its position.

b) Reading the data from an analog sensor with Raspberry using Arduino serial port or ADC CP3208 using SPI.

11. Post/read the data to/from the cloud via MQTT broker with a Raspberry Pi.
12. Send real-time sensor data to a smartphone using Raspberry Pi onboard Bluetooth.
13. Interface Picamera module using Raspberry Pi to perform operations of PiCamera- API or OpenCV library.
14. Implement an intruder alert system that alerts through email
15. Implement remote monitoring of smoke alarm systems using Raspberry Pi.
16. Create a user interface using Tkinter to control the API's in Raspberry Pi.

## TEXT BOOKS:

1. McGraw Hill Professional Massimo Banzi, "Getting Started with Arduino", First Edition, February 2009, O'Reilly Media, Inc.
2. Simon Monk, "Programming the Raspberry Pi: Getting Started with Python", January 2012.

## REFERENCE BOOKS:

1. Eben Upton and Gareth Halfacree, "Raspberry Pi User Guide", August 2016, 4th edition, JohnWiley & Sons.
2. Alex Bradbury and Ben Everard, "Learning Python with Raspberry Pi", Feb 2014, John Wiley & Sons.
3. Michael Margolis, "Arduino Cookbook", First Edition, March 2011, O'Reilly Media, Inc

# GAYATRI VIDYA PARISHAD COLLEGE FOR DEGREE AND PG COURSES (AUTONOMOUS)
## ENGINEERING AND TECHNOLOGY PROGRAM
## RUSHIKONDA CAMPUS, VISAKHAPATNAM–530045.

| COURSE CODE | TITLE OF THE COURSE | CONTACT HOURS/ WEEK | | | ALLOTMENT OF MARKS | | CREDITS |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int. | Ext | |
| CSM3208 | Big Data Analytics Lab | 0 | 0 | 3 | 50 | 50 | 1.5 |

**COURSE OBJECTIVES:**
To enable students to have skills that will help them to solve complex real-world problems using Hadoop for decision support.

**COURSE OUTCOMES:**
At the end of the course student will be able to
CO1: Configure single, pseudo and fully distribution node Hadoop Cluster.
CO2: Apply Map Reduce algorithms for various real time problems.
CO3: Perform various Analytic operations in Hadoop.

**CYCLE 1. Getting Hadoop Up and Running in a Pseudo distribution cluster**

1. Setting up Hadoop on standalone machine.
2. Wordcount Map Reduce program using standalone Hadoop.
3. HDFS basic command-line file operations.
4. Setting Hadoop in a Pseudo-distributed environment.
5. Running the Wordcount program in a distributed cluster environment.
6. Adding the combiner step to the Wordcount Map Reduce program.
7. Hadoop Services monitoring using UI.

**CYCLE 2. Hadoop Map Reduce Applications**

8. Implementing Custom Hadoop Writable data type.
9. Implementing Generic Hadoop Writable data type.
10. Emitting data of different value types from a mapper.
11. Choosing a suitable Hadoop Input Format for your input data format.

**CYCLE 3. Analytics**

12. Performing Group-By using Map Reduce.
13. Calculating frequency distributions and sorting using Map Reduce.
14. Plotting the Hadoop results using GNU plot.

**TEXT BOOK:**
1. Hadoop Map Reduce Cookbook, Srinath Perera &Thilina Gunarathne, 2013, PACKT PUBLISHING.

# GAYATRI VIDYA PARISHAD COLLEGE FOR DEGREE AND PG COURSES (AUTONOMOUS)
# ENGINEERING AND TECHNOLOGY PROGRAM
# RUSHIKONDA CAMPUS, VISAKHAPATNAM–530045.

| COURSE CODE | TITLE OF THE COURSE | CONTACT HOURS/ WEEK | | | ALLOTMENT OF MARKS | | CREDITS |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int. | | |
| CSM3208 | Cryptography And Network Security Lab | 0 | 0 | 3 | 50 | 50 | 1.5 |

**COURSE OBJECTIVES:**

**1.** To implement the essential cipher techniques.
**2.** To develop various symmetric/asymmetric and key exchange algorithms.

**COURSE OUTCOMES:**

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

**1**: **Construct** new security applications by experimenting the cipher techniques.
**2**: **Apply** various cryptographic authentication techniques.
**3. Design** symmetric and asymmetric cryptographic protocols for secure data transmission.

**MODULE-I**

1. Write a C program to XOR the string “Hello World” with zero and print the result.
2. Write a C program to AND and XOR the string “Hello World” with 127 and print the result.
3. Write Java Program to Implement Hill cipher.
4. Java Program to Implement Caesar cipher.
5. Java Program to Implement Substitution cipher.
6. Java Program to implement play fair cipher.
7. Write a C/JAVA program to encrypt a text using Cipher Block Chaining using any of the above ciphers.
8. a) Implement the Euclid Algorithm to generate the GCD of 2 numbers.
   b) Write a program for primality testing.

**MODULE-II**

1. Java Program to implement the encryption and decryption of 8-bit data using Simplified DES Algorithm.
2. Implement RSA algorithm for encryption and decryption in Java.
3. Write a Java Program to generate the points on Elliptic curve cryptography for given parameters.
4. Implement Diffie Hellman Key exchange.
5. Calculate the message digest of SHA-1 in Java.
6. Implement Blowfish algorithm.
7. Implement Rijendal Algorithm.

**REFERENCE BOOKS:**

1. Cryptography and Network Security Principles and Practice, William Stallings, Seventh Edition, Pearson Education.
2. Computer Security - Principles and Practice, 4th Edition by William Stallings, Pearson Education.
3. Cryptography and Network Security, Atul Kahate, 4th Edition, Tata McGraw Hill Publications.
4. Cryptography and Network Security Behrouz A Frorouzan, Second Edition, Tata McGraw Hill Pub Company Ltd, New Delhi.

# GAYATRI VIDYA PARISHAD COLLEGE FOR DEGREE AND PG COURSES (AUTONOMOUS)
# ENGINEERING AND TECHNOLOGY PROGRAM
# RUSHIKONDA CAMPUS, VISAKHAPATNAM–530045.

| COURSE CODE | TITLE OF THE COURSE | CONTACT HOURS/WEEK | | | ALLOTMENT OF MARKS | | CREDITS |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int. | Ext | |
| CSM3209 | Android Programming(Skill Course - 4) | 1 | 0 | 2 | 50 | 50 | 2 |

## COURSE OBJECTIVES

- To learn how to develop Applications in android environment.
- To learn how to develop user interface applications.
- To learn how to develop URL related applications.

## COURSE OUTCOMES

CO1: Student understands the working of Android OS Practically.
CO2: Student will be able to develop user interfaces.
CO3: Student will be able to develop, deploy and maintain the Android Applications.

**UNIT 1: Get started:**

**Week 1**: Build your first app
1.1: Android Studio and Hello World,
1.2 Part A: Your first interactive UI
1.2 Part B: The layout editor
1.3: Text and scrolling views
1.4: Learn to help yourself

**Week 2**: Activities and intents:
2.1: Activities and intents
2.2: Activity lifecycle and state
2.3: Implicit intents

**Week 3**: Testing, debugging, and using support libraries
3.1 : The debugger
3.2 : Unit tests
3.3 : Support libraries

**UNIT 2: User experience**

**Week 4:** User interaction
4.1 : Clickable images
4.2 : Input controls
4.3 : Menus and pickers
4.4 : User navigation

4.5 : Recycler View

**Week 5:** Delightful user experience
5.1: Drawables, styles, and themes
5.2: Cards and colors
5.3: Adaptive layouts

**Week 6:** Testing your UI
6.1: Espresso for UI testing

**UNIT 3: Working in the background**

**Week 7:** Background tasks
7.1: Async Task
7.2: Async Task and Async Task Loader
7.3: Broadcast receivers

**Week 8:** Alarms and schedulers
8.1: Notifications
8.2: The alarm manager
8.3: Job Scheduler

**UNIT 4: Saving user data**

**Week 9:** Preferences and Settings
9.1: Shared preferences
9.2: App settings

**Week 10:** Storing data with Room
10.1 Part A: Room, LiveData, and ViewModel
10.1 Part B: Deleting data from a Room database

**WEB REFERENCES:**

1. https://developer.android.com/courses/fundamentals-training/overview-v2

2. https://developer.android.com/codelabs/android-training-welcome?hl=en#0

3. https://developers.google.com/learn?text=android%20developer%20fundame.

# GAYATRI VIDYA PARISHAD COLLEGE FOR DEGREE AND PG COURSES (AUTONOMOUS)
# ENGINEERING AND TECHNOLOGY PROGRAM
# RUSHIKONDA CAMPUS, VISAKHAPATNAM–530045.

| Course code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int. | Ext | |
| CSM3210 | Essence of Indian Traditional Knowledge | 3 | 0 | 0 | 30 | 70 | 0 |

**COURSE OBJECTIVES:**

1. To know the contributions of scientists for the development of society over a period of time.
2. To understand the Science and Technological developments that lead to human welfare.
3. To appreciate the Science and Technological contributions for the development of various sectors of the economy.
4. To identify the technological transfer versus economic progress of the countries.

**COURSE OUTCOMES:**

At the end of the course student will be able to

**CO1 -** Demonstrate knowledge of broad concepts in the history of science, technology ranging over time, space and cultures and appreciate the science and technological contributions for the development of various sectors of the economy.
**CO2 -** Recognize the values of a wide range of methodologies, conceptual approaches and policies for the development of science and technology.
**CO3 -** Think independently and critically, using appropriate methodologies and technological developments in the critical areas of science and technology that lead to human welfare.
**CO4 -** Proficiently use contemporary technologies.

## UNIT-I

**Historical Perspective of Science and Technology:** Nature and Definitions; Roots of Science – In Ancient Period and Modern Period (During the British Period); Science and Society; Role of Scientist in the Society.

## UNIT-II

**Policies and Plans after Independence:** Science and Technology Policy Resolutions New Technology Fund; Technology Development (TIFAC); Programs aimed at Technological Self Reliance; Activities of Council of Scientific and Industrial Research

**UNIT-III**

**Science and Technological Developments in Critical Areas Space** – The Indian Space Program: India‟s Geostationary Satellite Services – INSAT System and INSAT Services; Defense Research and Technology – Research Coordination, Research efforts and Development of technologies and Spin-off technologies for civilian use; Nuclear Energy –Effects of a nuclear explosion and India's safety measures.

**UNIT-IV**

**Impact of Science and Technology in Major Areas**

Ocean Development: Objectives of Ocean Development, Biological and Mineral resources, Marine Research and Capacity Building; Biotechnology: Meaning, Biotechnology techniques Bioreactors, Cell fusion, Cell or Tissue Culture, DNA Fingerprinting, Cloning, Artificial Insemination and Embryo Transfer Technology and Stem Cell Technology; Application of Biotechnology – Medicine, Biocatalysts, Food Biotechnology, Fuel and Fodder and Development of Biosensors.

**UNIT-V**

Technology Transfer and Development Transfer of Technology – Types, Methods, Mechanisms, Process, Channels and Techniques; Appropriate Technology - Criteria and Selection of an Appropriate Technology; Barriers of Technological Change.

**TEXT BOOKS:**

1. Kalpana Rajaram, Science and Technology in India, Published and Distributed by Spectrum Books (P) Ltd., New Delhi-58. 2. Srinivasan, M., Management of Science and Technology (Problems & Prospects), East – West Press (P) Ltd., New Delhi

# GAYATRI VIDYA PARISHAD COLLEGE FOR DEGREE AND PG COURSES (AUTONOMOUS)
# ENGINEERING AND TECHNOLOGY PROGRAM
# RUSHIKONDA CAMPUS, VISAKHAPATNAM–530045.

| Course code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int. | Ext | |
| Honors 3-2 | Software Defined Networks | 4 | 0 | 0 | 30 | 70 | 4 |

## COURSE OBJECTIVES:

1. To understand the fundamental principles of software-defined networking (SDN) and its architecture.
2. To explore the key protocols, technologies, and methodologies used in SDN.
3. To analyze the impact of SDN on network management and operations.
4. To study the application and implementation of SDN in various networking scenarios.

## COURSE OUTCOMES:

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

**CO1.** Demonstrate knowledge of SDN architecture and its components.
**CO2.** Evaluate the benefits and limitations of different SDN protocols and technologies.
**CO3.** Apply SDN concepts to design and manage network infrastructures.
**CO4.** Analyze case studies and real-world applications of SDN.
**CO5.** Develop and implement SDN solutions using appropriate tools and techniques.

## UNIT 1: Introduction to Software Defined Networking

Introduction to SDN, The need for SDN, Evolution of networking technology, Mainframe networking to routed networks, Overview of traditional networking vs. SDN, Fundamentals of SDN, Definition and key characteristics, Plane separation: Control plane and data plane, Centralized control and network automation, Openness in networking.

## UNIT 2: How SDN Works

SDN Operation and Devices, Flow tables and their role in SDN, SDN software switches and hardware devices, Existing SDN device implementations, Scaling the number of flows in SDN, SDN Controllers, Core modules and interfaces of SDN controllers, Implementations and potential issues with controllers, Controller-switch secure communication.

## UNIT 3: OpenFlow Protocol

Overview of OpenFlow, OpenFlow switch and controller, The OpenFlow protocol and secure channels, OpenFlow Versions and Features, OpenFlow 1.0 basics and packet forwarding, Enhancements in OpenFlow 1.1 to 1.4, Example use cases and implementations.

## UNIT 4: SDN Applications and Methods

SDN Applications, Responsibilities and examples of SDN applications, Network virtualization and automation, SDN in data centers and cloud environments, Alternate SDN Methods, SDN via APIs and hypervisor-based overlay networks, Benefits and limitations of different SDN methods.

## UNIT 5: Advanced Topics and Future Trends

Security and Performance, Security applications of SDN, Traffic engineering and quality of service (QoS), SDN in Emerging Technologies, SD-WAN and network functions virtualization (NFV), Future trends and novel applications of SDN, Case studies and real-world implementations.

## Textbooks:

1. Paul Goransson, Chuck Black, Timothy Culver, Software Defined Networks: A Comprehensive Approach, Morgan Kaufmann, 2016.

## Reference Books:

1. Thomas D. Nadeau, Ken Gray, SDN: Software Defined Networks, O'Reilly Media, 2013.
2. Jim Doherty, SDN and NFV Simplified: A Visual Guide to Understanding Software Defined Networks and Network Function Virtualization, Addison-Wesley Professional, 2016.
3. William Stallings, Foundations of Modern Networking: SDN, NFV, QoE, IoT, and Cloud, Addison-Wesley Professional, 2015.

# B. Tech Computer Science and Engineering (AI and ML)
## (R-22 Regulation)
### IV Year – I Semester

| Course code | Category | Course Title | Hours per week | | | Internal Marks | External Marks | Total Marks | Credits |
|---|---|---|---|---|---|---|---|---|---|
| | | | L | T | P | | | | |
| CSM4101 | PEC | Elective -III | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM4102 | PEC | Elective -IV | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM4103 | PEC | Elective -V | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM4104 | OEC/JOE | Open Elective-III | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM4105 | OEC/JOE | Open Elective-IV | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM4106 | HSS/HSMS | Professional Ethics and Universal Human Values (Understanding Harmony) | 3 | 0 | 0 | 30 | 70 | 100 | 3 |
| CSM4107 | SAC/SC | Skill Course – 5:<br>POWER BI | 1 | 0 | 2 | 50 | 50 | 100 | 2 |
| CSM4108 | Industrial/ Research Internship (2 months Mandatory after 6th Semester to be evaluated in 7th Semester) | | 0 | 0 | 0 | 0 | 100 | 100 | 3 |
| **Total Credits** | | | | | | | | | **23** |

| Title of the Program | L | T | P | Credits |
|---|---|---|---|---|
| **Honors/Minor Courses (The hours distribution can be 3-0-2 or 3-1-0 also)** | **3** | **1** | **0** | **4** |

**Elective-III**

**Natural Language Processing**
Cloud Computing

Reinforcement Learning

**Elective-IV**

Computer Vision
Wireless Sensor Networks
**Cyber Security & Digital Forensics**

**Elective-V**
**Deep Learning**
R-Programming
Robotics

**Open Elective - III**

Offered by ME

Offered by CE

Offered by ECE

**Open Elective - IV**

Offered by ME
Offered by CE
Offered by ECE

# GAYATRI VIDYA PARISHAD COLLEGE FOR DEGREE AND PG COURSES (AUTONOMOUS)
# ENGINEERING AND TECHNOLOGY PROGRAM
# RUSHIKONDA CAMPUS, VISAKHAPATNAM–530045.

| Course code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int. | Ext | |
| CSM4101 | Natural Language Processing | 3 | 0 | 0 | 30 | 70 | 3 |

**COURSE OBJECTIVES:**

1. Learn the techniques in natural language processing.
2. Perform natural language generation.
3. Apply the techniques of machine translation.
4. Understand Semantic Analysis and Syntactic Analysis
5. Understand the information retrieval techniques

**COURSE OUTCOMES:**

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

CO1: Analyze the natural language text and language modeling.

CO2: Understand Words, Word classes and Syntactic Analysis.

CO3: Understand Semantic Analysis, coherence and structure

CO4: Generate the Language and do machine translation

CO5: Apply information retrieval techniques on different models.

**UNIT-I**

**Overview:** Origins and challenges of NLP-Language and Grammar-Processing Indian Languages - NLP Applications-Information Retrieval.

**Language Modelling:** Various Grammar - based Language Models-Statistical Language Model.

**UNIT-II**

**Word level analysis:** Regular Expressions-Finite-State Automata-Morphological Parsing-Spelling Error Detection and correction-Words and Word classes-Part-of Speech Tagging.
**Syntactic Analysis:**

Context-free Grammar-Constituency- Parsing-Probabilistic Parsing.

**UNIT-III**

**Semantic analysis:** Semantic Analysis: Meaning Representation-Lexical Semantics-Ambiguity Word Sense Disambiguation.

**Discourse Processing:** cohesion-Reference Resolution- Discourse Coherence and Structure.

## UNIT-IV

**Natural language Generation:** Architecture of NLG Systems- Generation Tasks and Representations Application of NLG. Machine Translation: Problems in Machine Translation-Characteristics of Indian Languages- Machine Translation Approaches-Translation involving Indian Languages

## UNIT-V

**Information Retrieval:** Design features of Information Retrieval Systems-Classical, Nonclassical, and Alternative Models of Information Retrieval – valuation.

**Lexical Resources:** World Net Frame Net- Stemmers-POS Tagger- Research Corpora.

**Textbooks:**

1. Tanveer Siddiqui, U.S. Tiwary, "Natural Language Processing and Information Retrieval",Oxford

University Press, 2008.

**Reference Books:**

1. Daniel Jurafsky and James H Martin, "Speech and Language Processing: An introduction to Natural Language Processing, Computational Linguistics and Speech Recognition", 2nd Edition, Prentice Hall, 2008.

2. James Allen, "Natural Language Understanding", 2 nd Edition, Benjamin /Cummings publishing

company, 1995.

# GAYATRI VIDYA PARISHAD COLLEGE FOR DEGREE AND PG COURSES (AUTONOMOUS)
# ENGINEERING AND TECHNOLOGY PROGRAM
# RUSHIKONDA CAMPUS, VISAKHAPATNAM–530045.

| Course code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int. | Ext | |
| CSM4101 | Reinforcement Learning | 3 | 0 | 0 | 30 | 70 | 3 |

**Course Objectives:**

The objectives of this course are to:

1. Analyse and articulate the fundamental components and agent–environment dynamics in Reinforcement Learning.
2. Design and evaluate advanced exploration–exploitation strategies to optimize decision-making efficiency.
3. Apply and compare model-based and model-free prediction techniques in realistic problem domains.
4. Implement, benchmark, and refine diverse value-based Reinforcement Learning algorithms for complex tasks.
5. Develop and optimize policy-based Reinforcement Learning methods to achieve robust, optimal policies.

**Course Outcomes (COs)**

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

CO1: Demonstrate various Components of Reinforcement Learning.

CO2: Make use of various exploration and exploitation strategies.

CO3: Apply Model based and Model Free Prediction techniques.

CO4: Make use of different value-based Reinforcement Learning Algorithms.

CO5: Demonstrate various Policy based Reinforcement Learning Algorithms.

**Unit I**

**Introduction:** Deep Reinforcement Learning, Suitability of RL, Components of Reinforcement Learning -Agent, Environment, Observations, Actions, Example-The Bandit Walk Environment, Agent-Environment interaction cycle, MDP (Markov Decision Process): The engine of the Environment-States, Actions, Transition Function, Reward Signal

**Unit 2**

**Planning:** Objective of a decision-making agent-environment. Plan, Optimal policy. Comparison of Policies, Bellman Equation/State-Value Function, Action-Value Function, Action-Advantage Function, Optimality.

**Unit 3**

**Exploitation and Exploration of Reinforcement Learning:** Bandits- Single-state decision problem(Multi-Armed Bandit(MAB) problem), The cost of exploration, approaches to solve MAB environments, Greedy Strategy, Random Strategy, Epsilon-Greedy Strategy, Decaying Epsilon-Greedy Strategy, Optimistic Initialization strategy, Strategic exploration, SoftMax exploration strategy. Upper confidence bound (UCB) equation strategy, Thompson sampling strategy.

**Unit 4**

**Model Free Reinforcement Learning:** Monté Carlo Prediction (MC), First-Visit MC (FVMC), Every-Visit MC (EVMC), Temporal Difference Learning (TD), Learning to estimate from multiple steps, N-step TD learning, Forward-view TD(λ), Backward-view TD(λ), Generalized policy iteration (GPI), Monte Carlo control, SARSA: On-Policy TD control, Q-learning: Off-Policy TD control, Double Q-learning, SARSA(λ), Watkins's Q(λ).

**Model Based Reinforcement Learning:** Dyna-Q, Trajectory sampling

**Unit 5**

**Policy Based Reinforcement Learning:** Policy Gradient and Actor-Critic Methods REINFORCE Algorithm and Stochastic Policy Search, Vanilla Policy Gradient (VPG), Asynchronous Advantage Actor-Critic (A3C), Generalized Advantage Estimation (GAE), Advantage Actor-Critic (A2C), Deep Deterministic Policy Gradient (DDPG), Twin-Delayed DDPG (TD3), Soft Actor-Critic (SAC), Proximal Policy Optimization (PPO).

**TEXTBOOKS:**

1. Miguel Morales, *Grokking Deep Reinforcement Learning*, Manning Publications, 2020.

**REFERENCE BOOKS:**

1. Richard S. Sutton and Andrew G. Barto, *Reinforcement learning: An Introduction*, Second Edition, MIT Press, 2019.

2. Marco Wiering, Martijn van Otterlo (Ed.), *Reinforcement Learning, State-of-the-Art, Adaptation, Learning, and Optimization* book series, ALO, volume 12, Springer, 2012.
3. Keng, Wah Loon, Graesser, Laura, *Foundations of Deep Reinforcement Learning: Theory and Practice in Python*, Addison Wesley Data & Analytics Series, 2020.
4. Francois Chollet, *Deep Learning with Python*, Manning Publications, 2018.

# GAYATRI VIDYA PARISHAD COLLEGE FOR DEGREE AND PG COURSES (AUTONOMOUS)
# ENGINEERING AND TECHNOLOGY PROGRAM
# RUSHIKONDA CAMPUS, VISAKHAPATNAM–530045.

| Course code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int. | Ext | |
| CSM4102 | Cyber Security & Digital Forensics | 3 | 0 | 0 | 30 | 70 | 3 |

**Course Objectives:**

The objectives of this course are to:

1. To understand underlying principles and many of the techniques associated with the digital forensic practices and cybercrimes.
2. To explore practical knowledge about ethical hacking Methodology.
3. To develop an excellent understanding of current cyber security issues (Computer Security Incident) and analysed the ways that exploits in securities.
4. To apply digital forensic knowledge to use computer forensic tools and investigation report writing

**Course Outcomes (COs)**

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

CO1: Gain the knowledge on effective use of computer, data and internet securely.

CO2: Understand concepts and ethics of cyber security and cyber laws.

CO3: Acquire the knowledge on various web architectures, vulnerabilities, penetration testing, attacks and security of web applications

CO4: Illustrate the methods for Forensic Technologies, evidence collection, Evidentiary Reporting and information risk management

CO5: Analyse and respond to the cyber incidents.

## Unit I

**Introduction to Information Security Fundamentals and Best Practices:** Protecting Your Computer and its Contents, Securing Computer Networks-Basics of Networking, Compromised Computers, Secure Communications and Information Security Best Practices, Privacy Guidelines, Safe Internet Usage.

## Unit 2

**Ethics in Cyber Security & Cyber Law:** Privacy, Intellectual Property, Professional Ethics, Freedom of Speech, Fair User and Ethical Hacking, Trademarks, Internet Fraud, Electronic Evidence, Cybercrimes.

## Unit 3

**Penetration Testing:** Overview of the web from a penetration testers perspective, Exploring the various servers and clients, Discussion of the various web architectures, Discussion of the different types of vulnerabilities, Defining types of penetration testing.

**Web Application Security:** Common Issues in Web Apps, what is XSS, SQL injection, CSRF, Password Vulnerabilities, SSL, CAPTCHA, Session Hijacking, Local and Remote File Inclusion, Audit Trails, Web Server Issues.

## Unit 4

**Forensics & Network Assurance:** Forensic Technologies, Digital Evidence Collection, Evidentiary Reporting, Layered Défense, Surveillance and Reconnaissance, Outsider Thread **Protection. Information Risk Management:** Asset Evaluation and Business Impact Analysis, Risk Identification, Risk Quantification, Risk Response Development and Control, Security Policy, Compliance, and Business Continuity. Forensic investigation using Access Data FTK.

## Unit 5

**UNIT-V**

**Cyber Incident Analysis and Response:** Incident Preparation, Incident Detection and Analysis. Containment, Eradication, and Recovery. Proactive and Post-Incident Cyber Service, CIA Triangle.

**TEXTBOOKS:**

1.The Official CHFI Study Guide for Computer Hacking Forensic Investigator by Dave Kleiman.

2. CISSP Study Guide, 6th Edition by James M. Stewart.

3. Title: Cyber Forensics by Dejey & S.Murugan, OXFORD University Press.

**REFERENCE BOOKS:**

1. Introduction to Information Security and Cyber Laws (English, Paperback, Tripathi Surya Prakash)

| Course code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int. | Ext | |
| **CSM4103** | **Deep Learning (Elective- V)** | **3** | **0** | **0** | **30** | **70** | **3** |

## Course Objectives:

1. Learn deep learning fundamentals, regularization techniques, and optimization methods.
2. Comprehend the architecture and applications of convolutional neural networks.
3. Understand various sequence modeling techniques, including recurrent and recursive networks.
4. Design and implement real-time deep learning applications.
5. Understand and apply different types of autoencoders and deep generative models

## Course Outcomes:

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

CO1: Understand deep learning, regularization techniques, and optimization methods, enabling them to design and train effective deep learning models.

CO2: Understand the convolutional neural networks.

CO3: Describe various sequence modeling techniques.

CO4: Design various real time deep learning applications.

CO5: Understand various types of Auto encoders and different techniques of Deep Generative Models.

**UNIT-I**

**Introduction to Deep Learning:** Introduction - Curse of Dimensionality, Historical trends in Deep Learning, Deep feed - Networks, Gradient Learning, Hidden Units, Architecture design, back propagation, activation functions.

**Regularization and Optimization:** Regularization techniques- L1, L2, dataset augmentation, early stopping, dropout, surrogate loss functions, batch and mini batch algorithms. batch normalization, Comparison of shallow learning and deep learning, Importance and Applications of Deep Learning.

## UNIT-II

**Convolutional Networks:** The Convolution Operation, Pooling, Convolution, Basic Convolution Functions, Structured Outputs, Data Types, Efficient Convolution Algorithms, Random or Unsupervised Features, The Neuroscientific Basis for Convolutional Networks, Dataset Augmentation, Image Captioning.

## UNIT- III

**Sequence Modeling- Recurrent and Recursive Nets:** Unfolding Computational Graphs, Recurrent Neural Networks, Bidirectional RNNs, Encoder-Decoder Sequence-to-Sequence Architectures, Deep Recurrent Networks, Recursive Neural Networks, Leaky Units, LSTM.

## UNIT-IV

**Practical Methodology:** Performance Metrics, Default Baseline Models, Determining Whether to Gather More Data, Selecting Hyper parameters, Debugging Strategies, Case study: Multi-Digit Number Recognition.

**Applications:** Large-Scale Deep Learning, Computer Vision, Speech Recognition, Natural Language Processing, Other Applications.

## UNIT-V

**Autoencoders:** Undercomplete Autoencoders, Regularized Autoencoders, Representational Power, Layer Size and Depth Stochastic Encoders and Decoders, Denoising Autoencoders, Applications of Autoencoders.

**Deep Generative Models:** Directed Generative Nets - Variable Autoencoders, Generative adversarial networks, Representation Learning.

**Textbooks:**

1. Ian Goodfellow, Yoshua Bengio, Aaron Courville, “Deep Learning”, MIT Press,2016.

**Reference Books:**

1. Josh Patterson and Adam Gibson, “Deep learning: A practitioner's approach”,

O'Reilly Media, First Edition, 2017.

2. Fundamentals of Deep Learning, Designing next-generation machine intelligence algorithms, Nikhil Buduma, O'Reilly, Shroff Publishers, 2019.
3. Deep learning CookBook, Practical recipes to get started Quickly, Douwe Osinga, O'Reilly, Shroff Publishers, 2019.

# GAYATRI VIDYA PARISHAD COLLEGE FOR DEGREE AND PG COURSES (AUTONOMOUS)
# ENGINEERING AND TECHNOLOGY PROGRAM
# RUSHIKONDA CAMPUS, VISAKHAPATNAM–530045.

| Course code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int. | Ext | |
| CSM4104 | R-Programming | 3 | 0 | 0 | 30 | 70 | 3 |

**Course Objectives:**

The objectives of this course are to:

1. Introduce the R environment, language fundamentals, and philosophy of statistical computing.
2. Develop proficiency in data structures, data manipulation, and exploratory data analysis using R.
3. Teach effective programming practices including control structures, functions, debugging, and performance profiling.
4. Enable students to perform data import/export, data cleaning, transformation, and visualization in R.
5. Apply R programming to real-world data analysis projects, including simulation and case studies

**Course Outcomes (COs)**

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

CO1: Explain the history, installation, and core components of the R ecosystem in data science.

CO2: Perform efficient data input/output, manipulation, cleaning, and transformation using base R and tidyverse.

CO3: Develop robust R scripts with control structures, functions, and best programming practices.

CO4: Create publication-quality visualizations, debug and profile code, and manage memory for scalable analyses.

CO5: Conduct simulations and end-to-end case study analyses with reproducible reporting using R Markdown and Git.

.

## Unit I

**Introduction and R Foundations:** History and overview of R: origins, philosophy, and ecosystem, Installation and setup: R and RStudio environments; basic interface (console, script editor, working directory).R objects and atomic classes: numeric, integer, complex, logical, character. Core data structures: vectors, matrices, arrays, lists, factors, and data frames

## Unit 2

**Data Input/Output and Manipulation:** Reading and writing data: read.table(), read.csv(), readr package; efficient import of large datasets. Data serialization: save(), load(), dput(), dump(); binary formats and memory considerations. Interfaces: file connections, reading from URLs and external sources. Managing missing values and data cleaning strategies. Data manipulation with base R: subsetting ([], [[ ]], $), vectorized operations, matrix arithmetic.Introduction to tidyverse: dplyr verbs (select, filter, arrange, mutate, rename, group by, summarize) and tidyr (pivot_longer, pivot_wider). Handling dates and times: Date, POSIXct, and lubridate basics.

## Unit 3

**Programming Constructs and Functional Programming:** Control structures: if-else, for loops, while loops, repeat, break, next. Writing functions: arguments, defaults, lazy evaluation, operator; writing reusable code. Scoping rules and environments: lexical scoping, closures; best practices for modular code. Loop functions and apply family: lapply, sapply, vapply, tapply, mapply, purrr map functions. Vectorizing user-defined functions for performance. Coding standards and best practices: style guidelines, version control with Git.

## Unit 4

**Visualization, Debugging, and Performance:** Base graphics: plots, histograms, boxplots, customizing visuals. ggplot2 fundamentals: grammar of graphics, layering, aesthetics, themes, and extensions. Exploratory Data Analysis workflows combining visualization and summary statistics. Debugging tools: traceback(), debug(), recover(), browser(), error handling techniques. Profiling R code: system.time(), Rprof(), summaryRprof(); identifying and addressing performance bottlenecks. Memory management: inspecting object sizes, garbage collection, efficient data handling for large datasets.

## Unit 5

**Simulation, Case Studies, and Reproducible Reporting:** Random number generation: set.seed(), generating samples; simulation of statistical models and Monte Carlo experiments. Case studies: domain-specific data analysis (e.g., environmental data, finance, healthcare); applying end-to-end workflows. Reproducible research: R Markdown for integrating code, outputs, and narrative; version control of reports. Project workflow: data acquisition, cleaning, analysis, visualization, interpretation, and presentation. Final project: students undertake an end-to-end analysis in R, deliver a reproducible report, and present findings.

**TEXTBOOKS:**

1. Roger D. Peng, *R Programming for Data Science*, Leanpub, 2015.
2. Garrett Grolemund & Hadley Wickham, *R for Data Science*, O'Reilly Media, 2017.

**REFERENCE BOOKS:**

1. Hadley Wickham, *Advanced R*, Chapman & Hall/CRC, 2014.
2. Paul Murrell, *R Graphics*, Chapman & Hall/CRC, 2005.
3. William N. Venables & Brian D. Ripley, *Modern Applied Statistics with S*, Springer, 2002.
4. Norman Matloff, *The Art of R Programming*, No Starch Press, 2011.
5. Yihui Xie, *R Markdown: The Definitive Guide*, CRC Press, 2020.

# GAYATRI VIDYA PARISHAD COLLEGE FOR DEGREE AND PG COURSES (AUTONOMOUS)
# ENGINEERING AND TECHNOLOGY PROGRAM
# RUSHIKONDA CAMPUS, VISAKHAPATNAM–530045.

| Course code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int. | Ext | |
| CSM4105 | Robotics | 3 | 0 | 0 | 30 | 70 | 3 |

**Course Objectives:**

The objectives of this course are to:

1. Understand and discuss the fundamental elementary concepts of Robotics.
2. Provide insight into different types of robots.
3. Explain intelligent module for robotic motion control.
4. Educate on various path planning techniques.
5. Illustrate the working of innovative robotic devices

**Course Outcomes (COs)**

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

CO1: Understand the significance, social impact and prospects of robotics and automation in various engineering applications.

CO2: Identify and describe the components and anatomy of robotic system.

CO3: Know about various path planning techniques and analyse different motions of robotics system CO4: Use the suitable drives and end-effectors for a given robotics application.

CO5: Apply robotics concept to automate the monotonous and hazardous tasks and categorize various types of robots based on the design and applications in real world scenarios.

## Unit I

**Introduction To Robotics:** Introduction to Robotics and Automation, laws of robot, brief history of robotics, basic components of robot, robot specifications, classification of robots, human system and robotics, safety measures in robotics, social impact, Robotics market and the prospects, advantages and disadvantages of robots.

## Unit 2

**Robot Anatomy and Motion Analysis:** Anatomy of a Robot, Robot configurations: polar, cylindrical, Cartesian, and jointed arm configurations, Robot links and joints, Degrees of freedom: types of movements, vertical, radial and rotational traverse, roll, pitch and yaw, Wok

volume/envelope, Robot kinematics: Introduction to direct and inverse kinematics, transformations and rotation matrix.

**Unit 3**

**Robot Drives And End Effectors:** Robot drive systems: Hydraulic, Pneumatic and Electric drive systems, classification of end effectors, mechanical grippers, vacuum grippers, magnetic grippers, adhesive gripper, gripper force analysis and gripper design, 1 DoF, 2 DoF, multiple degrees of freedom robot hand, tools as end effectors, Robot control types: limited sequence control, point-to-point control, playback with continuous path control, and intelligent control.

**Unit 4**

**Path Planning:** Definition-Joint space technique, Use of P-degree polynomial-Cubic, polynomial- Cartesian space technique, parametric descriptions, straight line and circular paths, position and orientation planning.

**Unit 5**

**Robotics Applications: Material Handling:** pick and place, palletizing and depalletizing, machining loading and unloading, welding & assembly, Medical, agricultural and space applications, unmanned vehicles: ground, Ariel and underwater applications, robotic for computer integrated manufacturing. Types of robots: Manipulator, Legged robot, wheeled robot, aerial robots, Industrial robots, Humanoids, Robots, Autonomous robots, and Swarm robots

**Textbooks**:

1. S.R. Deb, Robotics Technology and flexible automation, Tata McGraw-Hill Education, 2009.

2. Mikell P. Groover et. al., "Industrial Robots - Technology, Programming and Applications", McGraw Hill, Special Edition, (2012).

3. Ganesh S Hegde, “A textbook on Industrial Robotics”, University science press, 3rdedition, 2017.

**Reference Books**:

1. Richard D Klafter, Thomas A Chmielewski, Michael Negin, "Robotics Engineering – An Integrated Approach", Eastern Economy Edition, Prentice Hall of India Pvt. Ltd., 2006.

2. Fu K S, Gonzalez R C, Lee C.S.G, "Robotics: Control, Sensing, Vision and Intelligence",McGraw Hill, 1987. https://www.robots.com/applications.

# GAYATRI VIDYA PARISHAD COLLEGE FOR DEGREE AND PG COURSES (AUTONOMOUS)
# ENGINEERING AND TECHNOLOGY PROGRAM
# RUSHIKONDA CAMPUS, VISAKHAPATNAM–530045.

| Course code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int. | Ext | |
| **CSM4107** | **POWER BI** | **1** | **0** | **2** | **50** | **50** | **2** |

**Course Objectives:**

1. Learn fundamental concepts of Business Intelligence and Power BI
2. Develop skills in data connectivity, transformation, and visualization
3. Master advanced analytics, DAX functions, and enterprise-level reporting

**Course Outcomes:**

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

C01: Connect to various data sources and perform data transformation using Power Query

C02: Create interactive reports, dashboards, and implement security measures

C03: Develop advanced analytics solutions using DAX and optimize Power BI performance

**Week 1: Introduction & Ecosystem**

**Focus**: Getting started with Power BI.

**Theory Topics**:

- Benefits of Power BI in BI workflows.
- Power BI ecosystem components: Desktop, Service (cloud), Mobile.
- Licensing options (Free, Pro, Premium Per User, Premium capacity): comparison, use cases.
- Installation and initial setup of Power BI Desktop.

**Practical Exercises / Labs**:

1. Install Power BI Desktop; verify version and updates.
2. Explore interface: panes (Fields, Visualizations), ribbons, Report/Data/Model views.
3. Create first simple report: load sample data, build a bar chart and a slicer.
4. Organize files/workspaces locally: save PBIX, folder structure for projects.

**Week 2: Data Connectivity & Import Methods**

**Focus**: Connecting to data sources and understanding modes.

**Theory Topics**:

- Import vs Direct Query: definitions, pros/cons, performance and freshness trade-offs.
- Direct Query specifics: supported sources, limitations on transformations/DAX.
- Import mode specifics: data compression, refresh scheduling.
- Connectivity best practices: filtering early, minimizing data transferred.

**Practical Exercises / Labs**:

1. Connect to a Google Sheet via Web connector; inspect data privacy settings.
2. Connect to Azure Blob Storage: load a CSV from blob.
3. Connect to Snowflake or simulate with another relational database, review credentials and privacy levels.
4. Change data source for an existing query (e.g., switch from local CSV to cloud file).
5. Practice imports from various file formats: Excel, CSV, JSON; observe data type detection.

**Week 3: Power Query & Data Transformation**

**Focus**: Cleaning and shaping data in Power Query Editor.

**Theory Topics**:

- Power Query Editor overview: Applied Steps pane, formula bar, Query Settings.
- Data cleaning principles: handling missing or inconsistent data.
- Column quality, distribution, and profile concepts for data assessment.

**Practical Exercises / Labs**:

1. Add an Index column: discuss scenarios (e.g., ordering, merges).

2. Fill Down/Up missing values in hierarchical data.
3. Merge/join tables on multiple columns; inspect join types (Left, Inner, etc.) and row counts.
4. Convert data types; handle errors using "Replace Errors" or custom conditional columns.

**Week 4: Basic Visualizations & Design Principles**

**Focus**: Core visuals and design best practices.

**Theory Topics**:

- Chart types and their use cases: bar, line, pie (sparingly), table/matrix for details.
- Visual design principles: clarity, minimalism, appropriate labels, avoiding clutter.
- On-object formatting in Power BI Desktop: quick formatting techniques.

**Practical Exercises / Labs**:

1. Build basic visuals: Bar, Line, Pie, and Table from sample data.
2. Format visuals on-object: adjust titles, axes, data labels, and color palettes.
3. Enable total labels on stacked visuals.
4. Display values on rows for matrix visuals; adjust subtotals.

**Week 5: Data Modelling Fundamentals**

**Focus**: Designing a robust data model.

**Theory Topics**:

- Star vs snowflake schemas; pros/cons and best practices.
- Relationships and cardinality (one-to-many, many-to-many considerations).
- Importance of a Date/Calendar table for time intelligence.
- Hierarchies and grouping concepts for user navigation.

**Practical Exercises / Labs**:

1. Create and manage relationships in Model view; set correct cardinalities.
2. Build hierarchies (e.g., Year > Quarter > Month) for date fields.
3. Create Groups (buckets) for numeric or categorical fields.
4. Create a dedicated Measure table; move measures there for clarity.
5. Use Display Folders to group related measures logically.

**Week 6: Advanced Visualizations & Custom Visuals**

**Focus**: Enhancing reports with KPIs and custom visuals.

**Theory Topics**:

- Native vs custom visuals: advantages, potential performance impact, governance.
- KPI concepts and effective design.
- Basics of multi-dimensional analysis in visuals.

**Practical Exercises / Labs**:

1. Create KPI visuals: define target vs actual values; add trend indicators.
2. Build small multiple line charts to compare series across categories.
3. Install & configure a “Multi KPI” custom visual from the Marketplace.
4. Create a Table Heatmap visual: apply color scales within tables to highlight high/low values.
5. Use SVG-based visuals/icons for dynamic KPI displays.

**Week 7: Interactive Features & UX**

**Focus**: Report interactivity and user experience design.

**Theory Topics**:

- Interactivity concepts: slicers, drill-through, tooltips.
- UX principles: intuitive navigation, storytelling with data.
- Bookmarking and buttons for guided navigation.

**Practical Exercises / Labs**:

1. Sync slicers across multiple pages for consistent filtering.
2. Create custom report page tooltips to show extra details on hover.
3. Implement Page Navigation using Bookmarks & Buttons to simulate a dashboard flow.
4. Configure Drill-through to detail pages (e.g., click a region to see regional breakdown).
5. Add "Clear All Slicers" functionality via bookmark and DAX measure trick.
6. Use "Apply all filters" features/bookmarks to maintain consistent states across pages.

**Week 8: Dynamic Content & Conditional Formatting**

**Focus**: Making reports responsive to user selections.

**Theory Topics**:

- Dynamic content concepts: using DAX measures for titles and labels.
- Conditional formatting principles: highlighting insights, anomalies.

**Practical Exercises / Labs**:

1. Create Dynamic Title measures that reflect slicer selections.
2. Handle multi-select scenarios for dynamic titles (e.g., show "Multiple Regions Selected" when more than one).
3. Apply conditional formatting on visuals by numeric field values (e.g., color bars red/green based on thresholds).
4. Conditional formatting based on string fields (e.g., flag status "Delayed" in red).
5. Dynamically change measures (e.g., Top N items) based on slicer input.
6. Change column headers dynamically using field parameters and measures.
7. Apply conditional formatting for data labels (e.g., highlight labels above a target).

**Week 9: DAX Fundamentals**

**Focus**: Core DAX concepts and practices.

**Theory Topics**:

- Introduction to DAX language: syntax and patterns.
- Row context vs filter context; evaluation context basics.
- Readable DAX: indentation, comments, naming conventions.
- Calculated columns vs measures: storage, performance considerations.

**Practical Exercises / Labs**:

1. Write basic DAX measures (SUM, AVERAGE, ratio calculations).
2. Create calculated columns vs measures; discuss when to use each and performance impact.
3. Implement time intelligence functions (TOTALYTD, SAMEPERIODLASTYEAR) using a proper Date table.
4. Use Quick Measures to auto-generate DAX; inspect and learn from underlying formulas.
5. Solve sample scenarios: Year-over-Year growth, moving averages, percentage of total, ranking.

**Week 10: Security & Access Control**

**Focus**: Securing data at row level and managing permissions.

**Theory Topics**:

- Power BI security concepts overview.
- Row-Level Security (RLS): static vs dynamic approaches.
- Using USERPRINCIPALNAME() in dynamic RLS.
- Workspace roles and permissions in Power BI Service (Admin, Member, Contributor, Viewer).

**Practical Exercises / Labs**:

1. Define static RLS roles in Desktop; test with "View as Role" functionality.
2. Implement dynamic RLS based on user login via a mapping table and USERPRINCIPALNAME().
3. Model organizational hierarchy for RLS (e.g., manager-subordinate data access).
4. Test various RLS scenarios; troubleshoot common issues.
5. Explore and assign Power BI Service workspace roles; understand implications for editing and sharing.

**Week 11: Power BI Service & Collaboration**

**Focus**: Deploying and collaborating in Power BI Service.

**Theory Topics**:

- Power BI Service overview: workspaces, apps, dashboards vs reports.
- Data Gateway concepts: personal vs standard; use cases.
- Sharing and collaboration strategies: publishing, apps, permissions.
- Subscriptions, data alerts, and monitoring.

**Practical Exercises / Labs**:

1. Publish Power BI Desktop report to Power BI Service; configure dataset settings and refresh.
2. Create dashboards by pinning visuals; differentiate dashboards from report pages.
3. Install and configure On-Premises Data Gateway in Personal and Standard modes.
4. Set up scheduled refresh with proper credential management.
5. Subscribe to reports/dashboards; configure email notifications for stakeholders.

6. Configure data alerts on KPIs (e.g., notify when metric crosses threshold).
7. Create and manage Power BI Apps: package multiple reports/dashboards, manage access.

**Week 12: Performance Optimization & Enterprise Features**

**Focus**: Tuning performance and enterprise deployment patterns.

**Theory Topics**:

- Performance optimization principles: model size, DAX efficiency, visual complexity.
- Using Performance Analyzer Tool in Desktop.
- Advanced analytics concepts: aggregations, incremental refresh, composite models.
- Enterprise deployment strategies: dataflows, embedding reports, governance best practices.
- Custom branding and theming in Service.

**Practical Exercises / Labs**:

1. Enable Performance Analyzer; capture query durations vs rendering times; interpret findings.
2. Set visual query limits; handle scenarios with large datasets (e.g., summarization).
3. Optimize data model: remove unused columns, create aggregations, configure incremental refresh.
4. Apply DAX optimization techniques: use variables, avoid unnecessary context transitions, simplify expressions.
5. Create a Power BI Dataflow in Service for reusable ETL; demonstrate linking dataflows to datasets.
6. Embed a report in Microsoft Teams or a simple web portal; verify access/security.
7. Apply custom branding/theme in Service (custom color palette, logo).

8. Document the optimization steps and results in a brief case study (not in table).

- **Quizzes/Concept Checks**: Short quizzes or reflection questions after key topics (e.g., "Explain Import vs DirectQuery trade-offs," "Describe row vs filter context in DAX").
- **Lab Submissions**: Evaluate based on correctness of transformations, model design, visual clarity, and application of best practices.
- **Mini-Project Presentations**: Peer review focusing on UX, performance, and interactivity.
- **Capstone Evaluation**: Assess completeness (end-to-end flow), technical soundness (correct modeling, DAX, RLS), performance optimizations, documentation quality, and demonstration clarity.

**Resources & References**

**Textbooks**:

1. *The Definitive Guide to DAX* (2nd Ed) by Marco Russo & Alberto Ferrari.
2. *Mastering Microsoft Power BI* by Brett Powell.
3. *Power BI Cookbook* by Brett Powell.

# GAYATRI VIDYA PARISHAD COLLEGE FOR DEGREE AND PG COURSES (AUTONOMOUS)
# ENGINEERING AND TECHNOLOGY PROGRAM
# RUSHIKONDA CAMPUS, VISAKHAPATNAM–530045.

| Course code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int. | Ext | |
| CSM4-1 Honors | Multiagent Systems | 4 | 0 | 0 | 30 | 70 | 4 |

**Course Objectives:**

The objectives of this course are to:

1) Key objectives include understanding agent concepts, decision-making frameworks, communication and coordination mechanisms.
2) To design, analyse, and implement systems involving multiple interacting agents.
3) To make students to understand the application of game theory and other computational techniques to model and solve problems in distributed environments.

**Course Outcomes (COs)**

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

1) Understanding of what agents are, and some of the issues associated with building them.
2) Apply the knowledge of how the agents use logical rules to derive specific conclusions to problem-solving in various contexts.
3) Gain the ability to classify different types of multi-agent systems.
4) Acquire advanced knowledge of formal logics used for reasoning about multi-agent systems.
5) Apply the knowledge of Multiagent Systems to various fields and domain areas.

**Unit I**

**Intelligent Agents :** Environments , Intelligent Agents, Agents and Objects , Agents and Expert Systems, Agents as Intentional Systems, Abstract Architectures for Intelligent Agents, How to Tell an Agent What to Do, Synthesizing Agents.

**Unit 2**

**Deductive Reasoning Agents:** Agents as Theorem Provers, Agent-Oriented Programming, Practical Reasoning Agents, Practical Reasoning Equals Deliberation Plus Means, Ends

Reasoning, Means-Ends Reasoning, Implementing a Practical Reasoning Agent, HOMER: an Agent That Plans.

**Unit 3**

**Multiagent Interactions:** Utilities and Preferences, Multiagent Encounters, Dominant Strategies and Nash Equilibria, Competitive and Zero-Sum Interactions, The Prisoner's Dilemma, Other Symmetric 2x 2 Interactions, Dependence Relations in Multiagent Systems.

**Unit 4**

**Logics for Multiagent Systems:** Why Modal Logic, Possible-Worlds Semantics for Modal Logics, Normal Modal Logics, Epistemic Logic for Multiagent Systems, Pro-attitudes: Goals and Desires, Common and Distributed knowledge, Integrated Theories of Agency

**Unit 5**

**Applications**: Agents for Workflow and Business Process Management, Agents for Distributed Sensing, Agents for Information Retrieval and Management, Agents for Electronic Commerce, Agents for Human-Computer Interfaces, Agents for Virtual Environments, Agents for Social Simulation .

**TEXTBOOKS:**

1. An Introduction to MultiAgent Systems by MICHAEL WOOLDRIDGE, JOHN WILEY & SONS, LTD, 2nd Edition.

**REFERENCE BOOKS:**

1. Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations by KeviLeyton-Brown , Yoav Shoham,
2. Cambridge University Press Multiagent Systems by Gerhard Weiss , 2nd Edition.

# B. Tech Computer Science and Engineering (AI and ML)
## (R-22 Regulation)
### IV Year – II Semester

| Course code | Category | Course Title | Hours per week | | | Internal Marks | External Marks | Total Marks | Credits |
|---|---|---|---|---|---|---|---|---|---|
| | | | L | T | P | | | | |
| Proj | Major Project | Project Work, Seminar & Internship in Industry | 0 | 0 | 0 | 50 | 50 | 100 | 12 |
| | | | | | | | | **Total Credits** | **12** |

# GAYATRI VIDYA PARISHAD COLLEGE FOR DEGREE AND PG COURSES (AUTONOMOUS)
# ENGINEERING AND TECHNOLOGY PROGRAM
# RUSHIKONDA CAMPUS, VISAKHAPATNAM–530045.

| Course code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int. | Ext | |
| CSM4101 | Natural Language Processing | 3 | 0 | 0 | 30 | 70 | 3 |

**COURSE OBJECTIVES:**

1. Learn the techniques in natural language processing.
2. Perform natural language generation.
3. Apply the techniques of machine translation.
4. Understand Semantic Analysis and Syntactic Analysis
5. Understand the information retrieval techniques

**COURSE OUTCOMES:**

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

CO1: Analyze the natural language text and language modeling.

CO2: Understand Words, Word classes and Syntactic Analysis.

CO3: Understand Semantic Analysis, coherence and structure

CO4: Generate the Language and do machine translation

CO5: Apply information retrieval techniques on different models.

**UNIT-I**

**Overview:** Origins and challenges of NLP-Language and Grammar-Processing Indian Languages - NLP Applications-Information Retrieval.

**Language Modelling:** Various Grammar - based Language Models-Statistical Language Model.

**UNIT-II**

**Word level analysis:** Regular Expressions-Finite-State Automata-Morphological Parsing-Spelling Error Detection and correction-Words and Word classes-Part-of Speech Tagging.
**Syntactic Analysis:**

Context-free Grammar-Constituency- Parsing-Probabilistic Parsing.

**UNIT-III**

**Semantic analysis:** Semantic Analysis: Meaning Representation-Lexical Semantics-Ambiguity Word Sense Disambiguation.

**Discourse Processing:** cohesion-Reference Resolution- Discourse Coherence and Structure.

## UNIT-IV

**Natural language Generation:** Architecture of NLG Systems- Generation Tasks and Representations Application of NLG. Machine Translation: Problems in Machine Translation-Characteristics of Indian Languages- Machine Translation Approaches-Translation involving Indian Languages

## UNIT-V

**Information Retrieval:** Design features of Information Retrieval Systems-Classical, Nonclassical, and Alternative Models of Information Retrieval – valuation.

**Lexical Resources:** World Net Frame Net- Stemmers-POS Tagger- Research Corpora.

**Textbooks:**

1. Tanveer Siddiqui, U.S. Tiwary, "Natural Language Processing and Information Retrieval",Oxford

University Press, 2008.

**Reference Books:**

1. Daniel Jurafsky and James H Martin, "Speech and Language Processing: An introduction to Natural Language Processing, Computational Linguistics and Speech Recognition", 2nd Edition, Prentice Hall, 2008.

2. James Allen, "Natural Language Understanding", 2 nd Edition, Benjamin /Cummings publishing

company, 1995.

# GAYATRI VIDYA PARISHAD COLLEGE FOR DEGREE AND PG COURSES (AUTONOMOUS)
# ENGINEERING AND TECHNOLOGY PROGRAM
# RUSHIKONDA CAMPUS, VISAKHAPATNAM–530045.

| Course code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int. | Ext | |
| CSM4101 | Reinforcement Learning | 3 | 0 | 0 | 30 | 70 | 3 |

**Course Objectives:**

The objectives of this course are to:

1. Analyse and articulate the fundamental components and agent–environment dynamics in Reinforcement Learning.
2. Design and evaluate advanced exploration–exploitation strategies to optimize decision-making efficiency.
3. Apply and compare model-based and model-free prediction techniques in realistic problem domains.
4. Implement, benchmark, and refine diverse value-based Reinforcement Learning algorithms for complex tasks.
5. Develop and optimize policy-based Reinforcement Learning methods to achieve robust, optimal policies.

**Course Outcomes (COs)**

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

CO1: Demonstrate various Components of Reinforcement Learning.

CO2: Make use of various exploration and exploitation strategies.

CO3: Apply Model based and Model Free Prediction techniques.

CO4: Make use of different value-based Reinforcement Learning Algorithms.

CO5: Demonstrate various Policy based Reinforcement Learning Algorithms.

**Unit I**

**Introduction:** Deep Reinforcement Learning, Suitability of RL, Components of Reinforcement Learning -Agent, Environment, Observations, Actions, Example-The Bandit Walk Environment, Agent-Environment interaction cycle, MDP (Markov Decision Process): The engine of the Environment-States, Actions, Transition Function, Reward Signal

**Unit 2**

**Planning:** Objective of a decision-making agent-environment. Plan, Optimal policy. Comparison of Policies, Bellman Equation/State-Value Function, Action-Value Function, Action-Advantage Function, Optimality.

**Unit 3**

**Exploitation and Exploration of Reinforcement Learning:** Bandits- Single-state decision problem(Multi-Armed Bandit(MAB) problem), The cost of exploration, approaches to solve MAB environments, Greedy Strategy, Random Strategy, Epsilon-Greedy Strategy, Decaying Epsilon-Greedy Strategy, Optimistic Initialization strategy, Strategic exploration, SoftMax exploration strategy. Upper confidence bound (UCB) equation strategy, Thompson sampling strategy.

**Unit 4**

**Model Free Reinforcement Learning:** Monté Carlo Prediction (MC), First-Visit MC (FVMC), Every-Visit MC (EVMC), Temporal Difference Learning (TD), Learning to estimate from multiple steps, N-step TD learning, Forward-view TD(λ), Backward-view TD(λ), Generalized policy iteration (GPI), Monte Carlo control, SARSA: On-Policy TD control, Q-learning: Off-Policy TD control, Double Q-learning, SARSA(λ), Watkins's Q(λ).

**Model Based Reinforcement Learning:** Dyna-Q, Trajectory sampling

**Unit 5**

**Policy Based Reinforcement Learning:** Policy Gradient and Actor-Critic Methods REINFORCE Algorithm and Stochastic Policy Search, Vanilla Policy Gradient (VPG), Asynchronous Advantage Actor-Critic (A3C), Generalized Advantage Estimation (GAE), Advantage Actor-Critic (A2C), Deep Deterministic Policy Gradient (DDPG), Twin-Delayed DDPG (TD3), Soft Actor-Critic (SAC), Proximal Policy Optimization (PPO).

**TEXTBOOKS:**

1. Miguel Morales, *Grokking Deep Reinforcement Learning*, Manning Publications, 2020.

**REFERENCE BOOKS:**

1. Richard S. Sutton and Andrew G. Barto, *Reinforcement learning: An Introduction*, Second Edition, MIT Press, 2019.

2. Marco Wiering, Martijn van Otterlo (Ed.), *Reinforcement Learning, State-of-the-Art, Adaptation, Learning, and Optimization* book series, ALO, volume 12, Springer, 2012.
3. Keng, Wah Loon, Graesser, Laura, *Foundations of Deep Reinforcement Learning: Theory and Practice in Python*, Addison Wesley Data & Analytics Series, 2020.
4. Francois Chollet, *Deep Learning with Python*, Manning Publications, 2018.

# GAYATRI VIDYA PARISHAD COLLEGE FOR DEGREE AND PG COURSES (AUTONOMOUS)
# ENGINEERING AND TECHNOLOGY PROGRAM
# RUSHIKONDA CAMPUS, VISAKHAPATNAM–530045.

| Course code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int. | Ext | |
| CSM4102 | Cyber Security & Digital Forensics | 3 | 0 | 0 | 30 | 70 | 3 |

**Course Objectives:**

The objectives of this course are to:

1. To understand underlying principles and many of the techniques associated with the digital forensic practices and cybercrimes.
2. To explore practical knowledge about ethical hacking Methodology.
3. To develop an excellent understanding of current cyber security issues (Computer Security Incident) and analysed the ways that exploits in securities.
4. To apply digital forensic knowledge to use computer forensic tools and investigation report writing

**Course Outcomes (COs)**

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

CO1: Gain the knowledge on effective use of computer, data and internet securely.

CO2: Understand concepts and ethics of cyber security and cyber laws.

CO3: Acquire the knowledge on various web architectures, vulnerabilities, penetration testing, attacks and security of web applications

CO4: Illustrate the methods for Forensic Technologies, evidence collection, Evidentiary Reporting and information risk management

CO5: Analyse and respond to the cyber incidents.

## Unit I

**Introduction to Information Security Fundamentals and Best Practices:** Protecting Your Computer and its Contents, Securing Computer Networks-Basics of Networking, Compromised Computers, Secure Communications and Information Security Best Practices, Privacy Guidelines, Safe Internet Usage.

## Unit 2

**Ethics in Cyber Security & Cyber Law:** Privacy, Intellectual Property, Professional Ethics, Freedom of Speech, Fair User and Ethical Hacking, Trademarks, Internet Fraud, Electronic Evidence, Cybercrimes.

## Unit 3

**Penetration Testing:** Overview of the web from a penetration testers perspective, Exploring the various servers and clients, Discussion of the various web architectures, Discussion of the different types of vulnerabilities, Defining types of penetration testing.

**Web Application Security:** Common Issues in Web Apps, what is XSS, SQL injection, CSRF, Password Vulnerabilities, SSL, CAPTCHA, Session Hijacking, Local and Remote File Inclusion, Audit Trails, Web Server Issues.

## Unit 4

**Forensics & Network Assurance:** Forensic Technologies, Digital Evidence Collection, Evidentiary Reporting, Layered Défense, Surveillance and Reconnaissance, Outsider Thread **Protection. Information Risk Management:** Asset Evaluation and Business Impact Analysis, Risk Identification, Risk Quantification, Risk Response Development and Control, Security Policy, Compliance, and Business Continuity. Forensic investigation using Access Data FTK.

## Unit 5

**UNIT-V**

**Cyber Incident Analysis and Response:** Incident Preparation, Incident Detection and Analysis. Containment, Eradication, and Recovery. Proactive and Post-Incident Cyber Service, CIA Triangle.

**TEXTBOOKS:**

1.The Official CHFI Study Guide for Computer Hacking Forensic Investigator by Dave Kleiman.

2. CISSP Study Guide, 6th Edition by James M. Stewart.

3. Title: Cyber Forensics by Dejey & S.Murugan, OXFORD University Press.

**REFERENCE BOOKS:**

1. Introduction to Information Security and Cyber Laws (English, Paperback, Tripathi Surya Prakash)

| Course code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int. | Ext | |
| **CSM4103** | **Deep Learning (Elective- V)** | **3** | **0** | **0** | **30** | **70** | **3** |

## Course Objectives:

1. Learn deep learning fundamentals, regularization techniques, and optimization methods.
2. Comprehend the architecture and applications of convolutional neural networks.
3. Understand various sequence modeling techniques, including recurrent and recursive networks.
4. Design and implement real-time deep learning applications.
5. Understand and apply different types of autoencoders and deep generative models

## Course Outcomes:

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

CO1: Understand deep learning, regularization techniques, and optimization methods, enabling them to design and train effective deep learning models.

CO2: Understand the convolutional neural networks.

CO3: Describe various sequence modeling techniques.

CO4: Design various real time deep learning applications.

CO5: Understand various types of Auto encoders and different techniques of Deep Generative Models.

**UNIT-I**

**Introduction to Deep Learning:** Introduction - Curse of Dimensionality, Historical trends in Deep Learning, Deep feed - Networks, Gradient Learning, Hidden Units, Architecture design, back propagation, activation functions.

**Regularization and Optimization:** Regularization techniques- L1, L2, dataset augmentation, early stopping, dropout, surrogate loss functions, batch and mini batch algorithms. batch normalization, Comparison of shallow learning and deep learning, Importance and Applications of Deep Learning.

## UNIT-II

**Convolutional Networks:** The Convolution Operation, Pooling, Convolution, Basic Convolution Functions, Structured Outputs, Data Types, Efficient Convolution Algorithms, Random or Unsupervised Features, The Neuroscientific Basis for Convolutional Networks, Dataset Augmentation, Image Captioning.

## UNIT- III

**Sequence Modeling- Recurrent and Recursive Nets:** Unfolding Computational Graphs, Recurrent Neural Networks, Bidirectional RNNs, Encoder-Decoder Sequence-to-Sequence Architectures, Deep Recurrent Networks, Recursive Neural Networks, Leaky Units, LSTM.

## UNIT-IV

**Practical Methodology:** Performance Metrics, Default Baseline Models, Determining Whether to Gather More Data, Selecting Hyper parameters, Debugging Strategies, Case study: Multi-Digit Number Recognition.

**Applications:** Large-Scale Deep Learning, Computer Vision, Speech Recognition, Natural Language Processing, Other Applications.

## UNIT-V

**Autoencoders:** Undercomplete Autoencoders, Regularized Autoencoders, Representational Power, Layer Size and Depth Stochastic Encoders and Decoders, Denoising Autoencoders, Applications of Autoencoders.

**Deep Generative Models:** Directed Generative Nets - Variable Autoencoders, Generative adversarial networks, Representation Learning.

**Textbooks:**

1. Ian Goodfellow, Yoshua Bengio, Aaron Courville, “Deep Learning”, MIT Press,2016.

**Reference Books:**

1. Josh Patterson and Adam Gibson, “Deep learning: A practitioner's approach”,

O'Reilly Media, First Edition, 2017.

2. Fundamentals of Deep Learning, Designing next-generation machine intelligence algorithms, Nikhil Buduma, O'Reilly, Shroff Publishers, 2019.
3. Deep learning CookBook, Practical recipes to get started Quickly, Douwe Osinga, O'Reilly, Shroff Publishers, 2019.

# GAYATRI VIDYA PARISHAD COLLEGE FOR DEGREE AND PG COURSES (AUTONOMOUS)
# ENGINEERING AND TECHNOLOGY PROGRAM
# RUSHIKONDA CAMPUS, VISAKHAPATNAM–530045.

| Course code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int. | Ext | |
| CSM4104 | R-Programming | 3 | 0 | 0 | 30 | 70 | 3 |

**Course Objectives:**

The objectives of this course are to:

1. Introduce the R environment, language fundamentals, and philosophy of statistical computing.
2. Develop proficiency in data structures, data manipulation, and exploratory data analysis using R.
3. Teach effective programming practices including control structures, functions, debugging, and performance profiling.
4. Enable students to perform data import/export, data cleaning, transformation, and visualization in R.
5. Apply R programming to real-world data analysis projects, including simulation and case studies

**Course Outcomes (COs)**

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

CO1: Explain the history, installation, and core components of the R ecosystem in data science.

CO2: Perform efficient data input/output, manipulation, cleaning, and transformation using base R and tidyverse.

CO3: Develop robust R scripts with control structures, functions, and best programming practices.

CO4: Create publication-quality visualizations, debug and profile code, and manage memory for scalable analyses.

CO5: Conduct simulations and end-to-end case study analyses with reproducible reporting using R Markdown and Git.

.

## Unit I

**Introduction and R Foundations:** History and overview of R: origins, philosophy, and ecosystem, Installation and setup: R and RStudio environments; basic interface (console, script editor, working directory).R objects and atomic classes: numeric, integer, complex, logical, character. Core data structures: vectors, matrices, arrays, lists, factors, and data frames

## Unit 2

**Data Input/Output and Manipulation:** Reading and writing data: read.table(), read.csv(), readr package; efficient import of large datasets. Data serialization: save(), load(), dput(), dump(); binary formats and memory considerations. Interfaces: file connections, reading from URLs and external sources. Managing missing values and data cleaning strategies. Data manipulation with base R: subsetting ([], [[ ]], $), vectorized operations, matrix arithmetic.Introduction to tidyverse: dplyr verbs (select, filter, arrange, mutate, rename, group by, summarize) and tidyr (pivot_longer, pivot_wider). Handling dates and times: Date, POSIXct, and lubridate basics.

## Unit 3

**Programming Constructs and Functional Programming:** Control structures: if-else, for loops, while loops, repeat, break, next. Writing functions: arguments, defaults, lazy evaluation, operator; writing reusable code. Scoping rules and environments: lexical scoping, closures; best practices for modular code. Loop functions and apply family: lapply, sapply, vapply, tapply, mapply, purrr map functions. Vectorizing user-defined functions for performance. Coding standards and best practices: style guidelines, version control with Git.

## Unit 4

**Visualization, Debugging, and Performance:** Base graphics: plots, histograms, boxplots, customizing visuals. ggplot2 fundamentals: grammar of graphics, layering, aesthetics, themes, and extensions. Exploratory Data Analysis workflows combining visualization and summary statistics. Debugging tools: traceback(), debug(), recover(), browser(), error handling techniques. Profiling R code: system.time(), Rprof(), summaryRprof(); identifying and addressing performance bottlenecks. Memory management: inspecting object sizes, garbage collection, efficient data handling for large datasets.

## Unit 5

**Simulation, Case Studies, and Reproducible Reporting:** Random number generation: set.seed(), generating samples; simulation of statistical models and Monte Carlo experiments. Case studies: domain-specific data analysis (e.g., environmental data, finance, healthcare); applying end-to-end workflows. Reproducible research: R Markdown for integrating code, outputs, and narrative; version control of reports. Project workflow: data acquisition, cleaning, analysis, visualization, interpretation, and presentation. Final project: students undertake an end-to-end analysis in R, deliver a reproducible report, and present findings.

**TEXTBOOKS:**

1. Roger D. Peng, *R Programming for Data Science*, Leanpub, 2015.
2. Garrett Grolemund & Hadley Wickham, *R for Data Science*, O'Reilly Media, 2017.

**REFERENCE BOOKS:**

1. Hadley Wickham, *Advanced R*, Chapman & Hall/CRC, 2014.
2. Paul Murrell, *R Graphics*, Chapman & Hall/CRC, 2005.
3. William N. Venables & Brian D. Ripley, *Modern Applied Statistics with S*, Springer, 2002.
4. Norman Matloff, *The Art of R Programming*, No Starch Press, 2011.
5. Yihui Xie, *R Markdown: The Definitive Guide*, CRC Press, 2020.

# GAYATRI VIDYA PARISHAD COLLEGE FOR DEGREE AND PG COURSES (AUTONOMOUS)
# ENGINEERING AND TECHNOLOGY PROGRAM
# RUSHIKONDA CAMPUS, VISAKHAPATNAM–530045.

| Course code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int. | Ext | |
| CSM4105 | Robotics | 3 | 0 | 0 | 30 | 70 | 3 |

**Course Objectives:**

The objectives of this course are to:

1. Understand and discuss the fundamental elementary concepts of Robotics.
2. Provide insight into different types of robots.
3. Explain intelligent module for robotic motion control.
4. Educate on various path planning techniques.
5. Illustrate the working of innovative robotic devices

**Course Outcomes (COs)**

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

CO1: Understand the significance, social impact and prospects of robotics and automation in various engineering applications.

CO2: Identify and describe the components and anatomy of robotic system.

CO3: Know about various path planning techniques and analyse different motions of robotics system CO4: Use the suitable drives and end-effectors for a given robotics application.

CO5: Apply robotics concept to automate the monotonous and hazardous tasks and categorize various types of robots based on the design and applications in real world scenarios.

## Unit I

**Introduction To Robotics:** Introduction to Robotics and Automation, laws of robot, brief history of robotics, basic components of robot, robot specifications, classification of robots, human system and robotics, safety measures in robotics, social impact, Robotics market and the prospects, advantages and disadvantages of robots.

## Unit 2

**Robot Anatomy and Motion Analysis:** Anatomy of a Robot, Robot configurations: polar, cylindrical, Cartesian, and jointed arm configurations, Robot links and joints, Degrees of freedom: types of movements, vertical, radial and rotational traverse, roll, pitch and yaw, Wok

volume/envelope, Robot kinematics: Introduction to direct and inverse kinematics, transformations and rotation matrix.

**Unit 3**

**Robot Drives And End Effectors:** Robot drive systems: Hydraulic, Pneumatic and Electric drive systems, classification of end effectors, mechanical grippers, vacuum grippers, magnetic grippers, adhesive gripper, gripper force analysis and gripper design, 1 DoF, 2 DoF, multiple degrees of freedom robot hand, tools as end effectors, Robot control types: limited sequence control, point-to-point control, playback with continuous path control, and intelligent control.

**Unit 4**

**Path Planning:** Definition-Joint space technique, Use of P-degree polynomial-Cubic, polynomial- Cartesian space technique, parametric descriptions, straight line and circular paths, position and orientation planning.

**Unit 5**

**Robotics Applications: Material Handling:** pick and place, palletizing and depalletizing, machining loading and unloading, welding & assembly, Medical, agricultural and space applications, unmanned vehicles: ground, Ariel and underwater applications, robotic for computer integrated manufacturing. Types of robots: Manipulator, Legged robot, wheeled robot, aerial robots, Industrial robots, Humanoids, Robots, Autonomous robots, and Swarm robots

**Textbooks**:

1. S.R. Deb, Robotics Technology and flexible automation, Tata McGraw-Hill Education, 2009.

2. Mikell P. Groover et. al., "Industrial Robots - Technology, Programming and Applications", McGraw Hill, Special Edition, (2012).

3. Ganesh S Hegde, “A textbook on Industrial Robotics”, University science press, 3rdedition, 2017.

**Reference Books**:

1. Richard D Klafter, Thomas A Chmielewski, Michael Negin, "Robotics Engineering – An Integrated Approach", Eastern Economy Edition, Prentice Hall of India Pvt. Ltd., 2006.

2. Fu K S, Gonzalez R C, Lee C.S.G, "Robotics: Control, Sensing, Vision and Intelligence",McGraw Hill, 1987. https://www.robots.com/applications.

# GAYATRI VIDYA PARISHAD COLLEGE FOR DEGREE AND PG COURSES (AUTONOMOUS)
# ENGINEERING AND TECHNOLOGY PROGRAM
# RUSHIKONDA CAMPUS, VISAKHAPATNAM–530045.

| Course code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int. | Ext | |
| **CSM4107** | **POWER BI** | **1** | **0** | **2** | **50** | **50** | **2** |

**Course Objectives:**

1. Learn fundamental concepts of Business Intelligence and Power BI
2. Develop skills in data connectivity, transformation, and visualization
3. Master advanced analytics, DAX functions, and enterprise-level reporting

**Course Outcomes:**

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

C01: Connect to various data sources and perform data transformation using Power Query

C02: Create interactive reports, dashboards, and implement security measures

C03: Develop advanced analytics solutions using DAX and optimize Power BI performance

**Week 1: Introduction & Ecosystem**

**Focus**: Getting started with Power BI.

**Theory Topics**:

- Benefits of Power BI in BI workflows.
- Power BI ecosystem components: Desktop, Service (cloud), Mobile.
- Licensing options (Free, Pro, Premium Per User, Premium capacity): comparison, use cases.
- Installation and initial setup of Power BI Desktop.

**Practical Exercises / Labs**:

1. Install Power BI Desktop; verify version and updates.
2. Explore interface: panes (Fields, Visualizations), ribbons, Report/Data/Model views.
3. Create first simple report: load sample data, build a bar chart and a slicer.
4. Organize files/workspaces locally: save PBIX, folder structure for projects.

**Week 2: Data Connectivity & Import Methods**

**Focus**: Connecting to data sources and understanding modes.

**Theory Topics**:

- Import vs Direct Query: definitions, pros/cons, performance and freshness trade-offs.
- Direct Query specifics: supported sources, limitations on transformations/DAX.
- Import mode specifics: data compression, refresh scheduling.
- Connectivity best practices: filtering early, minimizing data transferred.

**Practical Exercises / Labs**:

1. Connect to a Google Sheet via Web connector; inspect data privacy settings.
2. Connect to Azure Blob Storage: load a CSV from blob.
3. Connect to Snowflake or simulate with another relational database, review credentials and privacy levels.
4. Change data source for an existing query (e.g., switch from local CSV to cloud file).
5. Practice imports from various file formats: Excel, CSV, JSON; observe data type detection.

**Week 3: Power Query & Data Transformation**

**Focus**: Cleaning and shaping data in Power Query Editor.

**Theory Topics**:

- Power Query Editor overview: Applied Steps pane, formula bar, Query Settings.
- Data cleaning principles: handling missing or inconsistent data.
- Column quality, distribution, and profile concepts for data assessment.

**Practical Exercises / Labs**:

1. Add an Index column: discuss scenarios (e.g., ordering, merges).

2. Fill Down/Up missing values in hierarchical data.
3. Merge/join tables on multiple columns; inspect join types (Left, Inner, etc.) and row counts.
4. Convert data types; handle errors using "Replace Errors" or custom conditional columns.

**Week 4: Basic Visualizations & Design Principles**

**Focus**: Core visuals and design best practices.

**Theory Topics**:

- Chart types and their use cases: bar, line, pie (sparingly), table/matrix for details.
- Visual design principles: clarity, minimalism, appropriate labels, avoiding clutter.
- On-object formatting in Power BI Desktop: quick formatting techniques.

**Practical Exercises / Labs**:

1. Build basic visuals: Bar, Line, Pie, and Table from sample data.
2. Format visuals on-object: adjust titles, axes, data labels, and color palettes.
3. Enable total labels on stacked visuals.
4. Display values on rows for matrix visuals; adjust subtotals.

**Week 5: Data Modelling Fundamentals**

**Focus**: Designing a robust data model.

**Theory Topics**:

- Star vs snowflake schemas; pros/cons and best practices.
- Relationships and cardinality (one-to-many, many-to-many considerations).
- Importance of a Date/Calendar table for time intelligence.
- Hierarchies and grouping concepts for user navigation.

**Practical Exercises / Labs**:

1. Create and manage relationships in Model view; set correct cardinalities.
2. Build hierarchies (e.g., Year > Quarter > Month) for date fields.
3. Create Groups (buckets) for numeric or categorical fields.
4. Create a dedicated Measure table; move measures there for clarity.
5. Use Display Folders to group related measures logically.

**Week 6: Advanced Visualizations & Custom Visuals**

**Focus**: Enhancing reports with KPIs and custom visuals.

**Theory Topics**:

- Native vs custom visuals: advantages, potential performance impact, governance.
- KPI concepts and effective design.
- Basics of multi-dimensional analysis in visuals.

**Practical Exercises / Labs**:

1. Create KPI visuals: define target vs actual values; add trend indicators.
2. Build small multiple line charts to compare series across categories.
3. Install & configure a “Multi KPI” custom visual from the Marketplace.
4. Create a Table Heatmap visual: apply color scales within tables to highlight high/low values.
5. Use SVG-based visuals/icons for dynamic KPI displays.

**Week 7: Interactive Features & UX**

**Focus**: Report interactivity and user experience design.

**Theory Topics**:

- Interactivity concepts: slicers, drill-through, tooltips.
- UX principles: intuitive navigation, storytelling with data.
- Bookmarking and buttons for guided navigation.

**Practical Exercises / Labs**:

1. Sync slicers across multiple pages for consistent filtering.
2. Create custom report page tooltips to show extra details on hover.
3. Implement Page Navigation using Bookmarks & Buttons to simulate a dashboard flow.
4. Configure Drill-through to detail pages (e.g., click a region to see regional breakdown).
5. Add "Clear All Slicers" functionality via bookmark and DAX measure trick.
6. Use "Apply all filters" features/bookmarks to maintain consistent states across pages.

**Week 8: Dynamic Content & Conditional Formatting**

**Focus**: Making reports responsive to user selections.

**Theory Topics**:

- Dynamic content concepts: using DAX measures for titles and labels.
- Conditional formatting principles: highlighting insights, anomalies.

**Practical Exercises / Labs**:

1. Create Dynamic Title measures that reflect slicer selections.
2. Handle multi-select scenarios for dynamic titles (e.g., show "Multiple Regions Selected" when more than one).
3. Apply conditional formatting on visuals by numeric field values (e.g., color bars red/green based on thresholds).
4. Conditional formatting based on string fields (e.g., flag status "Delayed" in red).
5. Dynamically change measures (e.g., Top N items) based on slicer input.
6. Change column headers dynamically using field parameters and measures.
7. Apply conditional formatting for data labels (e.g., highlight labels above a target).

**Week 9: DAX Fundamentals**

**Focus**: Core DAX concepts and practices.

**Theory Topics**:

- Introduction to DAX language: syntax and patterns.
- Row context vs filter context; evaluation context basics.
- Readable DAX: indentation, comments, naming conventions.
- Calculated columns vs measures: storage, performance considerations.

**Practical Exercises / Labs**:

1. Write basic DAX measures (SUM, AVERAGE, ratio calculations).
2. Create calculated columns vs measures; discuss when to use each and performance impact.
3. Implement time intelligence functions (TOTALYTD, SAMEPERIODLASTYEAR) using a proper Date table.
4. Use Quick Measures to auto-generate DAX; inspect and learn from underlying formulas.
5. Solve sample scenarios: Year-over-Year growth, moving averages, percentage of total, ranking.

**Week 10: Security & Access Control**

**Focus**: Securing data at row level and managing permissions.

**Theory Topics**:

- Power BI security concepts overview.
- Row-Level Security (RLS): static vs dynamic approaches.
- Using USERPRINCIPALNAME() in dynamic RLS.
- Workspace roles and permissions in Power BI Service (Admin, Member, Contributor, Viewer).

**Practical Exercises / Labs**:

1. Define static RLS roles in Desktop; test with "View as Role" functionality.
2. Implement dynamic RLS based on user login via a mapping table and USERPRINCIPALNAME().
3. Model organizational hierarchy for RLS (e.g., manager-subordinate data access).
4. Test various RLS scenarios; troubleshoot common issues.
5. Explore and assign Power BI Service workspace roles; understand implications for editing and sharing.

**Week 11: Power BI Service & Collaboration**

**Focus**: Deploying and collaborating in Power BI Service.

**Theory Topics**:

- Power BI Service overview: workspaces, apps, dashboards vs reports.
- Data Gateway concepts: personal vs standard; use cases.
- Sharing and collaboration strategies: publishing, apps, permissions.
- Subscriptions, data alerts, and monitoring.

**Practical Exercises / Labs**:

1. Publish Power BI Desktop report to Power BI Service; configure dataset settings and refresh.
2. Create dashboards by pinning visuals; differentiate dashboards from report pages.
3. Install and configure On-Premises Data Gateway in Personal and Standard modes.
4. Set up scheduled refresh with proper credential management.
5. Subscribe to reports/dashboards; configure email notifications for stakeholders.

6. Configure data alerts on KPIs (e.g., notify when metric crosses threshold).
7. Create and manage Power BI Apps: package multiple reports/dashboards, manage access.

**Week 12: Performance Optimization & Enterprise Features**

**Focus**: Tuning performance and enterprise deployment patterns.

**Theory Topics**:

- Performance optimization principles: model size, DAX efficiency, visual complexity.
- Using Performance Analyzer Tool in Desktop.
- Advanced analytics concepts: aggregations, incremental refresh, composite models.
- Enterprise deployment strategies: dataflows, embedding reports, governance best practices.
- Custom branding and theming in Service.

**Practical Exercises / Labs**:

1. Enable Performance Analyzer; capture query durations vs rendering times; interpret findings.
2. Set visual query limits; handle scenarios with large datasets (e.g., summarization).
3. Optimize data model: remove unused columns, create aggregations, configure incremental refresh.
4. Apply DAX optimization techniques: use variables, avoid unnecessary context transitions, simplify expressions.
5. Create a Power BI Dataflow in Service for reusable ETL; demonstrate linking dataflows to datasets.
6. Embed a report in Microsoft Teams or a simple web portal; verify access/security.
7. Apply custom branding/theme in Service (custom color palette, logo).

8. Document the optimization steps and results in a brief case study (not in table).

- **Quizzes/Concept Checks**: Short quizzes or reflection questions after key topics (e.g., "Explain Import vs DirectQuery trade-offs," "Describe row vs filter context in DAX").
- **Lab Submissions**: Evaluate based on correctness of transformations, model design, visual clarity, and application of best practices.
- **Mini-Project Presentations**: Peer review focusing on UX, performance, and interactivity.
- **Capstone Evaluation**: Assess completeness (end-to-end flow), technical soundness (correct modeling, DAX, RLS), performance optimizations, documentation quality, and demonstration clarity.

**Resources & References**

**Textbooks**:

1. *The Definitive Guide to DAX* (2nd Ed) by Marco Russo & Alberto Ferrari.
2. *Mastering Microsoft Power BI* by Brett Powell.
3. *Power BI Cookbook* by Brett Powell.

# GAYATRI VIDYA PARISHAD COLLEGE FOR DEGREE AND PG COURSES (AUTONOMOUS)
# ENGINEERING AND TECHNOLOGY PROGRAM
# RUSHIKONDA CAMPUS, VISAKHAPATNAM–530045.

| Course code | Title of the Course | Contact Hours/week | | | Allotment of Marks | | Credits |
|---|---|---|---|---|---|---|---|
| | | L | T | P | Int. | Ext | |
| CSM4-1 Honors | Multiagent Systems | 4 | 0 | 0 | 30 | 70 | 4 |

**Course Objectives:**

The objectives of this course are to:

1) Key objectives include understanding agent concepts, decision-making frameworks, communication and coordination mechanisms.
2) To design, analyse, and implement systems involving multiple interacting agents.
3) To make students to understand the application of game theory and other computational techniques to model and solve problems in distributed environments.

**Course Outcomes (COs)**

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

1) Understanding of what agents are, and some of the issues associated with building them.
2) Apply the knowledge of how the agents use logical rules to derive specific conclusions to problem-solving in various contexts.
3) Gain the ability to classify different types of multi-agent systems.
4) Acquire advanced knowledge of formal logics used for reasoning about multi-agent systems.
5) Apply the knowledge of Multiagent Systems to various fields and domain areas.

**Unit I**

**Intelligent Agents :** Environments , Intelligent Agents, Agents and Objects , Agents and Expert Systems, Agents as Intentional Systems, Abstract Architectures for Intelligent Agents, How to Tell an Agent What to Do, Synthesizing Agents.

**Unit 2**

**Deductive Reasoning Agents:** Agents as Theorem Provers, Agent-Oriented Programming, Practical Reasoning Agents, Practical Reasoning Equals Deliberation Plus Means, Ends

Reasoning, Means-Ends Reasoning, Implementing a Practical Reasoning Agent, HOMER: an Agent That Plans.

**Unit 3**

**Multiagent Interactions:** Utilities and Preferences, Multiagent Encounters, Dominant Strategies and Nash Equilibria, Competitive and Zero-Sum Interactions, The Prisoner's Dilemma, Other Symmetric 2x 2 Interactions, Dependence Relations in Multiagent Systems.

**Unit 4**

**Logics for Multiagent Systems:** Why Modal Logic, Possible-Worlds Semantics for Modal Logics, Normal Modal Logics, Epistemic Logic for Multiagent Systems, Pro-attitudes: Goals and Desires, Common and Distributed knowledge, Integrated Theories of Agency

**Unit 5**

**Applications**: Agents for Workflow and Business Process Management, Agents for Distributed Sensing, Agents for Information Retrieval and Management, Agents for Electronic Commerce, Agents for Human-Computer Interfaces, Agents for Virtual Environments, Agents for Social Simulation .

**TEXTBOOKS:**

1. An Introduction to MultiAgent Systems by MICHAEL WOOLDRIDGE, JOHN WILEY & SONS, LTD, 2nd Edition.

**REFERENCE BOOKS:**

1. Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations by KeviLeyton-Brown , Yoav Shoham,
2. Cambridge University Press Multiagent Systems by Gerhard Weiss , 2nd Edition.

# B. Tech Computer Science and Engineering with AI & ML
## (R-22 Regulation)
### IV Year – II Semester

| Course code | Category | Course Title | Hours per week | | | Internal Marks | External Marks | Total Marks | Credits |
|---|---|---|---|---|---|---|---|---|---|
| | | | L | T | P | | | | |
| Proj | Major Project | Project Work, Seminar & Internship in Industry | 0 | 0 | 0 | 50 | 50 | 100 | 12 |
| | | | | | | | | **Total Credits** | **12** |