

Shiksha Mandal's
Bajaj College of Science, Wardha
(Formerly known as Jankidevi Bajaj College of Science, Wardha)

Syllabus of
Bachelor of Science (B. Sc.) Three Year (Six Semesters) Degree Course
Semester I

Subject: Computer Science Major

(Approved in BoS meeting held on 15.05.2023 w.e.f. Academic Session 2023-24)

Programme Objective:

There are two main objectives to the B.Sc. Computer Science Programme.

- a) The primary goal of the degree program in Computer Science is to provide students the foundations for the future work and careers in computation-based courses. These foundations support both a successful career path in computing as well as provide appropriate qualifications for further degree work in computation related disciplines. Our degree emphasizes development of analytical skills, acquisition of knowledge and understanding of systems, languages and tools required for effective computation-based problem solving.
- b) A Computer Science major will provide you with knowledge of programming, operating systems, compiler design and principles of programming language. These skills will prepare graduates to enter technological fields such as systems programming, technical support, research and teaching.

On completion of the B.Sc. Computer Science Programme, the students:

- a) Will have sound knowledge of the Programming languages such as C, C++, Java, VB and SQL. Also they will have knowledge about Information Technology, Operating System, System Analysis and Design and Development.
- b) Can participate in off-campus internships, independent study projects, part-time computer employment, lectures by guest speakers, Society of Computing Students programming and on campus employment as lab assistants, lab supervisors and Helpdesk assistants.
- c) Be in a position to develop industrial and entrepreneur applications.

Aim of the Course

To understand the basics of computer hardware, software, network and algorithms, flowcharts and programming and also to provide sound knowledge of object oriented programming concepts and their understanding, development, working process of software.

Syllabus for B.Sc. SEMESTER I (To be implemented from Academic Session 2023-24)

Learning Objectives	Course Learning Outcomes
Students will try to learn:	After successful completion of the course student will be able to
1. To understand the concepts of fundamentals of a computer system, number systems, generations of computer languages, and other system tools like translator programs.	1. Understand various number systems, generations of computer languages and translators.
2. To understand various types of memories, storage devices, input and output devices.	2. Know various types of memories, storage devices, input and output devices.
3. To get acquainted with concepts of networks and its associated terminologies, topologies, architecture, communication devices and recent technologies like Bluetooth and Wi-Fi.	3. Learn the concepts of networks, topologies, architecture, communication devices and recent technologies.
4. To understand the basics of 'C' programming. Also get knowledge of algorithm and flowcharts.	4. Learn to make basics programs and also able to draw algorithm and flowcharts of programs.
5. To know the concepts of control and looping structure. Also student get the idea about concepts of arrays and functions.	5. Perform programs using control and looping structure also understand the basic programs on arrays and functions.
6. To understand the basics of structures, pointers and file handling.	6. Implementation of structures, pointers and file handling concepts in programming.

Subject: Computer Science Major
DSC - I
FUNDAMENTALS OF INFORMATION TECHNOLOGY and PROGRAMMING IN 'C'
Course Code: UCS110T

Credits: 6 (4 Th, 2 Pr)

No. of Lectures: 60

No. of Practical: 60

Course Outline

Unit I

Evolution of Computers, Generations of Computer, Characteristics of a Computer, Computer Classification.

Basic Components of Digital Computers: Block Diagram. CPU: Functions of Each Unit: Primary Memory, ALU and CU, Instruction format. Bus: Data, Control and Address Bus. Number Systems: Binary, Octal, Decimal, Hexadecimal, Their Conversions, Binary Arithmetic. ASCII, BCD, EBCDIC.

Language Evolution: Generation of Languages: Machine, Assembly, High Level Languages. Characteristics of Good Language Translators: Compiler, Interpreter and Assembler. Source and object Program.

Unit II

Memory: Memory hierarchy, RAM, Static & dynamic RAM, Types of RAM (SDRAM, RDRAM, DDR), ROM, PROM, EPROM, EEPROM, Flash and Cache. Storage Devices: Hard Disk, Zip Disk and Optical Disk. Pen Drive, Blu Ray and SSD.

Input Devices: Keyboard, Mouse, Light Pen, Touch Screen, Voice Input, MICR, OCR, OMR, Barcode Reader and Flatbed Scanner. Output Devices: VDU, Printers: Dot Matrix, Laser and Inkjet. Plotters: Drum, Flat-Bed and Inkjet.

Unit III

Computer Network: Goals and applications - Business Application, Home Application, Network terminology, Topologies: Linear, Circular, Tree and Mesh. Types of Networks: LAN, WAN, MAN. Repeaters, Bridge, Routers, Brouters and Gateway. Modem for Communication between PC's, Wi-Fi network, Introduction of Bluetooth and Infrared devices. Network protocols. Architecture: Peer-to-Peer, Client/Server. TCP/IP Reference Model with functionality of each layer.

Internet services (Introduction only): WWW – Web browser, URL, Internet search engines, WWW development languages, Electronic mail – E-mail address, e-mail message format, e-mail services (application based e-mail, webmail), how email works (client-server model), File Transfer Protocol – How FTP works (client-server model), Terminal network, Uses of Internet

Unit IV

Programming Languages and Tools, Machine language, Assembly language, High level languages. Problem Solving techniques: Development Tools: Algorithm,

Flowcharts and Pseudo code (Definition and its characteristics) Developing Algorithm and Drawing flowcharts.

C character set, Tokens, Identifiers, Keywords, Variables, Data types, Qualifiers. Operators and Expressions: Arithmetic, Relational, Logical, Bit-Wise, Increment, Decrement, Conditional and Special operators. typedef, Type Conversion, Constants, Declaring Symbolic Constants, Character Strings, Enumerated Data Types, Operator Precedence and Associativity. Library functions: Maths, character and string handling Functions.

Unit V

Control Structure: Compound statement, Selection statement: if, if-else, nested if, switch. Iteration statement: for, while, do..while, Nested loops, Jump statement: break, continue, goto. (Special emphasis on problem solving).

Arrays: Need, Types: Single and Two Dimensional Array. Strings: Strings Manipulation, Arrays of Strings, Evaluation order

Function: Function Components, Return Data type, Parameter Passing, Return by Reference, Default Arguments, Recursive Functions, Arrays with Functions. Macro: single-line and multi-line. Storage Classes. (Special emphasis on problem solving)

Unit VI

Structure: Declaration, Definition, Accessing structure members, Initialization, Nesting of Structures. Union: Unions, Differences between Structure and Union

Pointer: Introduction, Address Operator (&), Pointer variables, void pointers, Pointer Arithmetic, Pointers to Pointers.

File handling: Hierarchy of File Stream Classes, Opening & closing a file, Testing for errors, File Modes, File pointers and their manipulations, Sequential Access, Random Access, Command Line arguments.

Graphics: Initializing graphics, Drawing basic shapes, Graphics Color system, Types of pens and brushes.

Minor - I
FUNDAMENTALS OF INFORMATION TECHNOLOGY and PROGRAMMING IN 'C'
Course Code: UCS111T

Credits: 6 (4 Th, 2 Pr)

No. of Lectures: 60

No. of Practical: 60

Course Outline

Unit I

Evolution of Computers, Generations of Computer, Characteristics of a Computer, Computer Classification.

Basic Components of Digital Computers: Block Diagram. CPU: Functions of Each Unit: Primary Memory, ALU and CU, Instruction format. Bus: Data, Control and Address Bus. Number Systems: Binary, Octal, Decimal, Hexadecimal, Their Conversions, Binary Arithmetic. ASCII, BCD, EBCDIC.

Language Evolution: Generation of Languages: Machine, Assembly, High Level Languages. Characteristics of Good Language Translators: Compiler, Interpreter and Assembler. Source and object Program.

Unit II

Memory: Memory hierarchy, RAM, Static & dynamic RAM, Types of RAM (SDRAM, RDRAM, DDR), ROM, PROM, EPROM, EEPROM, Flash and Cache. Storage Devices: Hard Disk, Zip Disk and Optical Disk. Pen Drive, Blu Ray and SSD.

Input Devices: Keyboard, Mouse, Light Pen, Touch Screen, Voice Input, MICR, OCR, OMR, Barcode Reader and Flatbed Scanner. Output Devices: VDU, Printers: Dot Matrix, Laser and Inkjet. Plotters: Drum, Flat-Bed and Inkjet.

Unit III

Computer Network: Goals and applications - Business Application, Home Application, Network terminology, Topologies: Linear, Circular, Tree and Mesh. Types of Networks: LAN, WAN, MAN. Repeaters, Bridge, Routers, Brouters and Gateway. Modem for Communication between PC's, Wi-Fi network, Introduction of Bluetooth and Infrared devices. Network protocols. Architecture: Peer-to-Peer, Client/Server. TCP/IP Reference Model with functionality of each layer.

Internet services (Introduction only): WWW – Web browser, URL, Internet search engines, WWW development languages, Electronic mail – E-mail address, e-mail message format, e-mail services (application based e-mail, webmail), how email works (client-server model), File Transfer Protocol – How FTP works (client-server model), Terminal network, Uses of Internet

Unit IV

Programming Languages and Tools, Machine language, Assembly language, High level languages. Problem Solving techniques: Development Tools: Algorithm, Flowcharts and Pseudo code (Definition and its characteristics) Developing Algorithm and Drawing flowcharts.

C character set, Tokens, Identifiers, Keywords, Variables, Data types, Qualifiers. Operators and Expressions: Arithmetic, Relational, Logical, Bit-Wise, Increment, Decrement, Conditional and Special operators. typedef, Type Conversion, Constants, Declaring Symbolic Constants, Character Strings, Enumerated Data Types, Operator Precedence and Associativity. Library functions: Maths, character and string handling Functions.

Unit V

Control Structure: Compound statement, Selection statement: if, if-else, nested if, switch. Iteration statement: for, while, do..while, Nested loops, Jump statement: break, continue, goto. (Special emphasis on problem solving).

Arrays: Need, Types: Single and Two Dimensional Array. Strings: Strings Manipulation, Arrays of Strings, Evaluation order

Function: Function Components, Return Data type, Parameter Passing, Return by Reference, Default Arguments, Recursive Functions, Arrays with Functions. Macro: single-line and multi-line. Storage Classes. (Special emphasis on problem solving)

Unit VI

Structure: Declaration, Definition, Accessing structure members, Initialization, Nesting of Structures. Union: Unions, Differences between Structure and Union

Pointer: Introduction, Address Operator (&), Pointer variables, void pointers, Pointer Arithmetic, Pointers to Pointers.

File handling: Hierarchy of File Stream Classes, Opening & closing a file, Testing for errors, File Modes, File pointers and their manipulations, Sequential Access, Random Access, Command Line arguments.

Graphics: Initializing graphics, Drawing basic shapes, Graphics Color system, Types of pens and brushes.

Shiksha Mandal's
Bajaj College of Science, Wardha
(Formerly known as Jankidevi Bajaj College of Science, Wardha)

Bachelor of Science (B. Sc.) Computer Science

Three Year (Six Semesters) Degree Course

Semester II

Subject: Computer Science Major

(Approved in BoS meeting held on 31.08.2024 w.e.f. Academic Session 2024-25)

Aim of the Course

To understand the basics of OOP using C++, impart skills on various kinds of overloading, inheritance, file handling, and handling exceptions in C++ applications using suitable constructs.

Learning Objectives	Course Learning Outcomes
Students will try to learn:	After successful completion of the course student will be able to
1. To cover the fundamentals of object-oriented programming, such as classes, objects, abstraction, encapsulation, and polymorphism.	1. Recognize the importance of OOP and its features, as well as the differences between C and C++.
2. To understand the concepts of constructor and destructor; also familiar with operator overloading.	2. Understand and implement various sorts of overloading using C++ programs.
3. To get an idea of inheritance, dynamic objects, and inheritance.	3. Choose an appropriate inheritance when offering a solution to the given problem.
4. To investigate the working of virtual functions, polymorphism, and console I/O.	4. Implement virtual functions and console I/O.
5. To introduce pointers and file handling in C++.	5. Handle pointers, memory, and effective file management.
6. To study exception handling mechanisms and ways to handle exceptions.	6. Write C++ programs based on effective exception-handling constructs.

DSC-II
OBJECT-ORIENTED PROGRAMMING USING 'C++'
Course Code: UCS120T

Credits: 4

No. of Hrs.: 60

Course Outline

Unit I

Object Oriented Methodology: Elements of Object-Oriented programming, Objects, Classes, OOPs features.

Classes & Objects: Specifying a Class, Creating Objects, Accessing Class members, Defining member function, Outside Member Functions as inline, Accessing Member Functions within the class, Static data member, Access Specifiers: Private, Protected and Public Members.

Unit II

Constructors & Destructors: Introduction, Parameterized Constructors, Constructor Overloading, Constructors with Default Arguments, Copy Constructor, Dynamic Constructor, Destructor, Order of Construction and Destruction, Static data members with Constructor and Destructors, Virtual Destructor.

Operator Overloading: Definition, Overloadable Operators, Unary Operator Overloading, Binary overloading, Rules for Operator Overloading.

Unit III

Dynamic Objects: Pointers to Objects, Creating and Deleting Dynamic Objects: New and Delete operators, Array of Objects, Array of Pointers to Objects, Pointers to Object Members, this Pointer.

Inheritance: Definition, Abstract classes, Single, Multilevel, Multiple, Hierarchical, Hybrid Inheritance, Constructor and Destructor in Derived Classes.

Unit IV

Virtual Functions: Need for Virtual Functions, definition, Pure Virtual Functions, Abstract Classes and Rules for Virtual Function. Runtime polymorphism.

Managing Console I/O Operations: C++ Streams and Stream Classes, Unformatted and Formatted Console I/O operations.

Unit V

Working with Files: Introduction, Classes for File Stream operations, Opening and Closing a File, Detecting end-of-file, File Modes, File Pointers and their Manipulation, Sequential Input and Output Operations, Updating a File: Random Access

Unit VI

Exception Handling: Exception Handling Model, List of Exceptions, catch with ellipses catch() Handling Uncaught Exceptions, Fault Tolerant Design Techniques, Memory Allocation Failure Exception, Rules for Handling Exception Successfully.

Shiksha Mandal's
Bajaj College of Science, Wardha
(Formerly known as Jankidevi Bajaj College of Science, Wardha)

Syllabus of Semester II Computer Science Minor

(BSc Physics with Computer Science as Minor/ BSc Electronics with Computer Science as Minor/ BSc Mathematics with Computer Science as Minor)

(Approved in BoS meeting held on 31.08.2024 w.e.f. Academic Session 2024-25)

Aim of the Course

To understand the basics of OOP using C++, impart skills on various kinds of overloading, inheritance, file handling, and handling exceptions in C++ applications using suitable constructs.

Learning Objectives	Course Learning Outcomes
Students will try to learn:	After successful completion of the course student will be able to
1. To cover the fundamentals of object-oriented programming, such as classes, objects, abstraction, encapsulation, and polymorphism.	1. Recognize the importance of OOP and its features, as well as the differences between C and C++.
2. To understand the concepts of constructor and destructor; also familiar with operator overloading.	2. Understand and implement various sorts of overloading using C++ programs.
3. To get an idea of inheritance, dynamic objects, and inheritance.	3. Choose an appropriate inheritance when offering a solution to the given problem.
4. To investigate the working of virtual functions, polymorphism, and console I/O.	4. Implement virtual functions and console I/O.
5. To introduce pointers and file handling in C++.	5. Handle pointers, memory, and effective file management.
6. To study exception handling mechanisms and ways to handle exceptions.	6. Write C++ programs based on effective exception-handling constructs.

MINOR-II
OBJECT-ORIENTED PROGRAMMING USING 'C++'
Course Code: UCS121T

Credits: 4

No. of Hrs.: 60

Course Outline

Unit I

Object Oriented Methodology: Elements of Object-Oriented programming, Objects, Classes, OOPs features.

Classes & Objects: Specifying a Class, Creating Objects, Accessing Class members, Defining member function, Outside Member Functions as inline, Accessing Member Functions within the class, Static data member, Access Specifiers: Private, Protected and Public Members.

Unit II

Constructors & Destructors: Introduction, Parameterized Constructors, Constructor Overloading, Constructors with Default Arguments, Copy Constructor, Dynamic Constructor, Destructor, Order of Construction and Destruction, Static data members with Constructor and Destructors, Virtual Destructor.

Operator Overloading: Definition, Overloadable Operators, Unary Operator Overloading, Binary overloading, Rules for Operator Overloading.

Unit III

Dynamic Objects: Pointers to Objects, Creating and Deleting Dynamic Objects: New and Delete operators, Array of Objects, Array of Pointers to Objects, Pointers to Object Members, this Pointer.

Inheritance: Definition, Abstract classes, Single, Multilevel, Multiple, Hierarchical, Hybrid Inheritance, Constructor and Destructor in Derived Classes.

Unit IV

Virtual Functions: Need for Virtual Functions, definition, Pure Virtual Functions, Abstract Classes and Rules for Virtual Function. Runtime polymorphism.

Managing Console I/O Operations: C++ Streams and Stream Classes, Unformatted and Formatted Console I/O operations.

Unit V

Working with Files: Introduction, Classes for File Stream operations, Opening and Closing a File, Detecting end-of-file, File Modes, File Pointers and their Manipulation, Sequential Input and Output Operations, Updating a File: Random Access

Unit VI

Exception Handling: Exception Handling Model, List of Exceptions, catch with ellipses catch() Handling Uncaught Exceptions, Fault Tolerant Design Techniques, Memory Allocation Failure Exception, Rules for Handling Exception Successfully.