| Department : **Computer Science and Engineering** | | Programme: **B.Tech. (CS)** | | | | | | |
|---|---|---|---|---|---|---|---|---|
| Semester : **Seventh** | | Course Category Code: **PCC** | | | | Semester Exam Type: **TY** | | |
| Course Code | Course Name | Periods / Week | | | Credit | Maximum Marks | | |
| | | L | T | P | C | CA | SE | TM |
| **CS225** | **Artificial Intelligence** | 3 | - | - | 3 | 40 | 60 | 100 |
| **Prerequisite** | **Nil** | | | | | | | |
| **Course Outcome** | **CO1** | Identify the nature of problems suitable to apply artificial intelligence techniques | | | | | | |
| | **CO2** | Acquire an insight into the different search techniques, knowledge representation and reasoning, planning, and learning strategies for solving Artificial Intelligence problems | | | | | | |
| | **CO3** | Examine case studies on the applications of artificial intelligence techniques | | | | | | |
| | **CO4** | Formulate solutions to real world problems by applying the acquired knowledge | | | | | | |
| | **CO5** | Propose new algorithms on artificial intelligence techniques and validate their results | | | | | | |

| **UNIT-I** | **Introduction to Search Techniques** | **Periods: 9** |
|---|---|---|
| History of AI - Problem-solving through search: state-space - Blind search techniques: BFS, DFS, UCS, - Heuristic search techniques: Best-first search, Greedy search, A* search, AO* search- Adversarial search: Mini-max search - alpha-beta cut off - Problem reduction: AND-OR Graphs - Constraint satisfaction problem - Means Ends Analysis. | | **CO1 CO2** |
| **UNIT-II** | **Knowledge Representation and Inference Techniques** | **Periods: 9** |
| Types of Knowledge - Knowledge Engineering- Approaches for knowledge representation: Propositional Logic, Predicate logic, Representing knowledge using rules, Semantic Networks, Frames, Slots, Conceptual dependency, Scripts - Inference Techniques: Unification, Resolution, Forward and backward reasoning – Conflict Resolution. | | **CO2** |
| **UNIT-III** | **Uncertain Knowledge Representation and Reasoning** | **Periods: 9** |
| Non-Monotonic reasoning - Probabilistic Reasoning – Bayes rule – Bayesian Belief Networks –Causal Reasoning from Bayesian networks - Certainty factors – Fuzzy Logic: Fuzzification, Fuzzy Rule Base, Defuzzification -Reasoning using Fuzzy Logic – Dempster-Shafer Belief Update Theory. | | **CO2** |
| **UNIT-IV** | **Planning and Learning** | **Periods: 9** |
| Planning: State space planning - partial order planning - Planning graphs - Conditional planning- Continuous planning, Planning under uncertainty - Learning Types: Rote Learning, Learning by taking advice, Explanation based learning, Discovery, Analogy - Supervised and Unsupervised learning - Decision trees based learning – Reinforcement Learning. | | **CO2** |
| **UNIT-V** | **Applications of Artificial Intelligence** | **Periods: 9** |
| Expert Systems: Characteristics - Building blocks- Case Study, Intelligent agents: Agent Environment- Case Study - Robotics: Hardware, Perception, Planning - Natural Language Processing: Text classification, Information Retrieval and Information Extraction. | | **CO3 CO4 CO5** |

| **Lecture Periods: 45** | **Tutorial Periods: -** | **Practical Periods: -** | **Total Periods: 45** |
|---|---|---|---|

**Reference Books**


1. Deepak Khemani, A First Course in Artificial Intelligence, First Edition, MgGraw Hill Education (India) Private Limited, 2013
2. Parag Kulkarni and Prachi Joshi, Artificial Intelligence: Building Intelligent Systems, PHI Learning Private Limited, 2015.
3. Stuart Russell and Peter Norvig, Artificial Intelligence: A Modern Approach, Third Edition, Pearson Education Asia, 2015.
4. Vinod Chandra S.S. and Anand Hareendran, Artificial Intelligence and Machine Learning, First Edition, PHI Learning Private Limited, 2014.

| Department : **Computer Science and Engineering** | | Programme: **B.Tech. (CS)** | | | | | | |
|---|---|---|---|---|---|---|---|---|
| Semester : **Seventh** | | Course Category Code: **PCC** | | | | Semester Exam Type: **TY** | | |
| Course Code | Course Name | Periods / Week | | | Credit | Maximum Marks | | |
| | | L | T | P | C | CA | SE | TM |
| **CS226** | **Parallel and Distributed Systems** | 3 | 1 | - | 4 | 40 | 60 | 100 |
| **Prerequisite** | **Nil** | | | | | | | |

| **Course Outcome** | | |
|---|---|---|
| | **CO1** | Understand the architecture of parallel systems and identify the scope for parallelism in present day's processors |
| | **CO2** | Realize and knowing the various parallel computing models and the challenges involved in designing parallel algorithms |
| | **CO3** | Study distributed system models and the components of distributed system |
| | **CO4** | Study the different communication models and naming conventions of distributed systems |
| | **CO5** | Know the collaborative operations of collections of computers and the impacts |

| **UNIT-I** | **Introduction to Parallel Computing Systems** | **Periods: 12** |
|---|---|---|
| Need of high speed computing – increase the speed of computers – history of parallel computers and recent parallel computers; solving problems in parallel – temporal parallelism – data parallelism – comparison of temporal and data parallel processing – data parallel processing with specialized processors – inter-task dependency. Parallel Programming Platforms: Trends in microprocessor architectures - limitations of memory system performance – parallel computing platforms – communication costs in parallel machines – routing mechanisms for interconnection networks. | | **CO1** |
| **UNIT-II** | **Parallel Computation and Communication methods** | **Periods: 12** |
| Principles of Parallel Algorithm Design: Preliminaries – decomposition techniques – characteristics of tasks and interactions – mapping techniques for load balancing – methods for containing interaction overheads – parallel algorithm models. Basic Communication Operations: One-to-all broadcast and all-to-one reduction – all-to-all broadcast reduction – all-reduce and prefix-sum operations – scatter and gather – all-to-all personalized communication – circular shift – improving the speed of some communication operations. | | **CO2** |
| **UNIT-III** | **Introduction to Distributed Systems** | **Periods: 12** |
| Goals – Types of Distributed systems – Architecture styles – System Architecture. Architectures Versus Middleware – Self Management in distributed systems - Processes – Threads – Virtualization – Clients – Servers – Code Migration. | | **CO3** |
| **UNIT-IV** | **Communication and Naming** | **Periods: 12** |
| Communication: Fundamentals - Remote Procedure Call – Stream oriented communication – Message oriented communication – Multicast communication. Naming – Names, Identifiers, and addresses – Flat Naming - Structured Naming – Attribute based Naming. | | **CO3 CO4** |
| **UNIT-V** | **Synchronization, Consistency and Replication** | **Periods: 12** |
| Synchronization: Clock Synchronization – Logical clocks - Mutual Exclusion – Global positioning of nodes - Election Algorithms. Consistency and Replication: Introduction – Data centric consistency models – Client centric consistency models – Replica management – Consistency protocols. | | **CO3 CO5** |

| **Lecture Periods: 45** | **Tutorial Periods:15** | **Practical Periods: -** | **Total Periods: 60** |
|---|---|---|---|

**Reference Books**


1. V. Rajaraman and C. Siva Ram Murthy, Parallel Computers – Architecture and Programming, Prentice-Hall of India, 2003.
2. Ananth Grama, Anshul gupta, George Karypis and Vipin Kumar, Introduction to Parallel Computing, Second Edition, Pearson Education, 2004.
3. Andrew S. Tanenbaum and Maarten Van Steen, Distributed Systems – Principles and Paradigms, Second Edition , Prentice- Hall of India, Pvt. Ltd, 2008
4. Pradeep K Sinha, Distributed Operating Systems, Prentice-Hall of India, New Delhi, 2001.

| Department : **Computer Science and Engineering** | | Programme: **B.Tech. (CS)** | | | | | | |
|---|---|---|---|---|---|---|---|---|
| Semester : **Seventh** | | Course Category Code: **PCC** | | | | Semester Exam Type: **TY** | | |
| Course Code | Course Name | Periods / Week | | | Credit | Maximum Marks | | |
| | | L | T | P | C | CA | SE | TM |
| **CS227** | **Data Science Essentials** | 3 | 1 | - | 4 | 40 | 60 | 100 |
| **Prerequisite** | **Nil** | | | | | | | |
| **Course Outcome** | **CO1** | Ability to have a broad insight, understanding and intuition of the data science life cycle | | | | | | |
| | **CO2** | Demonstrate an ability to use Python to efficiently store retrieve and process data | | | | | | |
| | **CO3** | Discuss in depth a variety of data mining techniques, and their applicability to various problem domains | | | | | | |
| | **CO4** | Select and apply data mining technique to a practical case study | | | | | | |
| | **CO5** | Understand the concept, challenge and technology of big data | | | | | | |
| **UNIT-I** | **Introduction to Data Science** | | | | **Periods: 12** | | | |
| Introduction: Data Science -Epicycles of Analysis-Stating and Refining the Question- Exploratory Data Analysis- Using Models to Explore Data-Inference: A Primer- Formal Modeling-Inference vs. Prediction : Implications for Modeling Strategy -Interpreting results. | | | | | | | | **CO1** |
| **UNIT-II** | **Introduction to Programming Tools for Data Science** | | | | **Periods: 12** | | | |
| Python Basics – Types - Expressions and Variables - String Operations - Python Data Structures - Lists and Tuples – Sets – Dictionaries - Python Programming Fundamentals - Conditions and Branching – Loops – Functions - Objects and Classes - Introduction of Essential Python Libraries – Numpy – Pandas – Matplotlib - Scikit-learn. | | | | | | | | **CO2** |
| **UNIT-III** | **Supervised Learning** | | | | **Periods: 12** | | | |
| Regression - Linear Regression - Logistic Regression - Reasons to Choose and Cautions - Additional Regression Models - Classification - Decision Trees – Na'ive Bayes – Diagnostics of Classifiers – Additional Classification Methods – Time Series Analysis – Overview of Time Series Analysis – ARIMA Model – Additional Methods – Case study with Python. | | | | | | | | **CO3 CO4** |
| **UNIT-IV** | **Unsupervised Learning** | | | | **Periods: 12** | | | |
| Clustering - Overview of Clustering – K-means - Additional Algorithms –Association Rules- Overview - A priori Algorithm - Evaluation of Candidate Rules - Applications of Association Rules - Validation and Testing – Diagnostics - Text Analysis – Text Analysis Steps – Collecting Raw Text – Representing Text – Term Frequency-Inverse Document Frequency (TFIDF) - Categorizing Documents by Topics – Determining Sentiments – Gaining Insights - Case study with Python. | | | | | | | | **CO3 CO4** |
| **UNIT-V** | **Big Data Analytics** | | | | **Periods: 12** | | | |
| Data science in a Big Data world - Benefits and uses of data science and Big Data - Facets of data - The Big Data ecosystem and data science – Introduction of Hadoop - Handling large data on a single computer - The problems in handling large data - General techniques for handling large volumes of data - General programming tips for dealing with large datasets- Case study : Predicting malicious URLs, Recommender system - Steps in Big Data - Distributing data storage and processing with frameworks - Case study: Assessing loan risk. | | | | | | | | **CO5** |
| **Lecture Periods: 45** | **Tutorial Periods: 15** | **Practical Periods: -** | | | **Total Periods: 60** | | | |

**Reference Books**


1. Peng, R. D., & Matsui. E, The Art of Data Science- A Guide for Anyone Who Works with Data, Skybrude Consulting, 2015.
2. Martin Czygan, Phuong Vo.T.H, Getting Started with Python Data Analysis, Packt Publishing, 2015.
3. David Dietrich, Barry Heller & Beibei Yang, Data Science and Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data, John Wiley & Sons, 2015.
4. Davy Cielen, Arno Meysman, Mohamed Ali, Introducing Data Science: Big Data, Machine Learning, and More, Using Python Tools, Manning Publications, 2016.
5. Joel Grus, Data science from scratch: first principles with python, O'Reilly Media, Inc., 2015.
6. Steven S. Skiena, The Data Science Design Manual, First Edition, Springer, 2017.

| Department : **Computer Science and Engineering** | | | Programme: **B.Tech. (CS)** | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| Semester : **Seventh** | | | Course Category Code: **PCC** | | | | Semester Exam Type: **LB** | | |
| Course Code | Course Name | | Periods / Week | | | Credit | Maximum Marks | | |
| | | | L | T | P | C | CA | SE | TM |
| **CS228** | **Artificial Intelligence Laboratory** | | - | - | 3 | 1.5 | 40 | 60 | 100 |
| **Prerequisite** | **Nil** | | | | | | | | |
| **Course Outcome** | **CO1** | Acquire knowledge on how to do logic programming using AI languages | | | | | | | |
| | **CO2** | Construct solutions to apply blind and heuristic search techniques to AI problems | | | | | | | |
| | **CO3** | Illustrate the representation of facts and knowledge in prepositional and predicate logic | | | | | | | |
| | **CO4** | Choose solutions to perform inference on the knowledge base created | | | | | | | |
| | **CO5** | Build expert systems for solving real world problems and validate the results | | | | | | | |

| | |
|---|---|
| 1. Study about the fundamentals of Prolog programming. | **CO1** |
| 2. Execute simple programs using Prolog.<br>a. To represent facts and predicates.<br>b. To read and write input.<br>c. To use operators.<br>d. To use loops.<br>e. To perform list processing. | **CO1** |
| 3. Solve the Water Jug Problem using DFS, BFS blind search algorithms. | **CO2** |
| 4. Implement Mini-max adversarial search algorithm. | **CO2** |
| 5. Implement the Missionaries and cannibals problem using constraint satisfaction method. | **CO2** |
| 6. Find the optimal path between two cities using best first search and A* heuristic algorithms. | **CO2** |
| 7. Represent knowledge using Prepositional Logic and perform inference. | **CO3** |
| 8. Represent knowledge using Predicate Logic and perform inference. | **CO3** |
| 9. Apply unification on a set of facts. | **CO4** |
| 10. Apply forward chaining and backward chaining to infer from a set of facts. | **CO4** |
| 11. Develop an Expert System. | **CO5** |
| 12. Mini project based on industry topics / real time problems. | **CO5** |

| | | | |
|---|---|---|---|
| **Lecture Periods: 45** | **Tutorial Periods: -** | **Practical Periods: -** | **Total Periods: 45** |

**Reference Books**

1. Max Bramer, Logic Programming with Prolog, Springer, 2005.

| Department : **Computer Science and Engineering** | | | Programme: **B.Tech. (CS)** | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| Semester : **Seventh** | | | Course Category Code: **PAC** | | | | Semester Exam Type: **-** | | |
| Course Code | Course Name | | Periods / Week | | | Credit | Maximum Marks | | |
| | | | L | T | P | C | CA | SE | TM |
| **CS229** | **Seminar** | | - | - | 2 | 1 | 100 | - | 100 |
| **Prerequisite** | **Nil** | | | | | | | | |
| **Course Outcome** | **CO1** | Improve oral and written communication skills | | | | | | | |
| | **CO2** | Identify, understand and discuss current technologies | | | | | | | |
| | **CO3** | Learn and integrate through independent learning and collaborative study | | | | | | | |
| | **CO4** | Distinguish and integrate differing forms of knowledge and academic disciplinary approaches | | | | | | | |
| The student will present a seminar on following: | | | | | | | | | |
| Select on a topic in an emerging area in his/her specialization of Computer Science and Engineering.<br>Make a presentation for duration of 20 to 25 minutes.<br>Submit a brief report running to 15 or 20 pages for the purpose of evaluation. | | | | | | | | | **CO1**<br>**CO2**<br>**CO3**<br>**CO4** |
| **Lecture Periods: -** | | **Tutorial Periods: -** | | **Practical Periods: 30** | | | **Total Periods: 30** | | |
| **Reference Books** | | | | | | | | | |
| 1. Books related to the Seminar title.<br>2. Papers published in reputed journals and conferences related to the seminar. | | | | | | | | | |