CURRICULUM
FOR
SIXTH SEMESTER
OF
THREE-YEAR DIPLOMA
COURSE
IN
COMPUTER
ENGINEERING

STUDY SCHEME FOR SIXTH SEMESTER COMPUTER ENGINEERING

Code	Subjects	Study Scheme			Total Hours L+T+P	CREDITS			Total Credits L+T+P
		Periods Per Week							
		L	T	P		L	T	P	
COPC601	Software Project Management	3	0	0	3	3	0	0	3
COPC602	Artificial Intelligence and Neural Networks	3	0	0	3	3	0	0	3
COPC603	Artificial Intelligence and Neural Networks Lab	0	0	2	2	0	0	1	1
****	Elective-III	3	0	0	3	3	0	0	3
****	Elective-III Lab	0	0	2	2	0	0	1	1
	Open Elective	3	0	0	3	3	0	0	3
MP607	Major Project	0	0	16	16	0	0	8	8
		12	**0**	**20**	**32***	**12**	**0**	**10**	**22**

***Extra two hours shall be taken as supplementary classes per week**

***: The students have to choose Elective-III from the common pool of program electives and Open Elective from the list of Open Electives given at the end of the curriculum for Computer Engineering/IT. Furthermore, for the lab courses of program electives, it is mandatory that the lab course chosen is the corresponding lab course of the program elective only.

PROGRAM: THREE YEAR DIPLOMA IN COMPUTER ENGINEERING	
Course Code: CO**PC601**	Course Title: **Software Project Management**
Semester: **6th**	Credit: **3**
Periods Per Week: **3 (L: 03, T: 0, P: 0)**	

COURSE OBJECTIVE:

This course is designed to develop an understanding of basic software engineering techniques with the focus on requirements , design , planning and development of the software applications, with will be used by the students at industry level.

COURSE CONTENT:

1. Introduction to Software S/W Engineering

1.1 Introduction, size factors. Quality and productivity factors.
1.2 Management issues, Models: waterfall, spiral, prototyping, fourth generation techniques, s/w process
1.3 Introduction to agile technologies

2. Planning

2.1 The development process
2.2 an organizational structure
2.3 other planning activities

3. Software Cost Estimations

3.1 Cost factors.
3.2 Estimations techniques. Staffing level estimation, estimating software maintenance costs.
3.3 Cost model.
3.4 COCOMO model.

4. Software Requirements Definitions

4.1 Problem analysis, requirement engineering.
4.2 The software requirements specifications (SRS).

4.3 Formal specifications techniques, characteristics of a good SRS
4.4 Characteristics of a good SRS

5. Software Design and Implementation Issue

5.1 Fundamental design, concept design notations, design techniques
5.2 Structured coding techniques coding styles,
5.3 Documentation guidelines

6. Verification and Validation Techniques

6.1 Quality assurance work through and inspections static analysis
6.2 Symbolic execution unit testing, formal verifications
6.3 Black box and white box testing techniques

7. Software Maintenance

7.1 Maintenance Overview.
7.2 Configuration Management
7.3 Software Quality

COURSE OUTCOME:

After the completion of the course the student will be able to:

- Apply the software engineering lifecycle by demonstrating competence in communication, planning, analysis, design, construction, and deployment
- Work in one or more significant application domains
- Work as an individual and as part of a multidisciplinary team to develop and deliver quality software
- Demonstrate an understanding of and apply current theories, models, and techniques that provide a basis for the software lifecycle

RECOMMENDED BOOKS:

1. Software Engineering by Rajib Mall, PHI Publishers, New Delhi
2. An Integrated Approach to Software Engineering by Pankaj Jalote, Narosa Publishing House Pvt Ltd, Darya Ganj, New Delhi 110002
3. Software Engineering, Sangeeta Sabharwal, New Age International, Delhi
4. Software Engineering by KK Aggarwal and Yogesh Singh
5. Software Engineering – A Practitioner's Approach by RS Pressman, Tata McGraw Hill Publishers, New Delhi
6. Eagles's Software Engineering By Gaurav Gupta, Dipika Goel

UNIT WISE TIME AND MARKS DISTRIBUTION

Topic No.	Time Allotted (Hrs)	Marks Allotted (%)
1	08	15
2	07	15
3	09	20
4	07	15
5	07	15
6	05	10
7	05	10
Total	**48**	**100**

PROGRAM: THREE YEAR DIPLOMA IN COMPUTER ENGINEERING	
Course Code: **COPC602**	Course Title: **Artificial Intelligence and Neural Networks**
Semester: **6th**	Credit: **3**
Periods Per Week: **03 (L: 03, T: 00, P: 00)**	

COURSE OBJECTIVES:

This course aims to provide students with a comprehensive understanding of Artificial Intelligence (AI) fundamentals and their practical applications. Through a series of hands-on practicals, students will gain proficiency in key AI concepts such as search techniques, knowledge representation, machine learning, and neural networks. By the end of the course, students will be able to develop basic chatbots, solve complex problems using search algorithms, analyze data with regression and classification models, and implement neural networks for various tasks. Additionally, they will explore the ethical implications and societal impact of AI technologies, equipping them to contribute responsibly in the field of AI development and application.

COURSE CONTENT:

1. INTRODUCTION TO ARTIFICIAL INTELLIGENCE

1.1. Introduction to AI and its applications,
1.2. Brief historical perspective of AI
1.3. Agent and Environment
1.4. Types of AI systems
1.5. Applications of AI in healthcare, gamming and education.

2. SEARCH TECHNIQUES

2.1. Uniformed Search Techniques:
  2.1.1. Breadth First Search
  2.1.2. Depth First Search
2.2. Heuristic Search Techniques
  2.2.1. Hill climbing
  2.2.2. Best first search
  2.2.3. Mini-max search
  2.2.4. A* algorithm

3. KNOWLEDGE REPRESENTATION

3.1 Prepositional Logic—formulation of compound Prepositional from simpler ones using AND, OR, NOT, IMPLES and IFF connectives
3.2 Predicate Logic/First Order Logic) –Rule Based deduction systems, problem solving using predicates and quantifiers (Universal Quantifier, Existential Quantifier)

4. FUNDAMENTALS OF MACHINE LEARNING

- 4.1. Introduction to Machine Learning.
- 4.2. Overview of Supervised, Unsupervised and reinforcement Learning.
- 4.3. Various Learning techniques:
  - 4.3.1. Regression (Supervised ML)
    - 4.3.1.1. Linear Regression Algorithm
  - 4.3.2. Classification (Supervised ML)
    - 4.3.2.1. KNN Algorithm
  - 4.3.3. Clustering (Unsupervised ML).
    - 4.3.3.1. K-Means Clustering Algorithm

5. FUNDAMENTALS OF NEURAL NETWORKS

- 5.1. Introduction to neural networks and their applications,
- 5.2. Structure and working principles of artificial neurons (Perceptron)
- 5.3. Activation functions and their significance
- 5.4. Basic Overview of the Concept of Back-Propagation
- 5.5. Types of neural networks (feedforward, convolutional, recurrent)

COURSE OUTCOME:

After completion of the course the student will be able to:

- Understand the fundamentals of Artificial Intelligence (AI) and its various applications in healthcare, gaming, and education.
- Explain the history and evolution of AI, and recognize different types of AI systems, with a focus on expert systems.
- Apply heuristic search techniques, including Hill climbing, Branch and Bound, Best-first search, and A* algorithm, for problem-solving purposes.
- Gain a foundational understanding of Neural Networks, including their applications, artificial neuron structure, and activation functions.
- Demonstrate basic knowledge of Machine Learning and its main types: Supervised, Unsupervised, and Reinforcement Learning, along with various learning techniques like Regression, Classification, Transfer Learning, and Clustering.

RECOMMENDED BOOKS:

1. "Introduction to Artificial Intelligence"Author: Pradip K. Das Publisher: PHI Learning Private Limited

2. "Fundamentals of Neural Networks":Author: Mohan M. Trivedi Publisher: John Wiley & Sons Inc

3. "Fundamentals of Machine Learning":Author: S. B. Kotsiantis Publisher: Informatics

4. "Artificial Intelligence: Foundations of Computational Agents" by David L. Poole and Alan K. Mackworth

5. "Artificial Intelligence: A Modern Approach" by Stuart Russell and Peter Norvig
6. "Deep Learning with Python" by Francois Chollet
7. "Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques for Building Intelligent Systems" by Aurélien Géron

UNIT WISE TIME AND MARKS DISTRIBUTION

Unit No.	Time Allotted (Hrs)	Marks Allotted (%)
1	08	15
2	08	15
3	11	25
4	10	20
5	11	25
TOTAL	**64**	**100**

PROGRAM: THREE YEAR DIPLOMA IN COMPUTER ENGINEERING	
Course Code: **COPC603**	Course Title: **Artificial Intelligence and Neural Networks Lab**
Semester: **6th**	Credit: **1**
Periods Per Week: **02 (L: 00, T: 00, P: 02)**	

COURSE OBJECTIVE

This course aims to provide students with a comprehensive understanding of Artificial Intelligence (AI) fundamentals and their practical applications. Through a series of hands-on practicals, students will gain proficiency in key AI concepts such as search techniques, knowledge representation, machine learning, and neural networks. By the end of the course, students will be able to develop basic chatbots, solve complex problems using search algorithms, analyze data with regression and classification models, and implement neural networks for various tasks. Additionally, they will explore the ethical implications and societal impact of AI technologies, equipping them to contribute responsibly in the field of AI development and application.

LIST OF PRACTICALS TO BE PERFORMED

1. Review of Python
2. Brief Introduction to Python Libraries-numpy, pytorch, TensorFlow, pygame etc
3. Basic Chatbot Development in Python
4. Implementation of Simple Problems using Breadth First Search Algorithm
5. Implementation of Simple Problems using Depth First Search Algorithm
6. Simple Optimization with Hill Climbing Algorithm
7. Interactive A* Pathfinding in Python
8. Truth Table Generator in Python
9. Introductory Rule-Based System in Python
10. Feedforward Neural Network in PyTorch
11. Activation Function Exploration in TensorFlow

PROGRAM: THREE YEAR DIPLOMA IN COMPUTER ENGINEERING AND INFORMATION TECHNOLOGY	
Course Code: **MP607**	**Course Title**: **Major Project**
Semester: **6th**	Credit: **8**
Periods Per Week: **16 (L: 0, T: 0, P: 16)**	

Introduction: Major Project Work in the 3-year diploma course aims to enhance students' innovative skills by applying the knowledge and skills acquired during their coursework to solve specific problems or undertake projects. As students possess diverse aptitudes and strengths, project assignments should align with their capabilities. This document outlines the guidelines for major project assignments, including types of projects and collaboration with industry.

General Guidelines

1. **Student Preferences:**
   - Students should be encouraged to identify the type of project work they are interested in executing.
   - Faculty should facilitate a process for students to express their preferences regarding project assignments.
2. **Faculty Brainstorming:**
   - Faculty members within each department should collaborate to identify suitable project assignments for students.
   - Projects should be tailored to match students' strengths and interests.
3. **Group Assignments:**
   - Project work can be assigned either individually or in groups.
   - For group assignments, the maximum group size should not exceed five students.
4. **Advance Planning:**
   - Students should either identify themselves or accept the given project assignment at least two to three months in advance.
   - Adequate time should be provided for planning and execution of the project.
5. **Industry Collaboration:**
   - Projects identified in collaboration with industry partners should be preferred.
   - Collaboration with industry helps in real-world application and enhances the relevance of projects.
6. **Faculty Guidance:**
   - Each teacher is expected to guide the project work of 5–6 students.
   - Faculty members should provide necessary support and supervision throughout the project duration.

Types of Project Assignments:

The project assignments may include, but are not limited to:

- Artificial Intelligence and Machine Learning applications
- Internet of Things (IoT) projects
- Game Development
- Enterprise Resource Planning (ERP) Systems
- Project Management Systems
- Cybersecurity solutions and practices
- Cloud computing projects
- Mobile application development
- Virtual and Augmented Reality applications
- Data analytics and visualization projects
- Programming customer-based applications
- Web designing with database connectivity
- Database applications
- Projects on Networking and Network security
- Software Development
- Fabrication of computer-related components/equipment
- Fault diagnosis and rectification of computer systems and peripherals
- Improvements in existing systems/equipment
- Multimedia projects
- Computer Graphics projects

Conclusion: These guidelines aim to ensure that major project work in the 3-year diploma course provides students with opportunities to apply their knowledge and skills effectively. By aligning project assignments with student preferences and industry collaboration, students can enhance their practical skills and contribute meaningfully to the field of computer science.