ANNAMACHARYA INSTITUTE OF TECHNOLOGY AND SCIENCES, TIRUPATI
(AUTONOMOUS)

COMPUTER SCIENCE AND ENGINEERING

(Effective for the batches admitted in 2019-20)

III B. Tech – I Semester (Theory – 7, Lab – 4) – AK19

S. No	Category	Course Code	Course Title	Hours per week			Credits	Scheme of Examination (Max. Marks)		
				L	T	P		CIE	SEE	Total
THEORY										
1	PC	19APC0515	Operating Systems	3	0	0	3	30	70	100
2	PC	19APC0521	Artificial Intelligence	3	0	0	3	30	70	100
3	PC	19APC0520	Compiler Design	3	0	0	3	30	70	100
4	PC	19APC0507	Software Engineering	2	0	0	2	30	70	100
5	OE	19APE0417 19AOE0303 19APC0428	Open Elective I Sensors and IoT Optimization Techniques Microprocessor and Interfacing	3	0	0	3	30	70	100
6	PE	19APE0501 19APE0502 19APE0503	Professional Elective I Data Warehousing and Mining Design Patterns Computer Graphics	3	0	0	3	30	70	100
7	MC	19AMC9904	Professional Ethics and Human Values	3	0	0	0	30	-	30
PRACTICAL										
8	PR	19APR0502	Socially Relevant Projects (15 Hrs/Semester)	0	0	0	0.5	50	-	50
9	PC	19APC0517	Operating System Lab	0	0	3	1.5	30	70	100
10	PC	19APC0522	Artificial Intelligence Lab	0	0	3	1.5	30	70	100
11	PC	19APC0508	Compiler Design Lab	0	0	2	1	30	70	100
						TOTAL	**21.5**	**350**	**630**	**980**

AITS TPT - CSE

ANNAMACHARYA INSTITUTE OF TECHNOLOGY AND SCIENCES, TIRUPATI
(AUTONOMOUS)

B.Tech III Year I Semester

COURSE CODE	COURSE TITLE	L	T	P	CREDITS
19APC0515	OPERATING SYSTEMS	3	0	0	3

Course Objectives:

- To make the students understand the basic operating system concepts such as processes, threads, scheduling, synchronization, deadlocks, memory management, file and I/O subsystems and protection.
- To get acquaintance with the class of abstractions afford by general purpose operating systems that aid the development of user applications.

Course Outcomes:

- Distinguish between the different types of operating system environments.
- Apply the concepts of process synchronization & CPU scheduling
- Develop solutions to deadlock and memory management
- Analyze various disk scheduling algorithms and file system interfaces
- Analyze the various security issues and goals of protection

UNIT - 1:

Operating Systems Overview: Operating system functions, Operating system structure, operating systems Operations, protection and security, Computing Environments, Open- Source Operating Systems

System Structures: Operating System Services, User and Operating-System Interface, systems calls, Types of System Calls, system programs, operating system structure, operating system debugging, System Boot.

Processes: Process concept, process Scheduling, Operations on processes, Inter process Communication, Examples of IPC systems.

UNIT - 2:

Threads: overview, Multi-core Programming, Multithreading Models, Thread Libraries, Implicit Threading, Threading Issues.

Process Synchronization: The critical-section problem, Peterson's Solution, Synchronization Hardware, Mutex Locks, Semaphores, Classic problems of synchronization, Monitors, Synchronization examples, Alternative approaches.

CPU Scheduling: Scheduling-Criteria, Scheduling Algorithms, Thread Scheduling, Multiple-Processor Scheduling, Real-Time CPU Scheduling, Algorithm Evaluation.

UNIT - 3:

Memory Management: Swapping, contiguous memory allocation, segmentation, paging, structure of the page table.

Virtual memory: demand paging, page-replacement, Allocation of frames, Thrashing, Memory-Mapped Files, Allocating Kernel Memory

Deadlocks: System Model, deadlock characterization, Methods of handling Deadlocks, Deadlock prevention, Detection and Avoidance, Recovery from deadlock.

UNIT - 4:

Mass-storage structure: Overview of Mass-storage structure, Disk structure, Disk attachment, Disk scheduling, Swap-space management, RAID structure, Stable-storage implementation.

File system Interface: The concept of a file, Access Methods, Directory and Disk structure, File system mounting, File sharing, Protection.

File system Implementation: File-system structure, File-system Implementation, Directory Implementation, Allocation Methods, Free-Space management.

UNIT - 5:

I/O systems: I/O Hardware, Application I/O interface, Kernel I/O subsystem, Transforming I/O requests to Hardware operations.

Protection: Goals of Protection, Principles of Protection, Domain of protection, Access Matrix, Implementation of Access Matrix, Access control, Revocation of Access Rights, Capability- Based systems, Language – Based Protection

Security: The Security problem, Program threats, System and Network threats, Cryptography as a security tool, User authentication, Implementing security defenses, Firewalling to protect systems and networks, Computer–security classifications.

Text Books:

1. Operating System Concepts, Abraham Silberchatz, Peter B. Galvin, Greg Gagne, Wiley, Eight Edition, 2018.

Reference Books:

1. Operating systems by A K Sharma, Universities Press,
2. Operating Systems, S.Haldar, A.A.Aravind, Pearson Education.
3. Operating Systems, A.S.Godbole, Second Edition, TMH.
4. An Introduction to Operating Systems, P.C.P. Bhatt, PHI.
5. Operating Systems, G.Nutt, N.Chaki and S.Neogy, Third Edition, Pearson Education.
6. Operating Systems, R.Elmasri, A,G.Carrick and D.Levine, Mc Graw Hill.
7. Modern Operating Systems, Andrew S Tanenbaum, Second Edition, PHI.

List of COs	PO no. and keyword	Competency Indicator	Performance Indicator
CO1	PO3. Design/development of solutions	3.1 & 3.3	3.1.6 & 3.3.1
CO2	PO3. Design/development of solutions	3.1	3.1.6
CO3	PO2. Problem Analysis	2.2	2.2.1 & 2.1.3
CO4	PO5. Modern tool usage	5.1	5.1.1
CO5	PO2. Problem Analysis	2.1, 2.2	2.1.3 & 2.2.1

AITS IPT - CSE

ANNAMACHARYA INSTITUTE OF TECHNOLOGY AND SCIENCES, TIRUPATI
(AUTONOMOUS)

B.Tech III Year I Semester

COURSE CODE	COURSE TITLE	L	T	P	CREDITS
19APC0521	ARTIFICIAL INTELLIGENCE	3	0	0	3

Course Objectives:

- Define Artificial Intelligence and establish the cultural background for study
- Understand various learning algorithms
- Explore the searching and optimization techniques for problem solving
- Provide basic knowledge on Natural Language Processing and Robotics

Course Outcomes:

- Apply searching techniques for solving a problem
- Design Intelligent Agents
- Develop Natural Language Interface for Machines
- Design mini robots
- Summarize past, present and future of Artificial Intelligence

UNIT I

Introduction: What is AI, Foundations of AI, History of AI, The State of Art.
Intelligent Agents: Agents and Environments, Good Behaviour: The Concept of Rationality, The Nature of Environments, The Structure of Agents.

UNIT II

Solving Problems by searching: Problem Solving Agents, Example problems, Searching for Solutions, Uninformed Search Strategies, Informed search strategies, Heuristic Functions, Beyond Classical Search: Local Search Algorithms and Optimization Problems, Local Search in Continues Spaces, Searching with Nondeterministic Actions, Searching with partial observations, online search agents and unknown environments.

UNIT III

Reinforcement Learning: Introduction, Passive Reinforcement Learning, Active Reinforcement Learning, Generalization in Reinforcement Learning, Policy Search, applications of RL
Natural Language Processing: Language Models, Text Classification, Information Retrieval, Information Extraction.

UNIT IV

Natural Language for Communication: Phrase structure grammars, Syntactic Analysis, Augmented Grammars and semantic Interpretation, Machine Translation, Speech Recognition
Perception: Image Formation, Early Image Processing Operations, Object Recognition by appearance, Reconstructing the 3D World, Object Recognition from Structural information, Using Vision.

UNIT V

Robotics: Introduction, Robot Hardware, Robotic Perception, Planning to move, planning uncertain movements, Moving, Robotic software architectures, application domains
Philosophical foundations: Weak AI, Strong AI, Ethics and Risks of AI, Agent Components, Agent Architectures, Are we going in the right direction, What if AI does succeed.

Textbook:

1. Stuart J. Russell, Peter Norvig, "Artificial Intelligence A Modern Approach", 3rd Edition, Pearson Education, 2019.

References:

1. Nilsson, Nils J., and Nils Johan Nilsson. Artificial intelligence: a new synthesis. Morgan Kaufmann, 1998.
2. Johnson, Benny G., Fred Phillips, and Linda G. Chase. "An intelligent tutoring system for the accounting cycle: Enhancing textbook homework with artificial intelligence." Journal of Accounting Education 27.1 (2009): 30-39.

AITS TPT - CSE

List of COs	PO no. and keyword	Competency Indicator	Performance Indicator
CO1	PO2. Problem Analysis	2.2	2.2.3
CO2	PO3. Design/development of solutions	3.3	3.3.1
CO3	PO5. Modern tool usage	5.1	5.1.1
CO4	PO3. Design/development of solutions	3.4	3.4.1
CO5	PO12. Life-long learning	12.3	12.3.2

ANNAMACHARYA INSTITUTE OF TECHNOLOGY AND SCIENCES, TIRUPATI
(AUTONOMOUS)

B.Tech III Year I Semester

COURSE CODE	COURSE TITLE	L	T	P	CREDITS
19APC0520	COMPILER DESIGN	3	0	0	3

Course Objectives:

This course is a de facto capstone course in Computer Science, as it combines skills in software design, programming, data structures and algorithms, theory of computing, documentation, and machine architecture to produce a functional compiler.

- Realize that computing science theory can be used as the basis for real applications
- Introduce the major concept areas of language translation and compiler design.
- Learn how a compiler works
- Know about the powerful compiler generation tools and techniques, which are useful to the other non-compiler applications
- Know the importance of optimization and learn how to write programs that execute faster

Course Outcomes

- Able to design a compiler for a simple programming language
- Able to use the tools related to compiler design effectively and efficiently
- Ability to write optimized code

Unit - I

Introduction: Language processors, The Structure of a Compiler, the science of building a complier.
Lexical Analysis: The Role of the lexical analyzer, Input buffering, Specification of tokens, Recognition of tokens, The lexical analyzer generator Lex, Design of a Lexical Analyzer generator

Unit II

Syntax Analysis: Introduction, Context Free Grammars, Writing a grammar, TOP Down Parsing, Bottom Up Parsing,
Introduction to LR Parsing: Simple LR, More Powerful LR Parsers, Using ambiguous grammars, Parser Generators

UNIT III

Syntax Directed Translation: Syntax Directed Definitions, Evaluation orders for SDD's, Application of SDT, SDT schemes, Implementing L-attribute SDD's.
Intermediate Code Generation: Variants of syntax trees, three address code, Types and declarations, Translations of expressions, Type checking, control flow statements, backpatching, switch statements, intermediate code for procedure.

UNIT IV

Run Time Environment : storage organization, , Stack allocation of space, Access to non-local data on stack , Heap management
Symbol Table: Introduction, symbol table entries, operations on the symbol table, symbol table organizations, non block structured language, block structured language.

UNIT V

Code Generation: Issues in the design of a code generator, The Target language, Basic blocks and flow graphs, optimization of basic blocks, a simple code generator, register allocation and assignment, optimal code generation for expressions, dynamic programming code generation.
Code Optimization: Introduction, where and how to optimize, principle source of optimization, function preserving transformations, loop optimizations, global flow analysis, machine dependent optimization

Text Books :

1. "Compilers Principles, Techniques and Tools", Second Edition, Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman., Pearson, 2016.
2. "Compiler Construction", K.V.N Sunitha, Pearson, 2013

Reference Books :

1. Compiler Design", K. Muneeswaran., Oxford University Press, 2012
2. "Engineering A Compiler", Second Edition, Keith D. Cooper & Linda Torczon., MK(Morga Kaufmann) (ELSEVIER)
3. "Compilers Principles and Practice", Parag H. Dave, Himanshu B. Dave.,PEARSON
4. "Compiler Design", SandeepSaxena, Rajkumar Singh Rathore., S.Chand publications
5. "Compiler Design", SantanuChattopadhyay., PHI
6. "Principals of Compiler Design", Nadhni Prasad, Elsevier

List of COs	PO no. and keyword	Competency Indicator	Performance Indicator
CO1	PO3. Design/development of solutions	3.3	3.3.1
CO2	PO5. Modern tool usage	5.1	5.1.2
CO3	PO4. Conduct investigations of complex problems	4.2	4.2.1

AITS TPT - CSE

ANNAMACHARYA INSTITUTE OF TECHNOLOGY AND SCIENCES, TIRUPATI
(AUTONOMOUS)

B.Tech III Year I Semester

COURSE CODE	COURSE TITLE	L	T	P	CREDITS
19APC0507	Software Engineering	2	0	0	2

Course Objectives:

A student should be able to develop a software product that

- should be feasible for the software to evolve to meet changing requirements.
- is correct if the different requirements as specified in the SRS document have been correctly implemented.
- has good reusability if the different modules of the product can easily be reused to develop new products.
- facilitates both the establishment of test criteria and the evaluation of the software with respect to those criteria.
- can be expected to perform its desired function, over an arbitrary time period.

Course Outcomes:

- Introduce SE and Models
- Discusses Techniques on SPM
- Focus on Requirements analysis and Specification
- Highlights some important facets of Software Design
- Testing Techniques and Quality Control Activities
- Discusses on Software Quality Assurance and Trends

Unit 1:

Introduction: Evolution, Software Development Projects, Exploratory style of Software Development, Emergence, Notable Changes in Software Development Practices, Computer Systems Engineering

Software Life Cycle Models: A few basic concepts, Waterfall Model and its extensions, RAD, Agile Development Models, Spiral Model, Comparison

Unit 2:

Software Project Management: SPM complexities, Responsibility of a software Development Manager, Project Planning, Metrics for Project Size Estimation, Project Estimation Techniques, Empirical Estimation Techniques, COCOMO, Halstead's Software Science, Staffing Level-Estimation, Scheduling, Organization and Team Structures, Risk Management, Software Configuration Management

Requirement Analysis and Specification: Requirements Gathering and Analysis, SRS, Formal System Specification, Axiomatic Specification, Algebraic Specification, Executable Specification and 4GL

Unit 3:

Software Design: Overview of the Design Process, Characterize good design, Cohesion and Coupling, Layered Arrangement of Modules, Approaches to Software Design

Function-oriented Software Design: Overview, Structured Analysis, Developing the DFD model of a system, Structured Design, Detailed Design and Review

User Interface Design: Characteristics, Basic Concepts, Types, Fundamentals of Component-based GUI Development, A UI Design Methodology

Unit 4:

Coding and Testing: Coding, Code Review, Software Documentation, Testing, Unit Testing, Black-Box Testing, White-box Testing, Debugging, Program Analysis Tools, Integration Testing, Testing Object-oriented Programs, System Testing, Issues associated with Testing

Software Reliability and Quality Management: Software Reliability, Statistical Testing, Software Quality, Software Quality Management System, ISO 9000, SEI Capability Maturity Model, Other Important Standards, Six Sigma

Unit 5:

CASE: Scope, Environment, Support, Characteristics, Towards 2G CASE Tools. Software Maintenance: Characteristics.

Software Reuse: What can be reused, Issues, A Reuse Approach, Reuse at Organization level.

Emerging Trends: Client-Server Software, Architectures, CORBA, COM, DCOM, SOA, SAAS

Text Books:

1. Fundamentals of Software Engineering, Rajib Mall, PHI Learning, 5th edition
2. Software Engineering: A Practitioner's Approach, R S Pressman, McGraw Hill Education, 7th edition

Reference Books:

1. Software Engineering, Ian Sommerville, Pearson Education, Tenth edition
2. Pankaj Jalote's Software Engineering: A Precise Approach, Wiley publications

List of COs	PO no. and keyword	Competency Indicator	Performance Indicator
CO1	PO1. Engineering knowledge	1.3	1.3.1
CO2	PO2. Problem Analysis	2.2	2.2.2
CO3	PO2. Problem Analysis	2.3	2.3.2
CO4	PO3. Design/development of solutions	3.3	3.3.2
CO5	PO4. Conduct investigations of complex problems	4.3	4.3.4
CO6	PO12. Life-long learning	12.2	12.2.2

AITS TPT CSE

ANNAMACHARYA INSTITUTE OF TECHNOLOGY AND SCIENCES, TIRUPATI (AUTONOMOUS)

B. Tech III Year I Semester **Branch: CSE**

COURSE CODE	COURSE TITLE	L	T	P	CREDITS
19APE0417	Sensors & Internet of Things	3	0	0	3

Course Outcomes:

Upon completion of the course students will be able to:

- Understand the characteristics of sensors and Transducers.
- Identify the different types of sensors and recent trends.
- Determine the Market perspective of IoT.
- Compare and Contrast the use of Devices, Gateways and Data Management in IoT.
- To design IoT applications using Arduino

UNIT I:

Sensors: Introduction to sensors- Principles, Classifications, Parameters- Characteristics, sensor Types- Resistive Potentiometer, Inductive sensor, Capacitive senor, Thermal sensors, Magnetic sensors, Radiation sensors, Smart sensors, Sensor Classification, criteria to choose a sensor, Generation of sensors, Transducers: Active & Passive Transducers- Measurement of Displacement(Resistance, Capacitance, Inductance, LVDT), Force (Strain Gauges), Pressure(Piezoelectric transducers).

UNIT II:

Recent Trends in Sensor Technologies: Film Sensors, Semiconductor IC Technology, MEMS, Nano Sensors.
Applications of sensors: Temperature sensors, Home appliance sensors, Medical diagnostic sensor, Sensors for Environmental Monitoring-Pollution Hazards, Sensing Environmental Pollution.

UNIT III:

IoT: Characteristics of IoT, Design principles of IoT, IoT Architecture and Protocols, Enabling Technologies for IoT, IoT levels and IoT vs M2M, A Use case example.
M2M to IoT - A Market Perspective- Introduction, Some Definitions, M2M Value Chains, IoT Value Chains, An emerging industrial structure for IoT.

UNIT IV:

M2M to IoT-An Architectural Overview- Building an architecture, Main design principles and needed capabilities, An IoT architecture outline, standards considerations.
M2M and IoT Technology Fundamentals- Devices and gateways, Local and wide area networking, Data management, Knowledge management.

UNIT V:

IoT Design Methodology: Design methodology, Challenges in IoT Design, IoT System Management. IoT Servers.
Basics of Arduino: Introduction to Arduino, Arduino IDE, Basic Commands for Arduino, Connecting LEDs with Arduino, Connecting LCD with Arduino.

Text Books:

1. D.Patranabis, “Sensors & Transducers”, PHI, 2nd ed., 2018.
2. Jan Holler, Vlasios Tsiatsis, Catherine Mulligan, Stefan Avesand, Stamatis Karnouskos, David Boyle, “From Machine-to-Machine to the Internet of Things: Introduction to a New Age of Intelligence”, 1st Edition, Academic Press, 2014. (ISBN-13:978-0124076846)
3. Rajesh Singh, Anita Gehlot, Lovi Raj Gupta, Bhupendra Singh, Mahendra Swain, Internet of Things with Raspberry Pi and Arduino, CRC Press, 2019.

Reference Books:

1. H.S.Kalsi, "Electronic Instrumentation", 2nd ed., TataMcGrawHill 2004.
2. A.K. Sawhney,- A course in Electrical & Electronic Measurement and Instrumentation, Dhanpat Rai and Company Private Limited, Reprint: 2014.

List of COs	PO no. and keyword	Competency Indicator	Performance Indicator
CO1	PO1. Engineering knowledge	1.3	1.3.1
CO2	PO1. Engineering knowledge	1.3	1.3.1
CO3	PO5. Modern tool usage	5.1	5.1.1
CO4	PO5. Modern tool usage	5.1	5.1.1
CO5	PO5. Modern tool usage	5.2	5.2.2

ATS IPT - CSE

ANNAMACHARYA INSTITUTE OF TECHNOLOGY AND SCIENCES, TIRUPATI
(AUTONOMOUS)

B.Tech III Year I Semester

COURSE CODE	COURSE TITLE	L	T	P	CREDITS
19AOE0303	OPTIMIZATION TECHNIQUES	3	0	0	3

Course Outcomes

- Explain the need of optimization of engineering systems
- Understand optimization of electrical and electronics engineering problems
- Apply classical optimization techniques, linear programming, simplex algorithm, transportation problem
- Apply unconstrained optimization and constrained non-linear programming and dynamic programming
- Formulate optimization problems.

UNIT – I

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

UNIT – II

Linear Programming: Standard form of a linear programming problem – geometry of linear programming problems – definitions and theorems – solution of a system of linear simultaneous equations – pivotal reduction of a general system of equations – motivation to the simplex method – simplex algorithm.
Transportation Problem: Finding initial basic feasible solution by north – west corner rule, least cost method and Vogel's approximation method – testing for optimality of balanced transportation problems.

UNIT – III

Unconstrained Nonlinear Programming: One dimensional minimization method, Classification, Fibonacci method and Quadratic interpolation method Unconstrained Optimization Techniques: Univariant method, Powell's method and steepest descent method.

UNIT – IV

Constrained Nonlinear Programming: Characteristics of a constrained problem - classification - Basic approach of Penalty Function method - Basic approach of Penalty Function method - Basic approaches of Interior and Exterior penalty function methods - Introduction to convex programming problem.

UNIT – V

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

TEXT BOOKS:

1. Singiresu S. Rao, Engineering Optimization: Theory and Practice by John Wiley and Sons, 4th edition, 2009.
2. H. S. Kasene & K. D. Kumar, Introductory Operations Research, Springer (India), Pvt. Ltd., 2004

REFERENCE BOOKS:

1. George Bernard Dantzig, Mukund Narain Thapa, "Linear programming", Springer series in operations research 3rd edition, 2003.
2. H.A. Taha, "Operations Research: An Introduction", 8th Edition, Pearson/Prentice Hall, 2007.
3. Kalyanmoy Deb, "Optimization for Engineering Design – Algorithms and Examples", PHI Learning Pvt. Ltd, New Delhi, 2005.

List of COs	PO no. and keyword	Competency Indicator	Performance Indicator
CO1	PO1. Engineering knowledge	1.4	1.4.1
CO2	PO2. Problem Analysis	2.1	2.1.3
CO3	PO4. Conduct investigations of Complex Problems	4.1	4.1.2
CO4	PO2. Problem Analysis	2.1	2.1.2
CO5	PO1. Engineering knowledge	1.4	1.4.1

AITS TPT - CSE

ANNAMACHARYA INSTITUTE OF TECHNOLOGY AND SCIENCES, TIRUPATI
(AUTONOMOUS)

B.Tech III Year I Semester

COURSE CODE	COURSE TITLE	L	T	P	CREDITS
19APC0428	MICROPROCESSORS AND INTERFACING	3	0	0	3

Course Outcomes:

After completion of this subject the students will be able to:

- Understand concepts of Intel x85 and 8086 series of processors
- Develop various programming using 8086 instruction set.
- Understand concepts of 8086 interrupts and Memory, I/O interfacing
- Understand concepts of Interfacing programmable devices for 8086
- Understand concepts of Intel 8051 series of microcontrollers

UNIT I

Microprocessors-Evolution and Introduction: Microprocessors and Micro Controllers, Microprocessor based system, Origin of Microprocessor, Classification of Microprocessors, Types of Memory, I/O Devices, Technology Improvements Adapted to Microprocessors and Computers, Introduction to 8085 processor, Architecture of 8085, Microprocessor instructions, classification of instructions, Instruction set of 8085.

Intel 8086 Microprocessor architecture, Features, and Signals: Architecture of 8086, Accessing memory locations, PIN details of 8086.

UNIT II

Addressing Modes, Instruction Set and Programming of 8086: Addressing modes in 8086, Instruction set of 8086, 8086 Assembly Language Programming, Modular Programming.

UNIT III

8086 Interrupts: Interrupt types in 8086, Processing of Interrupts by 8086, Dedicated interrupt types in 8086, Software interrupts-types 00H-FFH, Priority among 8086 interrupts, Interrupt service routines, BIOS interrupts or functional calls, Interrupt handlers, DOS services-INT 21H, System calls-BIOS services.

Memory and I/O Interfacing: Physical memory organization in 8086, Formation of system bus, Interfacing RAM and EPROM chips using only logic gates, Interfacing RAM/ EPROM chips using decoder IC and logic gates, I/O interfacing, Interfacing 8-bit input device with 8086, Interfacing output device using 8086, Interfacing printer with 8086, Interfacing 8-bit and 16-bit I/O devices or ports with 8086, Interfacing CRT terminal with 8086.

UNIT IV

Features and Interfacing of programmable devices for 8086 systems: Intel 8255 programmable peripheral interface, Interfacing switches and LEDS, Interfacing seven segment displays, Traffic light control, Interfacing analog to digital converters, Intel Timer IC 8253, Introduction to serial communication, 8259 programmable controller, 8237 DMA controller.

UNIT V

Introduction to 8051 Micro controllers: Intel's MCS-51 series micro controllers, Intel 8051 architecture, Memory organization, Internal RAM structure, Power control in 8051, Stack operation. 8051 Instruction Set and Programming: Introduction, Addressing modes of 8051, Instruction set of 8051, Hardware features of 8051: Introduction, Parallel ports in 8051, External memory interfacing in 8051, Timers, Interrupts, Serial ports.

Interfacing Examples: Interfacing 8255 with 8051, Interfacing of push button switches and LEDS, Interfacing of seven segment displays.

Text Books:

1. "Microprocessor and Interfacing 8086, 8051, 8096 and advanced processors", Senthil Kumar, Saravanan, Jeevanathan, Shah, 1st edition, 2nd impression, 2012, Oxford University Press.
2. "The X86 Microprocessors", Lyla B. Das. , 2010, Pearson.

Reference Books:

1. "Microprocessor and Interfacing: Programming and Hardware", Douglas V.Hall, McGrawHill
2. "8086 microprocessor: Programming and Interfacing the PC", Kenneth Ayala, Cengage Learning
3. "ARM system-on-chip architecture", Steve Furber, Addison-Wesley Professional
4. "The Intel Microprocessors", Barry B. Brey, Prentice Hall

List of COs	PO no. and keyword	Competency Indicator	Performance Indicator
CO1	PO2:Problem analysis	2.2	2.2.1
CO2	PO3: design/development solutions	2.3	2.3.1
CO3	PO4: Conduct Investigation of complex problems	4.2	4.2.2
CO4	PO4: Conduct Investigation of complex problems	4.3	4.3.2
CO5	PO5: conduct Modern tool usage	5.2	5.2.3

AITSPT - CSE

ANNAMACHARYA INSTITUTE OF TECHNOLOGY AND SCIENCES, TIRUPATI
(AUTONOMOUS)

B.Tech III Year I Semester

COURSE CODE	COURSE TITLE	L	T	P	CREDITS
19APE0501	DATA WAREHOUSING & MINING	3	0	0	3

Course Objectives:

- To know the basic concepts and principles of data warehousing and data mining
- Learn pre-processing techniques and data mining functionalities
- Learn and create multidimensional models for data warehousing
- Study and evaluate performance of Frequent Item sets and Association Rules
- Understand and Compare different types of classification and clustering algorithms

Course Outcomes:

- Understand the basic concepts of data warehouse and data Mining
- Apply pre-processing techniques for data cleansing
- Analyze and evaluate performance of algorithms for Association Rules
- Analyze Classification and Clustering algorithms

UNIT I

Introduction: Fundamentals of data mining, Data Mining Functionalities, Classification of Data Mining systems, Data Mining Task Primitives, Integration of a Data Mining System with a Database or a Data Warehouse System, Major issues in Data Mining. **Data Preprocessing:** Need for Preprocessing the Data, Data Cleaning, Data Integration and Transformation, Data Reduction, Discretization and Concept Hierarchy Generation.

UNIT II

Data Warehouse and OLAP Technology for Data Mining: Data Warehouse, Multidimensional Data Model, Data Warehouse Architecture, Data Warehouse Implementation, Further Development of Data Cube Technology, From Data Warehousing to Data Mining. **Data Cube Computation and Data Generalization:** Efficient Methods for Data Cube Computation, Further Development of Data Cube and OLAP Technology, Attribute-Oriented Induction.

UNIT III

Mining Frequent Patterns, Associations and Correlations: Basic Concepts, Efficient and Scalable Frequent Itemset Mining Methods, Mining various kinds of Association Rules, From Association Mining to Correlation Analysis, Constraint-Based Association Mining, **Classification and Prediction:** Issues Regarding Classification and Prediction, Classification by Decision Tree Induction, Bayesian Classification, Rule-Based Classification, Classification by Back propagation, Support Vector Machines, Associative Classification, Lazy Learners, Other Classification Methods, Prediction, Accuracy and Error measures, Evaluating the accuracy of a Classifier or a Predictor, Ensemble Methods

UNIT IV

Cluster Analysis: Introduction, Types of Data in Cluster Analysis, A Categorization of Major Clustering Methods, Partitioning Methods, Hierarchical Methods, Density-Based Methods, Grid-Based Methods, Model-Based Clustering Methods, Clustering High-Dimensional Data, Constraint-Based Cluster Analysis, Outlier Analysis.

UNIT V

Mining Streams, Time Series and Sequence Data: Mining Data Streams, Mining Time- Series Data, Mining Sequence Patterns in Transactional Databases, Mining Sequence Patterns in Biological Data, Graph Mining, Social Network Analysis and Multi relational Data Mining, **Mining Object, Spatial, Multimedia, Text and Web Data**: Multidimensional Analysis and Descriptive Mining of Complex Data Objects, Spatial Data Mining, Multimedia Data Mining, Text Mining, Mining the World Wide Web.

TEXT BOOKS:

1. Data Mining: Concepts and Techniques, Jiawei Han and Micheline Kamber, Morgan Kaufmann Publishers, Elsevier, Second Edition, 2012.
2. Introduction to Data Mining – Pang-Ning Tan, Michael Steinbach and Vipin Kumar, Pearson Education.

REFERENCES:

1. Data Mining Techniques, Arun KPujari, Second Edition, Universities Press.
2. Data Warehousing in the Real World, Sam Aanhory & Dennis Murray Pearson EdnAsia.
3. Insight into Data Mining, K.P.Soman, S.Diwakar, V.Ajay, PHI,2008.

List of COs	PO no. and keyword	Competency Indicator	Performance Indicator
CO1	PO2:Problem Analysis	2.2	2.2.3
CO2	PO3:Design/development of Solutions	3.3	3.3.2
CO3	PO4:Conduct Investigations of Complex Problems	4.3	4.3.2
CO4	PO2:Problem Analysis	2.2	2.2.3

AITS TPT - CSE

ANNAMACHARYA INSTITUTE OF TECHNOLOGY AND SCIENCES, TIRUPATI
(AUTONOMOUS)

B.Tech III Year I Semester

COURSE CODE	COURSE TITLE	L	T	P	CREDITS
19APE0502	DESIGN PATTERNS	3	0	0	3

Course Objectives:
- To understand design patterns and their underlying object oriented concepts.
- To understand implementation of design patterns and providing solutions to real world software design problems.
- To understand patterns with each other and understanding the consequences of combining patterns on the overall quality of a system.

Course Outcomes:
- Know the underlying object oriented principles of design patterns.
- Understand the context in which the pattern can be applied.
- Understand how the application of a pattern affects the system quality and its tradeoffs.

UNIT-I
Introduction to Design Patterns: Design Pattern Definition, Design Patterns in Small Talk MVC, Describing Design Patterns, Catalog of Design Patterns, Organizing the Catalog, Solving of Design Problems using Design Patterns, Selection of a Design Pattern, Use of Design Patterns.

UNIT-II
Designing A Document Editor: A Case Study Design problems, Document structure, Formatting, Embellishing the User Interface, Supporting Multiple Look and Feel standards, Supporting Multiple Window Systems, User Operations, Spelling Checking and Hyphenation.
Creational Patterns: Abstract Factory, Builder, Factory Method, Prototype, Singleton, Discussion of Creational Patterns.

UNIT-III
Structural Patterns-1: Adapter, Bridge, Composite.
Structural Patterns-2: Decorator, Façade, Flyweight, Proxy, Discuss of Structural Patterns.

UNIT-IV
Behavioral Patterns-1: Chain of Responsibility, Command, Interpreter, Iterator.
Behavioral Patterns-2: Mediator, Memento, Observer.

UNIT-V
Behavioral Patterns-2(cont'd): State, Strategy, Template Method, Visitor, Discussion of Behavioral Patterns.
What to Expect from Design Patterns, A Brief History, The Pattern Community An Invitation, A Parting Thought.

TEXT BOOK :
1. Design Patterns By Erich Gamma, Pearson Education

REFERENCE BOOKS:

1. Pattern's in JAVA Vol-I By Mark Grand, Wiley DreamTech.
2. Pattern's in JAVA Vol-II By Mark Grand, Wiley DreamTech.
3. JAVA Enterprise Design Patterns Vol-III By Mark Grand, Wiley DreamTech.
4. Head First Design Patterns By Eric Freeman-Oreilly - spd
5. Design Patterns Explained By Alan Shalloway, Pearson Education.
6. Pattern Oriented Software Architecture, F.Buschmann & others, John Wiley & Sons.

List of COs	PO no. and keyword	Competency Indicator	Performance Indicator
CO1	PO1. Engineering knowledge	1.1	1.1.1
CO2	PO2. Problem Analysis	2.2	2.2.3
CO3	PO3. Design/development of solutions	3.3	3.3.1

AITS TPT - CSE

ANNAMACHARYA INSTITUTE OF TECHNOLOGY AND SCIENCES, TIRUPATI
(AUTONOMOUS)

B.Tech III Year I Semester

COURSE CODE	COURSE TITLE	L	T	P	CREDITS
19APE0503	COMPUTER GRAPHICS	3	0	0	3

Course Objectives:
This course is designed to:

- Introduce the use of the components of a graphics system and become familiar with the building approach of graphics system components and related algorithms.
- Understand the basic principles of 3- 3-dimensional computer graphics.
- Provide insights on how to scan, convert the basic geometrical primitives, how to transform the shapes to fit them as per the picture definition.
- Provide an understanding of mapping from world coordinates to device coordinates, clipping, and projections.
- Discuss the application of computer graphics concepts in the development of computer games, information visualization, and business applications.

Course outcomes:
Upon completion of the course, the students should be able to:

- Explain the basic concepts used in computer graphics.
- Inspect various algorithms to scan, convert the basic geometrical primitives, transformations, Area filling, clipping.
- Assess the importance of viewing and projections.
- Define the fundamentals of animation, virtual reality and its related technologies.
- Analyze the typical graphics pipeline

UNIT I
OVERVIEW OF COMPUTER GRAPHICS SYSTEM
OverView of Computer Graphics System – Video display devices – Raster Scan and randomscan system – Input devices – Hard copy devices.

UNIT II
OUTPUT PRIMITIVES AND ATTRIBUTES
Drawing line, circle and ellipse generating algorithms – Scan line algorithm – CharacterGeneration – attributes of lines, curves and characters – Antialiasing.

UNIT III
TWO DIMENSIONAL GRAPHICS TRANSFORMATIONS AND VIEWING:
Two-dimensional Geometric Transformations – Windowing and Clipping – Clipping of lines andclipping of polygons.

UNIT IV
THREE DIMENSIONAL GRAPHICS AND VIEWING
Three-dimensional concepts – Object representations- Polygon table, Quadric surfaces, Splines, Bezier curves and surfaces – Geometric and Modelling transformations – Viewing -Parallel and perspective projections.

UNIT V
REMOVAL OF HIDDEN SURFACES
Visible Surface Detection Methods – Computer Animation.

TEXTBOOK
1. Hearn, D. and Pauline Baker,M., Computer Graphics (C-Version), 2nd Edition, Pearson Education, 2002.

REFERENCES
1. Neuman, W.M., and Sproull, R.F., Principles of Interactive Computer Graphics, Mc Graw Hill Book Co., 1979.
2. Roger, D.F., Procedural elements for Computer Graphics, Mc Graw Hill Book Co., 1985.
3. Asthana, R.G.S and Sinha, N.K., Computer Graphics, New Age Int. Pub. (P) Ltd., 1996.
4. Floey, J.D., Van Dam, A, Feiner, S.K. and Hughes, J.F, Computer Graphics, Pearson Education, 2001.

List of COs	PO no. and keyword	Competency Indicator	Performance Indicator
CO1	PO1. Engineering knowledge	1.3	1.3.1
CO2	PO4. Conduct investigations of complex problems	4.1	4.1.1
CO3	PO4. Conduct investigations of complex problems	4.3	4.3.1
CO4	PO1. Engineering knowledge	1.3	1.3.1
CO5	PO2. Problem Analysis	2.1	2.1.1

ATS TPT - CSE

ANNAMACHARYA INSTITUTE OF TECHNOLOGY AND SCIENCES, TIRUPATI
(AUTONOMOUS)

B.Tech III Year I Semester

COURSE CODE	COURSE TITLE	L	T	P	CREDITS
19AMC9904	Professional Ethics And Human Values	3	0	0	0

Course Outcomes:

1. It ensures students sustained happiness through identifying the essentials of human values and skills.
2. The students will understand the importance of Values and Ethics in their personal lives and professional careers.
3. The students will learn the rights and responsibilities as an employee, team member and a global citizen.
4. Students understand practically the importance of trust, mutually satisfying human behavior and enriching interaction with nature.
5. Students can able to develop appropriate technologies and management patterns to create harmony in professional and personal life.

UNIT - I: **12hrs**

Introduction to Human Values: Need, basic Guidelines, Content and Process for Value Education, Self Exploration – 'Natural Acceptance' and Experiential Validation. Continuous Happiness and Prosperity - A look at basic Human Aspirations. Right understanding, Relationship and Physical Facilities. Understanding Happiness and Prosperity correctly.

UNIT - II: **12hrs**

Understanding Harmony in the Family and Society: Harmony in Human - Human Relationship: Understanding harmony in the Family the basic unit of human interaction. Understanding values in human - human relationship; meaning of Nyaya and program for its fulfillment to ensure Ubhay-tripti; Trust (Vishwas) and Respect (Samman) as the foundational values of relationship. Understanding the harmony in the society (society being an extension of family). Visualizing a universal harmonious order in society - Undivided Society (Akhand Samaj), Universal Order (Sarvabhaum Vyawastha) - from family to world family!

UNIT – III: **12hrs**

Introduction to Professional Ethics: Basic Concepts, Governing Ethics, Personal & Professional Ethics, Ethical Dilemmas, Life Skills, Emotional Intelligence, Thoughts of Ethics, Value Education, Dimensions of Ethics, Profession and professionalism, Professional Associations, Professional Risks, Professional Accountabilities, Professional Success, Ethics and Profession.

UNIT – IV: **15hrs**

Professional Practices in Engineering: Work Place Rights & Responsibilities, Professions and Norms of Professional Conduct, Norms of Professional Conduct vs. Profession; Responsibilities, Obligations and Moral Values in Professional Ethics, Professional codes of ethics, the limits of predictability and responsibilities of the engineering profession. Central Responsibilities of Engineers – The Centrality of Responsibilities of Professional Ethics; lessons from 1979 American Airlines DC-10 Crash and Kansas City Hyatt Regency Walk away Collapse.

UNIT – V: **12hrs**

Global issues in Professional Ethics: Introduction – Current Scenario, Technology Globalization of MNCs, International Trade, World Summits, Issues, Business Ethics and Corporate Governance, Sustainable Development Ecosystem, Energy Concerns, Ozone Depletion, Pollution, Ethics in Manufacturing and Marketing, Media Ethics, War Ethics, Bio Ethics, Intellectual Property Rights.

Text Books:

1. R. R. Gaur, R Sangal, G P Bagaria, 2009, A Foundation Course in Human Values and Professional Ethics.
2. Professional Ethics: R. Subramanian, Oxford University Press, 2015. 3.Ethics in Engineering Practice & Research, Caroline Whitbeck, 2e, Cambridge University Press 2015.

Reference Books:

1. Prof. K. V. Subba Raju, 2013, Success Secrets for Engineering Students, Smart Student Publications, 3rd Edition.
2. Ivan IIlich, 1974, Energy & Equity, The Trinity Press, Worcester, and HarperCollins, USA
3. Engineering Ethics, Concepts Cases: Charles E Harris Jr., Michael S Pritcha Michael J Rabins, 4e ,Cengage learning, 2015.
4. Business Ethics concepts & Cases: Manuel G Velasquez, 6e, PHI, 2008.

List of COs	PO no. and keyword	Competency Indicator	Performance Indicator
CO1	PO8. Ethics	8.1	8.1.1
CO2	PO8. Ethics	8.1	8.1.1
CO3	PO8. Ethics	8.1	8.1.1
CO4	PO8. Ethics	8.1	8.1.1
CO5	PO8. Ethics	8.1	8.1.1

AITS IPT - CSE

ANNAMACHARYA INSTITUTE OF TECHNOLOGY AND SCIENCES, TIRUPATI
(AUTONOMOUS)

B.Tech III Year I Semester

COURSE CODE	COURSE TITLE	L	T	P	CREDITS
19APC0517	OPERATING SYSTEMS LABORATORY	0	0	3	1.5

Course Objectives:

- To understand the design aspects of operating system
- To solve various synchronization problems

Course outcomes:

- Ensure the development of applied skills in operating systems related areas.
- Able to write software routines modules or implementing various concepts of operating system.

List of Experiments to be implemented in C/Java

1. Practicing of Basic UNIX Commands.
2. Write programs using the following UNIX operating system calls Fork, exec, getpid, exit, wait, close, stat, opendir and readdir
3. Simulate UNIX commands like cp, ls, grep, etc.,
4. Simulate the following CPU scheduling algorithms: a) Round Robin b) SJF c) FCFS d) Priority
5. Simulate all file allocation strategies: a) Sequential b) Indexed c) Linked
6. Simulate MVT and MFT
7. Simulate all File Organization Techniques a) Single level directory b) Two level c) Hierarchical d) DAG
8. Simulate Bankers Algorithm for Deadlock Avoidance
9. Simulate Bankers Algorithm for Deadlock Prevention
10. Simulate all page replacement algorithms a) FIFO b) LRU c) LFU Etc. …
11. Simulate Paging Technique of memory management
12. Control the number of ports opened by the operating system with a) Semaphore b) monitors
13. Simulate how parent and child processes use shared memory and address space
14. Simulate sleeping barber problem
15. Simulate dining philosopher's problem
16. Simulate producer and consumer problem using threads (use java)
17. Simulate little's formula to predict next burst time of a process for SJF scheduling algorithm.
18. Develop a code to detect a cycle in wait-for graph
19. Develop a code to convert virtual address to physical address
20. Simulate how operating system allocates frame to process
21. Simulate the prediction of deadlock in operating system when all the processes announce their resource requirement in advance.

Reference Books:

1. "Operating System Concepts", Abraham Silberchatz, Peter B. Galvin, Greg Gagne, Eighth edition, John Wiley.
2. "Operating Systems: Internals and Design Principles", Stallings, Sixth Edition– 2009,Pearson Education
3. "Modern Operating Systems", Andrew S Tanenbaum, Second Edition, PHI.
4. "Operating Systems", S.Haldar, A.A.Aravind, Pearson Education.
5. "Principles of Operating Systems", B.L.Stuart, Cengage learning, India Edition.2013-2014
6. "Operating Systems", A.S.Godbole, Second Edition, TMH.
7. "An Introduction to Operating Systems", P.C.P. Bhatt, PHI.

List of COs	PO no. and keyword	Competency Indicator	Performance Indicator
CO1	PO3. Design/development of solutions	3.3	3.3.1
CO2	PO5. Modern tool usage	5.1	5.1.1

AITSPT - CSE

ANNAMACHARYA INSTITUTE OF TECHNOLOGY AND SCIENCES, TIRUPATI
(AUTONOMOUS)

B.Tech III Year I Semester

COURSE CODE	COURSE TITLE	L	T	P	CREDITS
19APC0522	ARTIFICIAL INTELLIGENCE LABORATORY	0	0	3	1.5

Course Objectives:

This course is designed to:

- Explore the methods of implementing algorithms using artificial intelligence techniques
- Illustrate search algorithms
- Demonstrate building of intelligent agents

Course Outcomes:

Up on the completion of Course, The student will be able to

- Implement search algorithms
- Solve Artificial Intelligence Problems
- Design Chatbot
- Implement Text Classification

List of Experiments to be implemented in Java/Python

1. Write a Program to Implement BFS and DFS.
2. Write a Program to find the solution for travelling sales person problem.
3. Write a program to implement simulated annealing Algorithm.
4. Write a Program to Implement Tic-Tac-Toe game.
5. Write a Program to Implement 8-Puzzle problem.
6. Write a program to implement Towers of Hanoi problem.
7. Write a program to implement A* Algorithm.
8. Write a Program to Implement Water-Jug problem.
9. Write a program to implement Hangman game.
10. Write a program to solve N Queen problem using backtracking.
11. Generate Calendar for the given month and year using a python program.
12. Write a program to implement simple Chatbot.
13. Write a program to remove stop words for a given passage from a text file using NLTK.
14. Write a program to implement stemming for a given sentence using NLTK.
15. Write a program to POS (Parts of Speech) tagging for the give sentence using NLTK.
16. Write a program to implement Lemmatization using NLTK.

References:

1	Tensorflow: https://www.tensorflow.org/
2	Pytorch: https://pytorch.org/, https://github.com/pytorch
3	Theano: http://deeplearning.net/software/theano/https://github.com/Theano/Theano
4	https://www.nltk.org/

List of COs	PO no. and keyword	Competency Indicator	Performance Indicator
CO1	PO3. Design/development of solutions	3.2	3.2.1
CO2	PO4. Conduct investigations of complex problems	4.2	4.2.1
CO3	PO5. Modern tool usage	5.2	5.2.1
CO4	PO5. Modern tool usage	5.1	5.1.2

ATS TPT - CSE

ANNAMACHARYA INSTITUTE OF TECHNOLOGY AND SCIENCES, TIRUPATI
(AUTONOMOUS)

B.Tech III Year I Semester

COURSE CODE	COURSE TITLE	L	T	P	CREDITS
19APC0508	COMPILER DESIGN LAB	0	0	2	1

Course Objectives:

- To implement some of the functionality of the compiler
- To do programming using compiler related tools

Course Outcomes:

- Develop compiler tools
- Design simple compiler

List of Experiments

1. Design a lexical analyzer for given language and the lexical analyzer should ignore redundant spaces, tabs and new lines. It should also ignore comments. Although the syntax specification states that identifiers can be arbitrarily long, you may restrict the length to some reasonable value. Simulate the same in C language.
2. Write a C program to identify whether a given line is a comment or not.
3. Write a C program to recognize strings under 'a', 'a*b+', 'abb'.
4. Write a C program to test whether a given identifier is valid or not.
5. Write a C program to simulate lexical analyzer for validating operators.
6. Implement the lexical analyzer using JLex, flex or other lexical analyzer generating tools.
7. Write a C program for implementing the functionalities of predictive parser for the mini language specified in Note 1.
8. a) Write a C program for constructing of LL (1) parsing.
   b) Write a C program for constructing recursive descent parsing.
9. Write a C program to implement LALR parsing.
10. a) Write a C program to implement operator precedence parsing.
    b) Write a C program to implement Program semantic rules to calculate the expression that takes an expression with digits, + and * and computes the value.
11. Convert the BNF rules into Yacc form and write code to generate abstract syntax tree for the mini language specified in Note 1.
12. Write a C program to generate machine code from abstract syntax tree generated by the parser. The instruction set specified in Note 2 may be considered as the target code.

Note 1:

A simple language written in this language is

```
{int a[3],t1,t2;
T1=2;
A[0]=1;a[1]=2;a[t]=3;
T2=-( a[2]+t1*6)/(a[2]-t1);
If t2>5then
Print(t2)
Else{
Int t3;
T3=99;
T2=25;
Print(-t1+t2*t3);/*this is a comment on 2 lines*/
}endif
}
```

Comments(zero or more characters enclosed between the standard C/JAVA Style comment brackets/*…*/)can be inserted .The language has rudimentary support for1-dimenstional array,the declaration int a[3] declares an array of three elements,referenced as a[0],a[1] and a[2].Note also you should worry about the scopping of names.

Note 2:

Consider the following mini language, a simple procedural high –level language, only operating on integer data, with a syntax looking vaguely like a simple C crossed with pascal. The syntax of the language is defined by the following grammar.

```
<program>::=<block>
<block>::={<variable definition><slist>}
|{<slist>}
<variabledefinition>::=int <vardeflist>
<vardec>::=<identifier>|<identifier>[<constant>]
<slist>::=<statement>|<statement>;<slist>
<statement>::=<assignment>|<ifstament>|<whilestatement>
|<block>|<printstament>|<empty>
<assignment>::=<identifier>=<expression>
|<identifier>[<expression>]=<expression>
<if statement>::=if<bexpression>then<slist>else<slist>endif
|if<bexpression>then<slisi>endif
<whilestatement>::=while<bexpreession>do<slisi>enddo
<printstatement>:;=print(<expression>)
<expression>::=<expression>::=<expression><addingop><term>|<term>|<addingop>
<term>
<bexprssion>::=<expression><relop><expression>
<relop>::=<|<=|==|>=|>|!=
<addingop>::=+|-
<term>::=<term><multop><factor>|<factor>
<Multop>::=*|/
<factor>::=<constant>|<identifier>|<identifier>[<expression>]
|(<expression>)
```

ATS TPT - CSE

<constant>::=<digit>|<digit><constant>
<identifier>::=<identifier><letter or digit>|<letter>
<letter or digit>::=<letter>|<digit>
<letter>:;=a|b|c|d|e|f|g|h|I|j|k|l|m|n|o|p|q|r|s|t|u|v|w|x|y|z
<digit>::=0|1|2|3|4|5|^|7|8|9
<empty>::=has the obvious meaning

List of COs	**PO no. and keyword**	**Competency Indicator**	**Performance Indicator**
CO1	PO3. Design/development of solutions	3.2	3.2.1
CO2	PO3. Design/development of solutions	3.3	3.3.1

AITS TPT - CSE