# Savitribai Phule Pune University

*(Formerly University of Pune)*

**Three Year B.Sc. Degree Program in Computer Science**

**(Faculty of Science & Technology)**

# S.Y.B.Sc. (Computer Science)

Choice Based Credit System Syllabus
To be implemented from Academic Year
2020-2021

# S. Y. B. Sc.( Computer Science)

## Semester III

*(Total credits=22)*

| Course type | Paper Code | Paper title | Credits | Evaluation | | |
|---|---|---|---|---|---|---|
| | | | | CA | UE | TOTAL |
| CC-VIII | CS 231 | Data Structures and Algorithms – I | 2 | 15 | 35 | 50 |
| | CS 232 | Software Engineering | 2 | 15 | 35 | 50 |
| | CS 233 | Practical course on CS 231 and CS 232 | 2 | 15 | 35 | 50 |
| CC-IX | | Mathematics - I | 2 | 15 | 35 | 50 |
| | | Mathematics - II | 2 | 15 | 35 | 50 |
| | | Practical course in Mathematics | 2 | 15 | 35 | 50 |
| CC-X | | Electronics - I | 2 | 15 | 35 | 50 |
| | | Electronics - II | 2 | 15 | 35 | 50 |
| | | Practical course in Electronics | 2 | 15 | 35 | 50 |
| AECC-I | | Environment Science – I | 2 | | | |
| AECC-II | | Language Communication – I | 2 | | | |

# Semester IV

***(Total credits=22)***

| Course type | Paper Code | Paper title | Credits | Evaluation | | |
|---|---|---|---|---|---|---|
| | | | | CA | UE | TOTAL |
| CC-XI | CS 241 | Data Structures and Algorithms – II | 2 | 15 | 35 | 50 |
| | CS 242 | Computer Networks - I | 2 | 15 | 35 | 50 |
| | CS 243 | Practical course on CS 241 and CS 242 | 2 | 15 | 35 | 50 |
| CC-XII | | Mathematics - I | 2 | 15 | 35 | 50 |
| | | Mathematics - II | 2 | 15 | 35 | 50 |
| | | Practical course in Mathematics | 2 | 15 | 35 | 50 |
| CC-XIII | | Electronics - I | 2 | 15 | 35 | 50 |
| | | Electronics - II | 2 | 15 | 35 | 50 |
| | | Practical course in Electronics | 2 | 15 | 35 | 50 |
| AECC-I | | Environment Science – II | 2 | | | |
| AECC-II | | Language Communication –II | 2 | | | |

- Each theory Lecture time for S.Y. B.Sc Computer Science is of 50 min (3 lectures/ week for 2 credit course)
- Each practical session time for S.Y. B.Sc Computer Science is of 4 hrs 20 minutes (260 min)
- Practical batch size =12

**Savitribai Phule Pune University**
**S.Y.B.Sc. (Computer Science)**
**Computer Science Paper - I**
**Course Code: CS 231**
**Title : Data Structures and Algorithms – I**

| Teaching Scheme | No. of Credits | Examination Scheme |
|---|---|---|
| 3 Lectures / week (50 mins duration) | 2 | IE : 15 marks<br>UE: 35 marks |

**Prerequisites**:
Basic knowledge of algorithms and problem solving
Knowledge of C Programming Language

**Course Objectives**
1. To learn the systematic way of solving problem
2. To understand the different methods of organizing large amount of data
3. To efficiently implement the different data structures
4. To efficiently implement solutions for specific problems
5. To apply linear data structures.

**Course Outcomes:** On completion of the course, student will be able to
1. To use well-organized data structures in solving various problems.
2. To differentiate the usage of various structures in problem solution.
3. Implementing algorithms to solve problems using appropriate data structures.

**Course Contents**

| **Chapter 1** | **Introduction to Data Structures and Algorithm Analysis** | **4 lectures** |
|---|---|---|

1.1 Introduction
- 1.1.1 Need of Data Structure
- 1.1.2 Definitions - Data and information, Data type, Data object, ADT, Data Structure
- 1.1.3 Types of Data Structures

1.2 Algorithm analysis
1.2.1 Space and time complexity, Graphical understanding of the relation between different functions of n, examples of linear loop, logarithmic,quadratic loop etc.
1.2.2 Best, Worst, Average case analysis, Asymptotic notations (Big O, Omega Ω, Theta θ ), Problems on time complexity calculation.

| **Chapter 2** | **Array as a Data Structure** | **10 lectures** |
|---|---|---|

2.1 ADT of array, Operations
2.2Array applications - Searching
2.2.1 Sequential search, variations - Sentinel search, Probability search, ordered list search
- 2.2.2 Binary Search
- 2.2.3 Comparison of searching methods

2.3 Sorting Terminology- Internal, External, Stable, In-place Sorting
- 2.3.1 Comparison Based Sorting - Lower bound on comparison based sorting, Methods- Bubble Sort, Insertion Sort, Selection Sort, Algorithm design strategies - Divide and Conquer strategy, Merge Sort, Quick Sort, complexity analysis of sorting methods.

2.3.2 Non Comparison Based Sorting: Counting Sort, Radix Sort, complexity analysis.
2.3.3 Comparison of sorting methods

| **Chapter 3** | **Linked List** | **10 lectures** |
|---|---|---|

3.1 List as a Data Structure, differences with array.
3.2 Dynamic implementation of Linked List, internal and external pointers
3.3 Types of Linked List – Singly, Doubly, Circular
3.4 Operations on Linked List - create, traverse, insert, delete, search, sort, reverse, concatenate, merge, time complexity of operations.
3.5 Applications of Linked List – polynomial representation, Addition of two polynomials
3.6 Generalized linked list – concept, representation, multiple-variable polynomial representation using generalized list.

| **Chapter 4** | **Stack** | **6 lectures** |
|---|---|---|

4.1 Introduction
4.2 Operations – init(), push(), pop(), isEmpty(), isFull(), peek(), time complexity of operations.
4.3 Implementation- Static and Dynamic with comparison
4.4 Applications of stack
4.4.1 Function call and recursion, String reversal, palindrome checking
4.4.2 Expression types - infix, prefix and postfix, expression conversion and evaluation (implementation of infix to postfix, evaluation of postfix)
4.4.3Backtracking strategy - 4 queens problem (implementation using stack)

| **Chapter 5** | **Queue** | **6 lectures** |
|---|---|---|

5.1 Introduction
5.2 Operations - init(), enqueue(), dequeue(), isEmpty(), isFull(), peek(),time complexity of operations, differences with stack.
5.3 Implementation - Static and Dynamic with comparison
5.4 Types of Queue - Linear Queue, Circular Queue, Priority Queue, Double Ended Queue (with implementation)
5.5 Applications – CPU Scheduling in multiprogramming environment, Round robin algorithm

**Reference Books:**

1. Classic Data Structures-D. Samanta, Prentice Hall India Pvt. Ltd.
2. Fundamentals of Data Structures in C- Ellis Horowitz, SartajSahni,Susan Anderson-Freed, 2nd Edition, Universities Press.
3. Data Structures using C and C++-YedidyahLangsam, Moshe J. Augenstein, Aaron M. Tenenbaum, Pearson Education
4. Data Structures: A Pseudo code approach with C, Richard Gilberg ,Behrouz A. Forouzan, Cengage Learning.
5. Introduction to Data Structures in C-Ashok Kamthane, Pearson Education
6. Algorithms and Data Structures, Niklaus Wirth, Pearson Education

**Savitribai Phule Pune University**
**S.Y.B.Sc. (Computer Science)**
**Computer Science Paper -II**
**Course Code: CS 232**
**Title : Software Engineering**

| Teaching Scheme | No. of Credits | Examination Scheme |
|---|---|---|
| 3 lectures / week (50 mins duration) | **2** | IE : 15 marks<br>UE: 35 marks |

**Prerequisites**
ER Modeling

**Course Objectives**

1. To get knowledge and understanding of software engineering discipline.
2. To learn analysis and design principles for software project development.

**Course Outcomes**
On completion of the course, student will be able to-

1. Compare and chose a process model for a software project development.
2. Identify requirements analyze and prepare models.
3. Prepare the SRS, Design document, Project plan of a given software system.

**Course Contents**

| **Chapter 1** | **Title : Introduction To Software Engineering and Process Models** | **8 lectures** |
|---|---|---|

1.1 Definition of Software
1.2 Nature of Software Engineering
1.3 Changing nature of software
1.4 Software Process
- 1.4.1 The Process Framework
- 1.4.2 Umbrella Activities
- 1.4.3 Process Adaptation

1.5 Generic Process Model
1.6 Prescriptive Process Models
- 1.6.1 The Waterfall Model
- 1.6.2 Incremental Process Models
- 1.6.3 Evolutionary Process Models
- 1.6.4 Concurrent Models
- 1.6.5 The Unified Process

| **Chapter 2** | **Title : Agile Development** | **5lectures** |
|---|---|---|

2.1 What is Agility?
2.2 Agile Process
- 2.2.1 Agility Principles
- 2.2.2 The Politics Of Agile Development
- 2.2.3 Human Factors

2.3 Extreme Programming(XP)
- 2.3.1XP Values
- 2.3.2XP Process
- 2.3.3 Industrial XP

2.4 Adaptive Software Development(ASD)
2.5 Scrum
2.6 Dynamic System Development Model (DSDM)
2.7 Agile Unified Process (AUP)

| **Chapter 3** | **Title : Requirements Analysis** | **7 lectures** |
|---|---|---|

3.1 Requirement Elicitation,
3.2 Software requirement specification (SRS)
3.2.1 Developing Use Cases (UML)
3.3 Building the Analysis Model
3.3.1 Elements of the Analysis Model
3.3.2 Analysis Patterns
3.3.3 Agile Requirements Engineering
3.4 Negotiating Requirements
3.5 Validating Requirements

| **Chapter 4** | **Title : Requirements Modeling** | **10 lectures** |
|---|---|---|

4.1 Introduction to UML
4.2Structural Modeling
4.2.1 Use case model
4.2.2Class model
4.3Behavioral Modeling
4.3.1 Sequence model
4.3.2 Activity model
4.3.3 Communication or Collaboration model
4.4 Architectural Modeling
4.4.1 Component model
4.4.2 Artifact model
4.4.3 Deployment model

| **Chapter 5** | **Title :Design Concepts** | **6lectures** |
|---|---|---|

5.1 Design Process
5.1.1 Software Quality Guidelines and Attributes
5.1.2 Evolution of Software Design
5.2 Design Concepts
5.2.1 Abstraction
5.2.2 Architecture
5.2.3 Patterns
5.2.4 Separation of Concerns
5.2.5 Modularity
5.2.6 Information Hiding
5.2.7 Functional Independence
5.2.8 Refinement
5.2.9 Aspects
5.2.10 Refactoring
5.2.11 Object Oriented Design Concepts
5.2.12 Design Classes
5.2.13 Dependency Inversion
5.2.14 Design for Test
5.3 The Design Model
5.3.1 Data Design Elements
5.3.2 Architectural Design Elements

5.3.3 Interface Design Elements
5.3.4 Component-Level Diagram
5.4.5 Deployment-Level Diagram

**Reference Books:**

1. Software Engineering : A Practitioner's Approach - Roger S. Pressman, McGraw hill(Eighth Edition) ISBN-13: 978-0-07-802212-8, ISBN-10: 0-07-802212-6
2. A Concise Introduction to Software Engineering - Pankaj Jalote, Springer ISBN: 978-1-84800-301-9
3. The Unified Modeling Language Reference Manual - James Rambaugh, Ivar Jacobson, Grady Booch ISBN 0-201-30998-X

**Savitribai Phule Pune University**
**S.Y.B.Sc. (Computer Science)**
**Computer Science Paper - III**
**Course Code: CS 233**
**Title : Practical course on CS 231 (Data Structures and Algorithms I) and CS 232 (Software Engineering)**

| Teaching Scheme | No. of Credits | Examination Scheme |
|---|---|---|
| 4 hrs 20 mins / week<br>Batch Size : 12 | 2 | IE : 15 marks<br>UE: 35 marks |

**Operating Environment:**
For Data Structures:

- **Operating system:** Linux
- **Editor:** Any linux based editor like vi, gedit etc.
- **Compiler** : cc or gcc

**Lab Book:**
The lab book is to be used as a hands-on resource, reference and record of assignment submission and completion by the student. The lab book contains the set of assignments which the student must complete as a part of this course.

**Programming Assignments:**
Programs should be done individually by the student intheir respective login. The codes should be uploaded on either the local server, Moodle, Github or any open source LMS. Print-outs of the programs and output may be taken but not mandatory for assessment.

**Assessment:**
Continuous assessment of laboratory work is to be done based on overall performance and lab assignments performance of student. Each lab assignment assessment will be assigned grade/marks based on parameters with appropriate weightage. Suggested parameters for overall assessment as well as each lab assignment assessment include-timely completion, performance, innovation, efficient codes and good programming practices.

- **Internal Evaluation :**
  - 10 marks will be given based on a mini project of Software Engineering.
  - 5 marks will be allocated for Assignment completion and practical attendance.
- **University Evaluation :**
  - The Practical slip will be of 35 Marks which will be based on Data structures.

**Course Contents:**

**Suggested Assignments for Data Structures – I**

**Assignment1: Searching Algorithms**
Implementation of searching algorithms to search an element using: Linear Search, Sentinel Search, Binary Search (with time complexity)

**Assignment 2: Sorting Algorithms - I**
Implementation of sorting algorithms: Bubble Sort, Insertion Sort, Selection Sort

**Assignment 3: Sorting Algorithms - II**
Implementation of sorting algorithms: Quick Sort, Merge Sort , Counting Sort

**Assignment 4: Singly Linked List**

1. Dynamic implementation of Singly Linked List to perform following operations: Create, Insert, Delete, Display, Search, Reverse
2. Create a list in the sorted order.

**Assignment 5: Doubly Linked List**

1. Dynamic implementation of Doubly circular Linked List to perform following operations: Create, Insert, Delete, Display, Search

**Assignment 6: Linked List Applications**

1. Merge two sorted lists.

Addition of two polynomials in a single variable.

**Assignment 7: Stack**

1. Static and Dynamic implementation of Stack to perform following operations: Init, Push, Pop, Peek, Isempty, Isfull

**Assignment 8: Applications of Stack**

1. Implementation of an algorithm that reverses string of characters using stack and checks whether a string is a palindrome.
2. Infix to Postfix conversion.
3. Evaluation of postfix expression.

**Assignment 9: Linear Queue**

1. Static and Dynamic implementation of linear Queue to perform following operations: Init, enqueue, dequeue Peek, IsEmpty, IsFull.

**Assignment 10: Circular and Priority Queue**

1. Implementation of circular queue
2. Implementation of priority queue

| **Suggested Assignments for Software Engineering mini Project** | **3** |
|---|---|

1. Prepare detailed statement of problem for the selected mini project
2. Identify suitable process model for the same.
3. Develop Software Requirement Specification for the project.
4. Identify scenarios and develop UML Use case
5. Other artifacts: Class Diagram, activity diagram, sequence diagram, component diagram and any other diagrams as applicable to the project.

Sample project titles: (These are just samples, students are suggested to take up different case studies)

1. Online mobile recharge system
2. Credit calculation system
3. Image sharing and editing system
4. Internal examination system
5. e-learning management system

**Savitribai Phule Pune University**
**S.Y.B.Sc. (Computer Science)**
**Computer Science Paper - I**
**Course Code: CS 241**
**Title : DATA STRUCTURES AND ALGORITHMS-II**

| Teaching Scheme | No. of Credits | Examination Scheme |
|---|---|---|
| 3 Lectures / week (50 mins. duration) | 02 | IE : 15 marks<br>UE: 35 marks |

Prerequisites :

- Knowledge of C Programming Language
- Basic knowledge of algorithms
- Basic knowledge of linear data structures

Course Objectives

- To learn the systematic way of solving problems
- To design algorithms
- To understand the different methods of organizing large amount of data
- To efficiently implement the non-linear data structures

Course Outcomes: On completion of this course students will be able to

- Implementation of different data structures efficiently
- Usage of well-organized data structures to handle large amount of data
- Usage of appropriate data structures for problem solving

**Course Contents**

| **Chapter 1** | **Tree** | **10 lectures** |
|---|---|---|

1.1 Concept and Terminologies
1.2 Types of Binary trees - Binary tree, skewed tree, strictly binary tree, full binary tree, complete binary tree, expression tree, binary search tree, Heap
1.3 Representation – Static and Dynamic
1.4 Implementation and Operations on Binary Search Tree - Create, Insert, Delete, Search, Tree traversals– preorder, inorder, postorder ( recursive implementation), Level-order traversal using queue, Counting leaf, non-leaf and total nodes, Copy, Mirror.
1.5 Applications of trees
1.5.1 Heap sort, implementation
1.5.2 Introduction to Greedy strategy, Huffman encoding (implementation using priority queue)

| **Chapter 2** | **Efficient Search Trees** | **8 lectures** |
|---|---|---|

2.1 Terminology: Balanced trees - AVL Trees, Red Black tree, splay tree, Lexical search tree -Trie
2.2 AVL Tree- concept and rotations
2.3 Red Black trees - concept, insertion and deletion.
2.4 Multi-way search tree - B and B+ tree - Insertion, Deletion

| **Chapter 3** | **Graph** | **12 lectures** |
|---|---|---|

3.1 Concept and terminologies
3.2 Graph Representation –Adjacency matrix, Adjacency list, Inverse Adjacency list, Adjacency multilist
3.3 Graph Traversals – Breadth First Search and Depth First Search (with implementation)
3.4 Applications of graph

3.4.1 Topological sorting
3.4.2 Use of Greedy Strategy in Minimal Spanning Trees (Prims and Kruskals algorithm)
3.4.3 Single source shortest path - Dijkstra's algorithm
3.4.4 Dynamic programming strategy, All pairs shortest path - Floyd Warshall algorithm
3.4.5 Use of graphs in social networks

| **Chapter 4** | **Hash Table** | **6 lectures** |
|---|---|---|

4.1 Concept of hashing
4.2 Terminologies – Hash table,Hash function, Bucket, Hash address, collision, synonym, overflow etc.
4.3 Properties of good hash function
4.4 Hash functions : division function, MID square , folding methods
4.5 Collision resolution techniques
4.5.1 Open Addressing - Linear probing, quadratic probing, rehashing
4.5.2 Chaining - Coalesced , separate chaining

**Reference Books:**


1. Fundamentals of Data Structures in C- Ellis Horowitz, SartajSahni,Susan Anderson-Freed, 2nd Edition, Universities Press.
2. Data Structures using C and C++-YedidyahLangsam, Moshe J. Augenstein, Aaron M. Tenenbaum, Pearson Education
3. Data Structures: A Pseudo code approach with C, Richard Gilberg ,Behrouz A. Forouzan, Cengage Learning.
4. Introduction to Data Structures in C-Ashok Kamthane, Pearson Education
5. Algorithms and Data Structures, Niklaus Wirth, Pearson Education
6. Introduction to Algorithms—Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein--MIT Press
7. Fundamentals of Computer Algorithms-- Ellis Horowitz, SartajSahni, SanguthevarRajasekaran, Universities Press
8. The Algorithm Design Manual - Steven S Skiena, Springer

**Savitribai Phule Pune University**
**S.Y.B.Sc. (Computer Science)**
**Computer Science Paper - I Semester II**
**Course Code: CS 242 Title : Computer Networks-I**

| Teaching Scheme | No. of Credits | Examination Scheme |
|---|---|---|
| 3 lectures / week (50 mins. duration) | **02** | IE : 15 marks<br>UE: 35 marks |

**Prerequisites**
Principles of Digital Electronics
Communication Principles

**Course Objectives**
To prepare students with basic networking concepts: data communication, protocolsand standards, various topologies and applications of network.

**Course Outcomes**

1. Have a good understanding of the OSI and TCP/IP Reference Models and in particular have a good knowledge of Layers.
2. Understand the working of various protocols.
3. Analyze the requirements for a given organizational structure and select the most appropriate networking architecture and technologies

**Course Contents**

| **Chapter 1** | **Introduction to Networks and Network Models** | **4 lectures** |
|---|---|---|

1.1 Data communication, components, data representation
1.2 Networks, network criteria, network types - LAN, WAN, Switching, The Internet, Accessing the Internet
1.3 Network Software- Protocol hierarchies, Design Issues of the layer, Connection Oriented and Connectionless Services,
1.4 Reference models - OSI Reference Models, TCP/IP Reference model, Connection devices in different layers, Comparison of OSI and TCP/IP Reference Models.

| **Chapter 2** | **Lower Layers** | **10 lectures** |
|---|---|---|

2.1 Communication at the physical layer, data rate limits - Noiseless channel (Nyquist bit rate), noisy channel (Shannon capacity), Performance - bandwidth, throughput, latency, bandwidth-delay product, jitter
2.2 Design issues of Data Link Layer, Services - Framing, flow control, error control, congestion control, Link layer addressing
2.3 Framing Methods - Character Count, Flag bytes with Byte Stuffing, Flags bits with Bit Stuffing, Physical Layer Coding Violations
2.4 The Channel allocation problem, Static and dynamic allocation, Media Access Methods - Taxonomy of multiple-access protocols
2.5 Switching and TCP/IP layers, Types - circuit switching, packet switching and message switching
2.6 Wired LANs - Standard Ethernet characteristics, Addressing, Access method, implementation, Fast and Gigabit Ethernet
2.7 Wireless LANs - Architectural comparison, Characteristics, Access control, IEEE 802.11

architecture, Physical layer, MAC sublayer, Bluetooth architecture, Layers

| **Chapter 3** | **Network Layer** | **12 lectures** |
|---|---|---|

3.1 Network layer services - Packetizing, Routing and forwarding, other services
3.2 Open and closed loop congestion control
3.3 IPv4 addressing- Address space, classful addressing, Subnetting, Supernetting, classless addressing, Network address resolution (NAT)
3.4 Forwarding of IP packets- based on destination address, based on label
3.5 Network Layer Protocols- Internet Protocol (IP), IPv4 datagram format, Fragmentation, options
3.6 Mobile IP-addressing, agents, Three phases
3.7 Next Generation IP- IPv6 address representation, address space, address types, IPv6 protocol, packet format, extension header, Difference between IPv4 and IPv6
3.8 Routing - General idea, Algorithms - Distance vector routing, link state routing, path-vector routing

| **Chapter 4** | **Transport Layer** | **10 Lectures** |
|---|---|---|

4.1 Transport layer Services- Process-to-process communication, Addressing, Encapsulation and decapsulation, Multiplexing and demultiplexing, Flow control, Pushing or pulling, Flow control, Buffers, Sequence numbers, Acknowledgements, sliding window, congestion control
4.2 Connectionless and Connection-oriented service, Port numbers
4.3 Transport layer protocols- User datagram protocol, user datagram, UDP services
4.4 Transmission Control Protocol - TCP Services, TCP Features, TCP Segment format, three-way handshake for connection establishment and termination, State transition diagram, windows in TCP.

**Reference Books:**

1. Computer Networks-Andrew S. Tanenbaum, 5th Edition, Pearson Education
2. Data Communication and Networking- BehrouzFourouzan, 5th Edition, McGraw Hill Pvt. Ltd.

**Savitribai Phule Pune University**
**S.Y.B.Sc. (Computer Science)**
**Computer Science Paper - III**
**Course Code: CS 243**
**Title : Practical course on CS 241(Data Structures and Algorithms II) and CS 242 (Computer Networks I)**

| Teaching Scheme | No. of Credits | Examination Scheme |
|---|---|---|
| 4 hrs 20 mins / week<br>Batch size : 12 | 2 | IE : 15 marks<br>UE: 35 marks |

**Lab Book:**

The lab book is to be used as a hands-on resource, reference and record of assignment submission and completion by the student. The lab book contains the set of assignments which the student must complete as a part of this course.

**Programming Assignments:**

Programs should be done individually by the student in the respective login. The codes should be uploaded on either the local server, Moodle, Github or any open source LMS. Print-outs of the programs and output may be taken but not mandatory for assessment.

**Assessment:**

Continuous assessment of laboratory work is to be done based on overall performance and lab assignments performance of student. Each lab assignment assessment will be assigned grade/marks based on parameters with appropriate weightage. Suggested parameters for overall assessment as well as each lab assignment assessment include-timely completion, performance, innovation, efficient codes and good programming practices.

- **Internal Evaluation :**
  - 10 marks will be given based on Networking assignments.
  - 5 marks will be allocated for Assignment completion and practical attendance
- **University Evaluation :**
  - The Practical slip will be of 35 Marks which will be based on Advanced Data structures.

**Operating Environment:**

For Data Structures:

- **Operating system:** Linux
- **Editor:** Any linux based editor like vi, gedit etc.
- **Compiler** : cc or gcc

**Course Contents :-**

**Assignment 1 Binary Search Tree and Traversals**

1. Implement Binary Search Tree (BST) to perform following operations on BST– Create, Recursive Traversals - Inorder, Preorder, Postorder
2. Perform following operations: insert, delete

**Assignment 2 Binary Search Tree Operations**

1. Implement Binary Search Tree (BST) to perform following operations on BST–copy and mirror image of BST, counting leaf, non-leaf and total nodes.
2. Level-order traversal of binary search tree using queue.

**Assignment 3 Applications of Binary Tree**

1. Sort set of elements using Heap sort
2. Encode a set of characters using Huffman encoding

**Assignment 4 Graph implementation**

1. Implement Graph as adjacency matrix and adjacency list
2. Calculate indegree and outdegree of vertices
3. Graph traversals: BFS and DFS.

**Assignment 5 Graph Applications - I**

1. Implementation of Topological sorting
2. Implementation of Prims/Kruskals Minimum spanning tree algorithm

**Assignment 6 Graph Applications - II**

1. Implementation of Dijkstra's shortest path algorithm for finding Shortest Path from a given source vertex using adjacency cost matrix.
2. Implementation of Floyd Warshall algorithm for all pairs shortest path.

**Assignment 7 Hash Table**

1. Implementation of static hash table with Linear Probing.
2. Implementation of static hash table with chaining.

**Assignment 8 Hash Table-2**

1. Implementation of linked hash table with chaining.

**Assignment 9 Networking Assignment**

**Assignment 10 Networking Assignment**

**University of Pune**

**Board of Studies in Mathematics**

**S. Y. B. Sc. (Computer Science)**

**Syllabus of Mathematics**

## Introduction:

Savitribai Phule Pune University, Pune has decided to change the syllabi of various faculties from June, 2020. Taking into consideration the rapid changes in science and technology and new approaches in different areas of mathematics and related subjects Board of studies in Mathematics with concern of teachers of Mathematics from different colleges affiliated to Savitribai Phule Pune University, Pune has prepared the syllabus of S.Y.B.Sc. Computer Science Mathematics. To develop the syllabus the U.G.C. Model curriculum is followed.

## Aims:

i) Give the students a sufficient knowledge of fundamental principles ,methods and a clear perception of innumerous power of mathematical ideas and tools and know how to use them by modeling ,solving and interpreting.

ii) Reflecting the broad nature of the subject and developing mathematical tools for continuing further study in various fields of science.

iii) Enhancing students overall development and to equip them with mathematical modeling abilities, problem solving skills , creative talent and power of communication necessary for various kinds of employment .

iv) Enabling students to develop a positive attitude towards mathematics as an interesting and valuable subject of study.

## Objectives:

(i) A student should be able to recall basic facts about mathematics and should be able to display knowledge of conventions such as notations, terminology and recognize basic geometrical figures and graphical displays, state important facts resulting from their studies.

(ii) A student should get a relational understanding of mathematical concepts and concerned structures, and should be able to follow the patterns involved, mathematical reasoning.

(iii) A student should get adequate exposure to global and local concerns that explore them many aspects of Mathematical Sciences.

(iv) A student be able to apply their skills and knowledge, that is, translate information presented verbally into mathematical form, select and use appropriate mathematical formulae or techniques in order to process the information and draw the relevant conclusion.

(v) A student should be made aware of history of mathematics and hence of its past, present and future role as part of our culture.

*** Medium of Instruction: English**

*** Eligibility:** F.Y.B.Sc. Computer Science, as per University rules.

**Structure of the course:**

| **Semester - I** | | | **Semester -II** | |
|---|---|---|---|---|
| **Paper I** | **MTC-231** | **Groups and Coding Theory** | **MTC-241** | **Computational Geometry** |
| **Paper II** | **MTC-232** | **Numerical Techniques** | **MTC-242** | **Operations Research** |
| **Paper III** | **MTC-233** | **Mathematics Practical: Python Programming Language-I** | **MTC-243** | **Mathematics Practical: Python Programming Language-II** |

* **All three above courses are compulsory.**

*** External Students: Not allowed.**

*** Variation / Revaluation: Allowed for Paper- I and Paper-II.**

*** Qualifications for Teacher: M.Sc. Mathematics (with NET /SET as per existing rules )**

**Equivalence of Previous syllabus along with new syllabus:**

| | **Semester-III** | | **Semester-IV** | |
|---|---|---|---|---|
| | **New Course** | **Old Course** | **New Course** | **Old Course** |
| **Paper I** | **MTC-231: Groups and Coding Theory** | **MTC-211 : Applied Algebra** | **MTC-241: Computational Geometry** | **MTC-221: Computational Geometry** |
| **Paper II** | **MTC-232: Numerical Techniques** | **MTC-212: Numerical Analysis** | **MTC-242: Operations Research** | **MTC-222: Operations Research** |

| | | | | |
|---|---|---|---|---|
| **Paper III** | **MTC-233: Mathematics Practical: Python Programming Language-I** | **MTC-213 : Mathematics Practical** | **MTC-243: Mathematics Practical: Python Programming Language-II** | **MTC–223: Mathematics Practical** |

---

**Semester III**

**MTC-231 : Groups and Coding Theory**

**Unit 1. Integers** **[05 Lectures]**

1.1 Division Algorithm (without Proof)

1.2 G.C.D. using division algorithm and expressing it as linear combination

1.3 Euclid's lemma

1.4 Equivalence relation (revision), Congruence relation on set of integers, Equivalence class partition

**Unit 2. Groups** **[03 Lectures]**

2.1 Binary Operation

2.2 Group: Definition and Examples

2.3 Elementary Properties of Groups

**Unit 3. Finite Groups and Subgroups** **[10 Lectures]**

3.1 Order of a group, order of an element

3.2 Examples (Zn, +) and (U(n), *)

3.3 Subgroup definition, Finite subgroup test, subgroups of Zn

3.4 Generator, cyclic group, finding generators of Zn( Corollary 3,4 without proof)

3.5 Permutation group, definition, composition of two permutations, representation as product of disjoint cycles, inverse and order of a permutation, even/ odd permutation

3.6 Cosets: Definition, Examples and Properties, Lagrange Theorem(without Proof)

**Unit 4. Groups and Coding Theory** **[18 Lectures]**

4.1 Coding of Binary Information and Error detection

4.2 Decoding and Error Correction

4.3 Public Key Cryptography

**Text Books:-**

1. **Contemporary Abstract Algebra By J. A, Gallian (Seventh Edition)**
   **Unit 1:Chapter 0, Unit 2: Chapter 2, Unit 3: Chapter 3 ,4, 5 and 7**
2. **Discrete Mathematical Stuctures By Bernard Kolman, Robert C. Busby and Sharon Ross (6th Edition) Pearson Education Publication**
   **Unit 4: Chapter 11**

**MTC-232 : Numerical Techniques**

**Unit 1: Algebraic and Transcendental Equation [04 Lectures]**

1.1 Introduction to Errors

1.2 False Position Method

1.3 Newton-Raphson Method

**Unit 2: Calculus of Finite Differences and Interpolation [16 Lectures]**

2.1 Differences

2.2. Forward Differences

2.3 Backward Differences

2.4 Central Differences

2.5 Other Differences ($\delta$, $\mu$ operators)

2.6 Properties of Operators

2.7 Relation between Operators

2.8 Newton's Gregory Formula for Forward Interpolation

2.9 Newton's Gregory Formula for Backward Interpolation

2.10 Lagrange's Interpolation Formula

2.11 Divided Difference

2.12 Newton's Divided Difference Formula

**Unit 3: Numerical Integration [08 Lectures]**

3.1 General Quadrature Formula

3.2 Trapezoidal Rule

3.3 Simpson's one-Third Rule

3.4 Simpson's Three-Eight Rule

**Unit 4: Numerical Solution of Ordinary Differential Equation [08 Lectures]**

4.1 Euler's Method

4.2 Euler's Modified Method

4.3 Runge-Kutta Methods

**Text Book:-**

**1. A textbook of Computer Based Numerical and Statistical Techniques, by A. K. Jaiswal and Anju Khandelwal. New Age International Publishers.**

**Unit 1:** Chapter 2: Sec. 2.1, 2.5, 2.7
**Unit 2:** Chapter 3: Sec. 3.1, 3.2, 3.4, 3.5, Chapter 4: Sec. 4.1, 4.2, 4.3, Chapter 5: Sec. 5.1, 5.2, 5.4, 5.5
**Unit 3:** Chapter 6: Sec. 6.1, 6.3, 6.4, 6.5, 6.6, 6.7
**Unit 4:** Chapter 7: Sec. 7.1, 7.4, 7.5, 7.6

**Reference Books:-**


1. S.S. Sastry; Introductory Methods of Numerical Analysis, 3rd edition, Prentice Hall of India, 1999.
2. H.C. Saxena; Finite differences and Numerical Analysis, S. Chand and Company.
3. K.E. Atkinson; An Introduction to Numerical Analysis, Wiley Publications.
4. Balguruswamy; Numerical Analysis.


## MTC-233: Mathematics Practical: Python Programming Language-I

**Unit 1: Introduction to Python**

1.1 Installation of Python
1.2 Values and types: int, float and str,
1.3 Variables: assignment statements, printing variable values, types of variables.
1.4 Operators, operands and precedence:+, -, /, *, **, % PEMDAS(Rules of precedence)
1.5 String operations: + : Concatenation, * : Repetition
1.6 Boolean operator:
    1.6.1 Comparison operators: ==, !=, >, =, <=
    1.6.2 Logical operators: and, or, not
1.7 Mathematical functions from math, cmath modules.
1.8 Keyboard input: input() statement

**Unit 2: String, list, tuple**

2.1 Strings:
    2.1.1 Length (Len function)
    2.1.2 String traversal: Using while statement, Using for statement
    2.1.3 String slice
    2.1.4 Comparison operators (>, <, ==)
2.2 Lists:
    2.2.1 List operations
    2.2.2 Use of range function
    2.2.3 Accessing list elements
    2.2.4 List membership and for loop
    2.2.5 List operations
    2.2.6 Updating list: addition, removal or updating of elements of a list

2.3 Tuples:
- 2.3.1 Defining a tuple,
- 2.3.2 Index operator,
- 2.3.3 Slice operator,
- 2.3.4 Tuple assignment,
- 2.3.5 Tuple as a return value

**Unit 3: Iterations and Conditional statements**

3.1 Conditional and alternative statements, Chained and Nested Conditionals: if, if-else, if-elif-else, nested if, nested if-else

3.2 Looping statements such as while, for etc, Tables using while.

3.3 Functions:
- 3.3.1 Calling functions: type, id
- 3.3.2 Type conversion: int, float, str
- 3.3.3 Composition of functions
- 3.3.4 User defined functions, Parameters and arguments

**Unit 4: Linear Algebra**

4.1 Matrix construct, eye(n), zeros(n,m) matrices

4.2 Addition, Subtraction, Multiplication of matrices, powers and invers of a matrix.

4.3 Accessing Rows and Columns, Deleting and Inserting Rows and Columns

4.4 Determinant, reduced row echelon form, nullspace, columnspace, Rank

4.5 Solving systems of linear equations (Gauss Elimination Method, Gauss Jordan Method, LU- decomposition Method)

4.6 Eigenvalues, Eigenvectors, and Diagonalization

**Unit 5: Numerical methods in Python**

5.1 Roots of Equations

5.2 Newton-Raphson Method

5.3 False Position (Regula Falsi) Mehtod

5.4 Numerical Integration:
- 5.1.1 Trapezoidal Rule,
- 5.1.2 Simpson's 1/3$^{rd}$ Rule,
- 5.1.3 Simpson's 3/8$^{th}$ Rule

**Text Books:-**

1. **Downey, A. et al., How to think like a Computer Scientist: Learning with Python, John Wiley, 2015.**
   Sections: 1, 2, 3
2. **Robert Johansson, Introduction to Scientific Computing in Python**
   Section: 4

**Reference Books:-**


1. **Lambert K. A., Fundamentals of Python - First Programs, Cengage Learning India, 2015.**
2. **Guzdial, M. J., Introduction to Computing and Programming in Python, Pearson**

**India.**
3. **Perkovic, L., Introduction to Computing Using Python, 2/e, John Wiley, 2015.**
4. **Zelle, J., Python Programming: An Introduction to Computer Science, Franklin, Beedle & Associates Inc.**
5. **Sandro Tosi, Matplotlib for Python Developers, Packt Publishing Ltd.(2009)**

**Practicals:**
**Practical 1:** Introduction to Python, Python Data Types-I (Unit 1)
**Practical 2:** Python Data Types- II (Unit 2)
**Practical 3:** Control statements in Python-I (Unit 3- 3.1, 3.2)
**Practical 4:** Control statements in Python-II (Unit 3- 3.3)
**Practical 5:** Application : Matrices (Unit 4 – 4.1-4.3)
**Practical 6:** Application : Determinants, system of Linear Equations (Unit 4- 4.4, 4.5)
**Practical 7:** Application : System of equations (Unit 4- 4.5)
**Practical 8:** Application : Eigenvalues, Eigenvectors (Unit 4 – 4.6)
**Practical 9:** Application : Eigenvalues, Eigenvectors (Unit 4 – 4.6)
**Practical 10:** Application : Roots of equations (Unit 5 – 5.1)
**Practical 11:** Application : Numerical integration (Unit 5 – 5.2, 5.3)
**Practical 12:** Application : Numerical integration (Unit 5 – 5.4)

## Semester - IV

### MTC-241: Computational Geometry

**Unit 1. Two dimensional transformations: [12 Lectures]**

1.1 Introduction.
1.2 Representation of points.
1.3 Transformations and matrices.
1.4 Transformation of points.
1.5 Transformation of straight lines
1.6 Midpoint Transformation
1.7 Transformation of parallel lines
1.8 Transformation of intersecting lines
1.5 Transformation: rotations, reflections, scaling, shearing.
1.6 Combined transformations.
1.7 Transformation of a unit square.
1.8 Solid body transformations.
1.9 Translations and homogeneous coordinates.
1.10 Rotation about an arbitrary point.
1.11 Reflection through an arbitrary line.

**Unit 2. Three dimensional transformations: [08 Lectures]**

2.1 Introduction.
2.2 Three dimensional – Scaling, shearing, rotation, reflection, translation.
2.3 Multiple transformations.
2.4 Rotation about – an axis parallel to coordinate axes, an arbitrary line
2.5 Reflection through – coordinate planes, planes parallel to coordinate planes , an arbitrary plane

**Unit 3. Projection** **[08 Lectures]**

3.1 Orthographic projections.
3.2 Axonometric projections.
3.3 Oblique projections
3.4 Single point perspective projection

**Unit 4. Plane and space Curves:** **[08 Lectures ]**

4.1 Introduction.
4.2 Curve representation.
4.3 Parametric curves.
4.4 Parametric representation of a circle and generation of circle.
4.5 Bezier Curves – Introduction, definition, properties (without proof), Curve fitting (up to n = 3), equation of the curve in matrix form (upto n = 3)

**Textbook:**

**1. D. F. Rogers, J. A. Adams, Mathematical elements for Computer graphics, Mc Graw Hill Intnl Edition.**

Unit 1: Chapter 2: Sec. 2-1 to 2.17
Unit 2: Chapter 3: Sec. 3.1 to 3.10,
Unit 3: Chapter 3: Sec. 3.12 to 3.14
Unit 4: Chapter 4: Sec. 4.1, 4.2, 4.5, Chapter 5: Sec. 5.1, 5.8

**Reference books:**

1. Computer Graphics with OpenGL, Donald Hearn, M. Pauline Baker, Warren Carithers, Pearson ($4^{th}$ Edition)
2. Schaum Series, Computer Graphics.

## MTC-242: Operations Research

**Unit 1: Linear Programming Problem I** **[12 Lectures]**

1.1 Introduction Definition and Examples
1.2 Problem solving using Graphical method
1.3 Theory of Linear Programming, Slack and surplus variables, Standard form of LPP, Some important definitions, Assumptions in LPP, Limitations of Linear programming, Applications of Linear programming, Advantages of Linear programming Techniques
1.4 Simplex method, Big- M-method

**Unit 2: Linear Programming Problem II** **[08 Lectures]**

2.1 Special cases of LPP : Alternative solution, Unbounded solution, Infeasible solution
2.2 Duality in Linear Programming, Primal to dual conversion, Examples

**Unit 3: Assignment Models** **[06 Lectures ]**

3.1 Assignmment Model -Introduction
3.2 Hungerian method for Assignment problem

**Unit 4: Transportation Models** **[10 Lectures]**

4.1 Introduction, Tabular representation

4.2 Methods of IBFS (North-West rule, Matrix-minima, Vogel's Approximation), Algorithms

4.3 The Optimality Test of Transportation Model (MODI method only)

**Text Book:-**

**Operation Research (12 th Edition), by S.D.Sharma**.

Unit 1: Chapter 1: Sec. 1.1, 1.3-1, 1.3-2, 1.5, 1.6, 1.8, 1.9, 1.10, 1.11, 1.12,
Chapter 3: Sec. 3.1, 3.2, 3.3, 3. 4, 3.5-4,

Unit 2: Chapter 3: Sec. 3.8-1,3.8-2, Chapter 5: Sec. 5.1-1, 5.2-1,5.3,5.7-1, 5.7-2

Unit 3: Chapter 9: Sec. 9.1, 9.2, 9.4-1, 9.4-2, 9.5, 9.6, 9.7-1, 9.7-2

Unit 4: Chapter 10: 10.1, 10.2, 10.5, 10.8-1,10.9, 10.10

**Reference Books:-**


1. Operations Research by H. A. Taha
2. Operations Research by R. Panneerselvam, Prentice Hall of India.
3. Principles of Operations Research by H. M. Wagner, Prentice Hall of India.
4. Operations Research by Gupta and Hira.
5. Operation Research by J.K. Sharma


## MTC-243: Mathematics Practical: Python Programming Language-II

**Unit 1: 2D, 3D Graphs**

1.1 Installation of numpy, matplotlib packages

1.2 Graphs plotting of functions such as ... etc.

1.3 Different formats of graphs.

1.3 Three-dimensional Points and Lines

1.4 Three-dimensional Contour Plots

1.5 Wireframes and Surface Plots

1.6 Graphs plotting of functions such as... etc.

**Unit 2: Computational Geometry**

1.1 Points: The distance between two points, Lists of Points - the PointList class, Integer point lists, Ordered Point sets, Extreme Points of a PointList, Random sets of Points not in general position

2.2 Points: Displaying Points and other geometrical objects, Lines, rays, and line segments, The geometry of line segments, Displaying lines, rays and line segments

**2.3 Polygon :** Representing polygons in Python, Triangles, Signed area of a triangle, Triangles and the relationships of points to lines, is Collinear, is Left, is Left On, is Right, is Right On, Between

**2.4 Two dimensional rotation and reflection**

**2.5 Three dimensional rotation and reflection**

2.6 Generation of Bezier curve with given control points

**Unit 3: Study of Operational Research in Python**

3.1 Linear Programming in Python
3.2 Introduction to Simplex Method in Python

**Practicals:**

**Practical 1:** Graph Plotting (Unit 1 – 1.1 – 1.3)
**Practical 2:** Graph Plotting (Unit 1 – 1.4 – 1.7)
**Practical 3:** Application to Computational Geometry (Unit 2 – 2.1)
**Practical 4:** Application to Computational Geometry (Unit 2 – 2.2)
**Practical 5:** Application to Computational Geometry (Unit 2 – 2.3)
**Practical 6:** Study of Graphical aspects of Two dimensional transformation matrix using matplotlib
**Practical 7:** Study of Graphical aspects of Three dimensional transformation matrix using matplotlib
**Practical 8:** Study of Graphical aspects of Three dimensional transformation matrix using matplotlib
**Practical 9:** Study of effect of concatenation of Two dimensional and Three dimensional transformations
**Practical 10:** Generation of Bezier curve using given control points
**Practical 11:** Study of Operational Research in Python (Unit 3.1)
**Practical 12:** Study of Operational Research in Python (Unit 3.2)

**Text Books:-**

1. **Jaan Kiusalaas, Numerical Methods in Engineering with Python, Cambridge University Press, (2005)**
   Sections: 3
2. **Robert Johansson, Introduction to Scientific Computing in Python**
   Section: 1
3. **Jason Brownlee, Basics of Linear Algebra for Machine Learning, Discover the Mathematical Language of Data in Python**
   Sections: 2

**Reference Books:-**


**1. Lambert K. A., Fundamentals of Python - First Programs, Cengage Learning India, 2015.**
**2. Guzdial, M. J., Introduction to Computing and Programming in Python, Pearson India.**
**3. Perkovic, L., Introduction to Computing Using Python, 2/e, John Wiley, 2015.**
**4. Zelle, J., Python Programming: An Introduction to Computer Science, Franklin, Beedle and Associates Inc.**
**5. Jim Arlow, Interactive Computational Geometry in Python**


**Note:**

**(i) In paper -I , paper-II and paper-III, each course is of 50 marks ( 35 marks theory and 15 marks internal examination).**

**(ii) Paper III: Mathematics Practical - MTC-233 and MTC-243 is practical course and**

**is of 50 marks. Practicals shall be perforemed on computer.**

**Examination:**

**A) Pattern of examination: Paper- I, Paper-II and paper-III: Semesterwise**

**B) Pattern of question papers: For Paper -I and Paper-II**

**Q 1. Attempt any 05 out of 07 questions each of 01 marks. [05 Marks]**
**Q 2. Attempt any 02 out of 04 questions each of 05 marks. [10 Marks]**
**Q 3. Attempt any 02 out of 04 questions each of 05 marks. [10 Marks]**
**Q 4. Attempt any 02 out of 04 questions each of 10 marks. [10 Marks]**

**C) Instructions Regarding Practical:**

**Paper-III:Mathematics Practical:**

**(i) Mathematics Practical, external examiner shall be appointed by Savitribai Phule Pune University, Pune.**
**(ii) The minimum duration of parctical examination is 3 hours.**
**(iii) The semester examination is of 35 marks 15 marks are from internal evaluation (Journal, attendence and viva-voce or internal test etc.)**
**(iv) The slips for the questions on programming and problem solving using python shall be prepared and provided and these can be used at least for 3 years.**

**D) Standard of passing:**

**For Paper- I, Paper-II and Papaer -III: 14 Marks out of 35 and 06 marks out of 15 marks and total should be 20 marks for each course.**

# SAVITRIBAI PHULE PUNE UNIVERSITY

**(Formerly University of Pune)**

## S.Y. B. Sc. (Computer Science), Electronics

## Choice Based Credit System Syllabus

## To be implemented from

## Academic Year 2020–2021

## ( Under the faculty of Science and Technology )

# Savitribai Phule Pune University

## (Formerly University of Pune)

## SYLLABUS OF

## S. Y. B. Sc. (Computer Science), Electronics

### Choice Based Credit System

### To be implemented from A.Y. 2020-21

### Structure of S. Y. B. Sc.(Computer Science) Electronics

| Semester | Paper Code | Paper | Paper title | No. of Credit | Lectures/Week | Evaluation | | |
|---|---|---|---|---|---|---|---|---|
| | | | | | | CA | UE | Total |
| III | ELC-231 | I | Microcontroller Architecture & Programming | 2 | 3 (each lecture of 50 minutes) | 15 | 35 | 50 |
| | ELC-232 | II | Digital Communication and Networking | 2 | 3 (each lecture of 50 minutes) | 15 | 35 | 50 |
| | ELC-233 | III | Practical Course I | 2 | 1 pract / week (each practical of 04 hours & 20 minutes) | 15 | 35 | 50 |
| IV | ELC-241 | I | Embedded System Design | 2 | 3 (each lecture of 50 minutes) | 15 | 35 | 50 |
| | ELC-242 | II | Wireless Communication and Internet of Things | 2 | 3 (each lecture of 50 minutes) | 15 | 35 | 50 |
| | ELC-243 | III | Practical Course II | 2 | 1 pract / week (each practical of 04 hours & 20 minutes ) | 15 | 35 | 50 |

## S.Y.B.Sc.(Computer Science), Electronics- Semester III

## Paper-I: Microcontroller Architecture & Programming (ELC 231)

### Objectives:

1. To study the basics of 8051microcontroller
2. To study the Programming of8051microcontroller
3. To study the interfacing techniques of 8051microcontroller
4. To design different application circuits using 8051microcontroller

**Course Outcomes :** On completion of the course, student will be able

1. To write programs for 8051 microcontroller
2. To interface I/O peripherals to 8051 microcontroller
3. To design small microcontroller based projects

### COURSE CONTENTS

**UNIT- 1: Basics of Microcontroller & Intel 8051 architecture [08]**

Introduction to microcontrollers, difference in controller and processor.
Architecture of 8051, Internal block diagram, Internal RAM organization, SFRS, pin functions of 8051, I/O port structure & Operation, External Memory Interface.

**UNIT-2: Programming model of 8051 [10]**

Instruction classification, Instruction set, Addressing Modes: Immediate, register, direct, indirect and relative, assembler directives (ORG, END), features with examples, I/O Bit & Byte programming using assembly language for LED and seven segment display (SSD) interfacing.
Introduction to 8051 programming in C.

**UNIT- 3: Timer /Counter, Interrupts [10]**

Timer / counter: TMOD, TCON, SCON, SBUF, PCON Registers, Timer modes, programming for time delay using mode 1 and mode 2.
Interrupts: Introduction to interrupt, Interrupt types and their vector addresses, Interrupt enable register and interrupt priority register (IE, IP)

**UNIT- 4: Interfacing, Serial Communication [08]**

Programming of serial port without interrupt, Serial Communication: Synchronous and asynchronous serial communication, Use of timer to select baud rate for serial communication. Interfacing : ADC, DAC, LCD, stepper motor.

## Recommended books:

1. 8051 microcontroller and Embedded system using assembly and C : Mazidi and McKinley, Pearson publications
2. The 8051 microcontroller – Architecture, programming and applications: K.Uma Rao and Andhe Pallavi, Pearson publications.

## S.Y.B.Sc. Computer Science), Electronics, Semester III

## Paper-II, Digital Communication and Networking, ELC- 232

### Objectives:

1. To introduce to all aspects of data communication system
2. To introduce various digital modulation schemes
3. To identify the need of data coding and error detection/correction mechanism.
4. To study bandwidth utilization techniques : multiplexing and Spectrum spreading
5. To know data link layer protocol: Media Access Control
6. To study OSI and TCP/IP models of Networking.

**Course Outcomes :** On completion of the course, student will be able

1. Define and explain terminologies of data communication
2. Understand the impact and limitations of various digital modulation techniques
3. To acknowledge the need of spread spectrum schemes.
4. Identify functions of data link layer and network layer while accessing communication link
5. To choose appropriate and advanced techniques to build the computer network

.

### COURSE CONTENTS

**UNIT 1: Introduction to Electronic Communication (9)**

**Introduction to Communication:** Elements of Communication system, types of noise sources, Electromagnetic spectrum, signal and channel bandwidth,

**Types of communication:** simplex, half duplex, full duplex, baseband and broadband,

**Serial communication:** asynchronous and synchronous,

**Information Theory:** Information entropy, rate of information (data rate, baud rate), channel capacity, Nyquist theorem, Signal to noise ratio, Noise Figure, Shannon theorem,

**Error handling codes:** Necessity, Hamming code, CRC

**UNIT 2: Modulation and Demodulation (5)**

**Introduction to modulation and demodulation:** Concept and need of modulation and demodulation,

**Digital Modulation techniques:** Pulse Code Modulation (PCM), FSK, QPSK, QAM.

**UNIT 3: Multiplexing, Spectrum Spreading and Media Access Control (12)**

**Multiplexing techniques:** Frequency division multiplexing, wavelength division multiplexing, Time division multiplexing

**Spread Spectrum techniques:** Frequency hopping Spread Spectrum, Direct Sequence Spread Spectrum

**Media Access Control (MAC):**

Random Access Protocol: ALOHA, CSMA, CSMA/CD, CSMA/CA,
Controlled Access Protocols: Reservation, Polling, Token passing,
Channelization Protocols: FDMA, TDMA, CDMA.

**UNIT 4: Computer Networking (10)**

**Introduction to computer networks**

**Types of networks** : LAN, MAN, WAN, Wireless networks, Switching, Internet,

**Network topology :** point to point, Star, Ring, Bus, Mesh, Tree, Daisy Chain, Hybrid

**Network devices :** Repeater, Switch, Networking cables, Router, Bridge, Hub, Brouter, Gateway.

Wired LANs:-

**Ethernet:** Ethernet protocol, standard Ethernet, 100 MBPS Ethernet, Gigabit Ethernet, 10 Gigabit Ethernet,

**Computer network model:** OSI and TCP/IP.

## Recommended books:

1.Communication Electronics: Principles and Applications, Frenzel, Tata Mc Graw Hill publication, 5th edition.

2. Data Communication and Networking, Forouzan, Mc Graw Hill publication, 5th edition

3. Computer Networks, Tanenbaum, pHI publication, 5th edition

## S.Y.B.Sc.(Computer Science), Electronics, Semester III
## Paper III, Practical Course (ELC-233)

**Objectives:**

1. To get hands on training of Embedded C
2. To study experimentally interfacing of microcontroller
3. To design, build and test modulator and demodulators of digital communication
4. To build and test experimentally various techniques of wired communication
5. To develop practical skills of network setup

**Course Outcomes :** On completion of the course, student will be able

1. To design and build his/her own microcontroller based projects.
2. To acquire skills of Embedded C programming
3. To know multiplexing and modulation techniques useful in developing wireless application
4. Do build and test own network and do settings.

**Guidelines for Practical:**

- Practical batch size : 12
- Minimum no of Practical to be performed : 10
- At least five practical from each Group
- Electronics lab should have set up for embedded programming (Computers and microcontroller target and interfacing boards)

**COURSE CONTENTS**

### Group A: (Any 5)

1. Arithmetic, logical & code conversion problems using assembly/C programming
2. Interfacing of thumbwheel & seven segment display to 8051 microcontroller
3. Traffic light controller using 8051 microcontroller
4. Interfacing LCD to 8051Microcontroller
5. Waveform generation using DAC Interface to 8051Microcontroller

6. Event counter using opto-coupler, seven segment LED/LCD display interface to 8051Microcontroller
7. Speed Control of stepper motor using 8051 microcontroller

## Group B: (Any 5)

1. Study of 3 or 4 Bit Pulse Code Modulation technique
2. Study of Frequency Shift Keying
3. Study of Time Division Multiplexing
4. Study of Frequency Division Multiplexing
5. Study of Code Division Multiple Access System
6. Study of Error detection and correction by using Hamming Code technique
7. Study of Computer network components : Cables, Connectors, Routers, Switches, Ethernet and related interfacing cards
8. To study Configuration of IP and MAC address and to study Local Area Network setup

# S.Y.B. Sc. (Computer Science), Electronics, Semester IV

## Paper I : Embedded System Design (ELC-241)

**Objectives:**

1. To understand the concept of Embedded systems.
2. To study the design flow and available tools for an Embedded system.
3. To understand the implementation of embedded system using firmware and hardware components.
4. To acquire programming skills for the development of Embedded system design.
5. To develop practical skills for designing embedded system Applications.

**Course Outcomes :** On completion of the course, student will be able

1. To understand the difference between general computing and the Embedded systems.
2. To know the fundamentals of embedded systems.
3. Understand the use of Single board Computer (Such as Raspberry Pi) for an embedded system application.
4. Familiar with the programming environment to develop embedded systems and their interfaces with peripheral devices.
5. To develop familiarity with tools used to develop in an embedded environment.

## COURSE CONTENTS

**Unit 1:Introduction to Embedded systems using single board computers (SBC) (08)**

Single boards computer block diagram, types, Comparison of SBC models, Specifications, I/O devices (Storage, display, keyboard and mouse), Network access devices

**Unit 2: Architecture of System on Chip (SOC) (08)**

Architecture of SoC, Basic version Broad Coprocessor, Pin Description of Raspberry Pi, Architectural features: CPU Overview, CPU Pipeline stages, CPU Cache Organization, Branch Prediction & Folding (Concept), GPU Overview

**Unit 3:Programming using Python (10)**

Overview of Rasberian OS (Operating System), Installation, different types of Operating Systems

Basic Python Programming (Script programming):Variable & data types, Flow Control structures, Conditional statements ( If…Then…else),
Functions: I/O function (GPIO, Digital),Time functions, Library functions
Basic Arithmetic Programs: Addition, Subtraction, Multiplication, Division

**Unit 4 : Interfacing of devices using Python Programming (10)**

Basic interfacing: LED, Switch, LCD
Internal Advanced: Bluetooth, Wifi, Ethernet,
External advanced: Camera, Serial Communication GSM, Ultrasonic Sensor, PIR, Finger Print reader.

## Recommended Books:

1. Rasberry Pi CookBook: Software & Hardware problems and Solutions By Simon Monk( O'Reilly Media Inc.)
2. Raspberry Pi Hardware Reference by Warren Gay ( Apress)
3. Rasberry Pi User Guide By Eben Upton, Greath Halfacree (John Wiley & Sons, Inc.)
4. Learning Python with Rasberry Pi, by Alex Bradbury, Ben Everard, John Wiley & Sons, Inc
5. Learn Raspberry Pi programming with Python By Wolfram Donat ( Apress)

# S.Y.B.Sc.(Computer Science), Electronics, Semester IV
# Paper II: Wireless Communication and Internet of Things (ELC242)

## Objectives:

1. To learn and understand applications of wireless communication system
2. To learn and understand cellular system
3. To learn and understand architecture of short range Wireless Technologies
4. To learn and understand basics of Internet of Things
5. To study applications of IoT

**Course Outcomes:** Students will be able to

1. Know working of wireless technologies such as Mobile communication, GSM, GPRS
2. Become familiar with 3G and 4G Cellular Network Technologies for Data Connections.
3. Understand working principles of short range communication application
4. Get introduce to upcoming technology of Internet of Things
5. Explore themselves and develop new IoT based applications

## COURSE CONTENTS

**Unit1: Wireless Communication: Cellular Telephony (12)**

**Overview of wireless communication**,

**Introduction of cellular telephony system:** Frequency reuse, handoff strategies, Co-channel and adjacent channel interference, block diagram of mobile handset

**Overview of Cellular Telephony generations:** 1G to 5G,3G (W-CDMA, UMTS), 4G(LTE)

**GSM:** architecture, frame structure, mobility management,

**GPRS :** architecture, application

**Unit 2 : Short Range Wireless Technologies and Location Tracking (12)**

**Short range Technologies :**

*Bluetooth:* Bluetooth architecture, Bluetooth protocol stack, Bluetooth frame structure
*Zigbee:* Architecture, topologies, applications, Z wave: Protocol architecture, applications
*RFID:* working of RFID system, types of RFID tags, RFID frequencies, applications

**Location Tracking:** GPS system**:** components of GPS system (space segment, control segment, user segment), GPS receiver, Applications

**Unit 3: IoT Architecture (08)**

**Introduction to IOT** : Evolution of IOT, M2M and/or IOT, Seven layer architecture of IoT, Role of cloud in IoT, cloud topologies, Cloud access, Protocols in IoT, Cross connectivity across IoT system components:

- Device to Gateway-short range Wireless: cellphone as gateway, dedicated wireless Access points
- Gateway to cloud: Long range connectivity, (wired, cellular, Satellite, WAN)
- Direct Device to Cloud connectivity ,

**Networking technologies:** Low power local area networking (LPLAN), Low power wide area networking (LPWAN) technologies, comparison of LoRa, sigfox NB-IoT, Cat –M.

**Unit 4: IoT Applications (04)**

**Application domains**,

**Challenges in IoT :** Power consumption, Physical security, durability, Secure Connectivity, Secure Data Storage, Data volume, Scalability

**Case studies:**

Case Study 1: Smart Irrigation system for Agricultural field
Case Study 2:Home Automation
Case Study 3: Smart Cities

## Recommended books:

1. Wireless Communications Principles and Practice, Rappaport, Pearson publication
2. Mobile Communications, Jochen Schiller, Pearson publication
3. Internet of Things : Principles and Paradigms, Rajkumar Buyya and Dastjerdi, MK publishers
4. Internet of Things, Mayur Ramgir, Pearson publication

## S.Y.B.Sc.(Computer Science), Electronics, Semester IV
## Paper III, Practical Course (ELC-243)

### Objectives:

1. To use basic concepts for building various applications of embedded electronics.
2. To build experimental setup and test the circuits.
3. To develop skills of analyzing test results of given experiments.
4. Developing Trained Personals for educating and training for upcoming graduates in wireless communication.
5. Implement basic IoT applications on embedded platform

**Course Outcomes :** On completion of the course, students will be able

1. To design and develop own smart applications using Rasberry-Pi
2. To write Python program for simple applications
3. To build own IoT based system

### Guidelines :

- Practical batch size : 12
- Minimum no of Practical to be performed : 10
- Eight compulsory experiments: At least four practical from each Group
- One activity equivalent to 2 experiments by the student.
  a. Continuation of F. Y. activity.
  b. Electronics project Based on the Theory Courses learnt
  c. Documentation type experiments
  d. Presentation/Seminar on Electronics /advanced topic/research topics.

**Prerequisite:** Rasberry Pi boards, Arduino / LoRa boards

### COURSE CONTENTS

**Group A (any 4)**

1. Programming of Raspberry Pi to control LEDs attached to the GPIO pins
2. Programming of Raspberry Pi to get feedback from a switch connected to the GPIO pins

3. Programming of Raspberry Pi to detect temperature using temperature sensor

4. Programming of Raspberry Pi to detect light intensity using photocell sensor
5. Programming of Raspberry Pi for Motion detection
6. Programming of Raspberry Pi for image detection

**Group B (any 4)**

1. Study of GSM system (Message transmission & Reception).
2. To study working of SIM card in GSM handset
3. Study of GPRS system
4. Study of Zig-bee for one application
5. Study of RFID system
6. Introduction to Python programming.
7. To study Arduino based LED switching using mobile
8. Temperature and humidity sensing using Arduino
9. LoRa Interfacing.