**Sant Gadge Baba Amravati University, Amravati**

**Faculty: Science and Technology**

**Programme: B.Sc. with Computer Science/ Computer Application [Voc/Non Voc] / IT**

**POs:**

After completion of graduation, students will be competent to:

PO1: Critical Thinking: Take informed actions after identifying the assumptions that frame our thinking and actions, checking out the degree to which these assumptions are accurate and valid, and looking at our ideas and decisions (intellectual, organizational, and personal) from different perspectives.

PO2: Effective Communication: Speak, read, write and listen clearly in person and through electronic media in English and in one Indian language, and make meaning of the world by connecting people, ideas, books, media and technology.

PO3: Social Interaction: Elicit views of others, mediate disagreements and help reach conclusions in group settings.

PO4: Effective Citizenship: Demonstrate empathetic social concern and equity centred national development, and the ability to act with an informed awareness of issues and participate in civic life through volunteering.

PO5: Ethics: Recognize different value systems including your own, understand the moral dimensions of your decisions, and accept responsibility for them.

PO6: Environment and Sustainability: Understand the issues of environmental contexts and sustainable development.

PO7: Self-directed and Life-long Learning: Acquire the ability to engage in independent and life-long learning in the broadest context socio-technological changes

**PSOs:**

At the end of this program, the students would be able to:

PSO1: Understand the computer hardware and software.

PSO2: use the knowledge of software installation.

PSO3: Select modern computing tools and techniques for programming task.

PSO4: Identify, analyze, formulate and develop computer-based solutions to meet desired needs within realistic constraints.

PSO5: Develop databases and perform operations on them.

PSO6: Identify research and development areas in multiple disciplines.

PSO7: Design and develop the small web applications.

**Employability Potential of the B.Sc. with Computer Science/Computer Application(Voc/Non-Voc)/Information Technology:**

The BSc with Computer Science, Computer application and IT helps develop a widely applicable skill set in computing with strong programming and mathematics skills, as well as wide ranging skills in project management, effective presentations and teamwork. Graduate with a portfolio of work fit to present to potential employers. Depending on your chosen pathway, you can focus on particular areas of interest such as programming, web development, design of database design and video games. Graduate with the Computer Science will be able to apply for a range of computational and mathematical jobs in the creative industries, business, finance, education, medicine, engineering and science. Typical job titles include:

- Data analyst
- Assistant Programmer
- Web designer
- Web developer
- Applications developer
- UI Developer
- Cyber security analyst
- Game designer
- Games developer

- Mobile App Developer
- Software Developer

Graduates of BSc Computer Science can find jobs in a variety of sectors like IT departments, MNCs, colleges, etc. in both private and government companies.

Some of the common job sectors, where a fresher and experienced professional can find a relevant job after completing this course are:

IT dept., Consultancies, Technical Support, Cyber Security, Software Engineering, MNCs, Website Development, Mobile App Development, Website Designing, Data Analyzation, Computer Manufacturers, Government Agencies, etc.

**Bachelor degree with Computer Science/ Computer Application/IT** has become one of the most favoured undergraduate programs for students now a day. A **career in Computer field** has been proved rewarding since last decade. This field has the potential to boost the career. After completing B.Sc in Computer science/ Computer Application/IT, one can always go for higher studies for a better career prospects. They can join Master of Computer Application (MCA) of M.Sc in Computer science course. (MCA and M.Sc – Computer Science are equivalent degrees as recognized by UGC).
India is known to be a leader in software and the IT sector. The software and IT companies are the major employers of computer science graduates and offers the best packages to the young graduates which are unmatched with any other branches of science. Information Technology is a consistently growing field in respect of job opportunities. **Computer science professionals** or software professionals in recent scenario have a very bright career prospect. With growing of IT and software companies, a variety of job opportunities for trained computer professionals are being offered not only in India and abroad as well. IT sector is quite broad in terms of employment and job options, which gives fresher's new opportunities to make successful careers. Computer graduate can also get jobs in non-IT companies like universities, research, private and public industries, government departments, business organizations, commercial organizations and the manufacturing sector, etc.
Future scope for B.Sc. Computer Science/CA and IT graduates:

- Offer higher studies such M.Sc. and Ph.D. , MCA and MBA
- Likewise, foreign Universities also accept computer graduates for higher studies.
- Computer student can become small or medium scale entrepreneur.
- Union and State Public service commissions like UPSC, MPSC, Bank Probationary officers, other competitive examinations, etc. offer a multitude of jobs and positions like Data Entry operator, Assistant Programmer, etc.
- Students can become Content Developer for IT industries.
- Employee at Security Printing and Minting co-operation of India.

**Sant Gadge Baba Amravati University, Amravati**

**Part B**

**Syllabus Prescribed for 03 Year UG Programme**

**Programme: B.Sc. Part I (Computer Science/ Computer Application [Voc/Non-Voc]/IT)**

**Semester I**

| Code of the Course/Subject | Title of the Course/Subject | (Total Number of Periods) |
|---|---|---|
| 1CS1 | **Fundamentals of Computer and C Programming** | **84** |

**COs**

Upon completion of this course successfully, Students would be able to -

1. Understand the computer, I/O and peripheral devices.
2. Understand concept of Operating systems.
3. Apply the Programming concepts.
4. Learn C language.
5. Write Simple C Programs.

| Unit | Content |
|---|---|
| Unit I | Introduction to Computer, Characteristics, Generations of Computers, Block diagram of Computer. **Memories**: Primary Memories : RAM, ROM, and its types, Cache Memory, Secondary Storage Devices : Hard Disk, SSD, Pen drives. **I/O Devices:** Keyboard, Mouse, Scanner, Touch Screen, Monitors: LCD & LED. Printers: Impact and non-impact. **(14 periods)** |
| Unit II | **Operating System**: Definition, Functions of Operating System, Types: Batch Mode, Multiprogramming, Time sharing , Online Real Time, Distributed O.S. Booting process. **Windows**: Introduction, Features and taskbars, Desktop, Customizing Desktop. **(14 periods)** |
| Unit III | Programming Concept: Algorithm, flowcharting, Types of programming languages, Programming process: Program design, Coding, Compilation & Execution, Testing & Debugging, Documentation. Structured Programming : History of C language, Advantages, Structure of C program, Character set, Identifiers, Keywords, Constants and Variables, Symbolic constants, Qualifiers, Type conversion. Operators and Expressions. **(14 periods)** |
| Unit IV | **I/O Operations** : Formatted I/O : scanf(), printf()<br>**Unformatted I/O** : getch(), getchar(), gets(), putch(), putchar(), puts().<br>**Control structures**: Branching: if, if-else, Conditional operator(? : ), nested if, switch. Looping: while, do-while, for statements, comma operator, goto, break, continue, nested loops. **(14 periods)** |
| Unit V | **Arrays** - Declaration and initialization of one and two dimensional array.<br>**Structure** - Definition, declaration, initialization, array of structure, nested structure, union.<br>**Pointers** - Declaration, initialization, pointers arithmetic **(11 periods)** |
| Unit VI | **Functions in C**: Introduction, definition of function, function prototype, categories of function, actual argument, formal argument, function calling: call by value, call by reference, function parameters, local and global variable, functions with array, function recursion.<br>**String functions** - String functions : strlen(), strcpy(), strcmp() & strcat()<br>**(14 periods)** |
| ***SEM:** Assignment, Class test, Study tour, Industrial visit, Group discussion or any other innovative practice/activity | |

| | |
|---|---|
| COs: | 1. To draw flowchart, learn Algorithms and write simple programs.<br>2. To assess the curricular skills acquired by students at college level through Assignments, Unit test, Internal Test, Group Discussion/Seminar/Mini Project, Study Tour |
| Activities | 1. Assignment<br>2. Group discussion<br>3. Study tour/ Industrial visit **(4 periods)** |

**Course Material/Learning Resources**

**Text books:**

1) Computer Fundamentals & Networking - P.K.Sinha
2) Programming in C: E Balagurusamy : TMH Publication.

**Reference Books:**

1) Fundamentals of Computer - V.Rajaraman
2) Computer Network-Andrew Tanenbaum
3) ABC of Internet - Christian Crumblish (BPB)
4) ANSI C- Dennis Ritche
5) Programming in C - V.Rajaraman
6) Programming with C: Venugopal K.R. TMH, Publication.
7) Programming with C: Byson Gottfried , Schaum Series Publication.
8) Fundamentals of IT and C programming by C H Sawarkar, A P Chendke, G P Gawali Dnyanpath Publication.
9) Web Technology and Advance Programming by Dr. P N Mulkalwar, M M Bhonde, A A Tayade. Dnyanpath Publication.

**Weblink to Equivalent MOOC on SWAYAM if relevant:**

- https://onlinecourses.swayam2.ac.in/cec19_cs06/preview
- https://onlinecourses.swayam2.ac.in/nou20_cs03/preview
- https://www.classcentral.com/course/swayam-computer-fundamentals-13950
- https://onlinecourses.nptel.ac.in/noc19_cs42/preview
- https://onlinecourses.swayam2.ac.in/aic20_sp06/preview
- https://onlinecourses.swayam2.ac.in/cec20_cs02/preview
- https://www.classcentral.com/course/swayam-introduction-to-programming-in-c-2486
- https://swayamprabha.gov.in/asset/new_team/images/course_files/R12-Introduction%20to%20Programming%20in%20C%20.pdf

**Any pertinent media (recorded lectures, YouTube, etc.) if relevant:**

- https://www.youtube.com/watch?v=eEo_aacpwCw
- https://www.youtube.com/watch?v=OGM2BJ29Syg
- https://www.youtube.com/playlist?list=PLWPirh4EWFpF_2T13UeEgZWZHc8nHBuXp

**Syllabus Prescribed for –BSc-I Year UG Programme**
**Programme: B.Sc. Part-I Sem-I** ( Computer Science / Computer Application [Voc/Non-Voc]/IT)
**Semester 1**

| Code of the Course/Subject | Title of the Course/Subject | (No. of Periods/Week) |
|---|---|---|
| **ICSLAB1** | Laboratory/Practical of Fundamentals of Computer and C Programming | **06 periods per Batch per Week** |

*** List of Practical/Laboratory Experiments/Activities etc.**

**Course Name: Fundamentals of Computer and C Programming**

**COs**

Upon completion of this course successfully, Students would be able to demonstrate/perform/accomplish the following

1. Write word processing task.
2. Create worksheet and perform operations on it.
3. Design, compile and debug programs in C language.
4. Classify conditional expressions and looping statement to solve problems associated with conditions and repetitions.
5. Demonstrate the programs using arithmetic and relational operators.
6. Implement the concept of various string handling functions.
7. Classify programming components that efficiently solve computing problems in real-world.

**List of Practical:**

1. Practical on Word Processing.
2. Practical on Spread Sheets.
3. Practical on Design of Presentation.
4. Write a program in 'C' to demonstrate Arithmetic Operations.
5. Write a program in 'C' to demonstrate If -Else Statement.
6. Write a program in 'C' to demonstrate Nested If Statement.
7. Write a program in 'C' to demonstrate Else..If ladder Statement.
8. Write a program in C to demonstrate Switch-case Statement.
9. Write a program in 'C' to demonstrate For Loop Statement.
10. Write a program in 'C' to demonstrate Nested For Loop Statement.
11. Write a program in 'C' to demonstrate While Loop Statement.
12. Write a program in 'C' to demonstrate Nested While Loop Statement.
13. Write a program in 'C' demonstrate Do-While Loop Statement.
14. Write a program in 'C' demonstrate Nested Do-While Loop Statement.
15. Write a program in 'C' demonstrate One-Dimensional Array.
16. Write a program in 'C' demonstrate Two-Dimensional Array.
17. Write a program in 'C' demonstrate String Functions.
18. Write a program in 'C' demonstrate Pointers.
19. Write a program in 'C' demonstrate Function.
20. Write a program in 'C' demonstrate Function Recursion.

**Weblink to Equivalent Virtual Lab if relevant:**

- https://www.programiz.com/c-programming/online-compiler/
- https://www.onlinegdb.com/online_c_compiler
- https://www.tutorialspoint.com/compile_c_online.php

**Distribution of Marks for Practical Examination**

**Time: 4 hours (One Day Examination) Marks: 50**
**Exercise-I .....15**
**Exercise-II.....15**
**Viva-Voce.....10**
**Record.....10**
**Total: 50**

## INSTRUCTIONAL GUIDELINES

**Laboratory/practical/practicum/hands-on/activity-based learning** is a learning that occurs in a space where students can observe, practice, do some activity, get hands-on, get practical training, gain programming knowledge and ideas either individually or in groups. This learning is not confined within a physical laboratory space, but can also occur in various forms of space such as the e-learning management system and computer-simulated virtual laboratories. Within the laboratory, learning may occur in many ways, often through observing a case or phenomena, performing hands-on practical trainings.

**Sample Examples for COs of some Lab/Practical Courses are as follows, which may be used for Reference purpose only.**

BOS should decide the COs for practical/lab courses/practicum/activities conscientiously.

**By the end of the Lab/Practical Course, generally students should be able to:**

1. Design Algorithm and flowchart, develop data base, procedure iteratively, reflectively, and responsively
2. Design and execute program, work independently, interpret results, and draw a reasonable, accurate conclusion.
3. Evaluate the process and outcomes of an experiment quantitatively and qualitatively,
4. Communicate the process and output of program and
5. Design Conduct an experiment collaboratively and ethically.

**Part B**

**Syllabus Prescribed for 2022-23 Year UGProgramme**

**Programme: B.Sc. I Semester – II (Computer Science / Computer Application [Voc/Non-Voc]/IT)**

| Code of the Course/Subject | Title of the Course/Subject | (Total Number of Periods) |
|---|---|---|
| ICS2 | Data Structure and OOPS | 84 Periods |

**COs**

Upon completion of this course successfully, Students would be able to -

1. Implement basic data structures such as arrays, stacks.
2. use linked list, trees and queues.
3. Apply Algorithm for solving problems like sorting, searching, insertion and deletion of data.
4. Describe the procedural and object-oriented paradigm with concepts of streams, classes, functions, data and objects.
5. Perform programming on functions, inline functions, constructor and destructor.
6. Perform programming on the concept of function overloading, operator overloading, virtual functions and polymorphism.

| Units | Content | |
|---|---|---|
| Unit I | **Data structure:** Introduction to data structure, Types of data structure: Primitive and Non-primitive, Linear and Non-linear data structure, Data structure operations. **Array:** Definition and concepts, Memory Representations, Operations: Traversing, Insertion, Deletion. **Stacks:** Definition and concepts, Memory Representations, Operations: Traversing, Insertion, Deletion. | 14 (Periods) |
| Unit II | **Queue:** Definition and concepts, Memory Representations, Operations: Traversing, Insertion, Deletion. Types of Queue. **Linked List:** Definition and concepts, Memory Representations, Types of Linked List, and Operations: Traversing, Insertion, Deletion. **Tree:** Definition and Terminologies, Memory Representations of Trees, Types of Trees : Binary Trees, Complete Binary Trees, Binary Search Trees, Traversing : Preorder, Inorder, Postorder, Insertion, Deletion. | 14 (Periods) |
| Unit III | **Searching and Sorting:** Definition and concept. **Searching Techniques:** Linear Search, Binary Search and Indexed Sequential Search. **Sorting Techniques:** Bubble Sort, Selection Sort, Insertion Sort, Radix Sort, Merge Sort and Quick Sort. | 14 (Periods) |
| Unit IV | **Object Oriented Programming:** Features, Advantages and Applications of OOPS. Comparisons between POP and OOP, Introduction to C++, Program structure in C++.<br>**Classes and Objects:** Classes and Objects Specifiers, Defining data member and member functions, Accessing members.<br>**Managing Console I/O:** Formatted and Unformatted, Usage of manipulators: endl & setw, Scope Resolution Operator. | 14 (Periods) |
| Unit V | **Functions in C++:** Passing objects to and returning objects from functions. Function Overloading and Default argument, Inline function, Friend function. Array of Objects, Pointer to objects, 'this' pointer. Constructor and Destructor: Types of constructor, Usage of Constructor. | 14 (Periods) |
| Unit VI | **Operator Overloading:** Definition, Overloading Unary and Binary operators. **Inheritance:** Definition, Types of Inheritance, Visibility mode; Types of inheritance with example, Virtual base classes and Abstract base classes. | 14 (Periods) |

| |
|---|
| ***SEM :** Assignment, Class test, Study tour, Industrial visit, Group discussion or any other innovative practice/activity |
| COs: 1. Acquire skill to work with core components of data structure<br>2. Acquire object oriented programming skill. |

| | |
|---|---|
| **Activities | 1. Assignment<br>2. Group discussion<br>3. Study tour/ Industrial visit |

**Course Material/Learning Resources**

**Text books:**

1. Object Oriented Programming with C++ : E Balagurusamy TMH
2. Data Structures , Seymour Lipschutz , Schaum's Outlines Series, Tata McGraw-Hill.
3. Text Book of Computer Science (Data Structure and C++): S D Pachpande, R B Ghayalkar and Athar Iqbal, Dnyanpath Publication.

**Reference Books:**

1. Fundamentals of Data Structures in C, Ellis Horowitz, Sartaj Sahni and Susan Anderson-Freed. W. H. Freeman and Company.
2. Object-Oriented Programming in C++, Fourth Edition, Robert Lafore, SAMS Publication.
3. Data Structure and Algorithms :Aho, Hopcroft, Ulman
4. Introduction to Data Structure : Bhagat Singh, Thomas L Naps
5. Mastering in C++ by K. R. Venugopalan.
6. Data Structure and C++: P.S.Bodkhe, A.A.Tayade, S.B.Agarmore, Dnyanpath Publication.

**Weblink to Equivalent MOOC on SWAYAM if relevant:**

1. https://onlinecourses.swayam2.ac.in/cec22_cs19/
2. https://onlinecourses.nptel.ac.in/noc22_cs92/
3. https://onlinecourses.nptel.ac.in/noc22_cs70/

**Any pertinent media (recorded lectures, YouTube, etc.) if relevant:**

1. https://www.youtube.com/watch?v=RBSGKIAvoiM
2. https://www.youtube.com/watch?v=zg9ih6SVACc

**Syllabus Prescribed for 2022-23 Year UG Programme**

**Programme: LAB 2 B.Sc. I Semester II (Computer Science / Computer Application [Voc/Non-Voc]/IT)**

**Semester - II**

| Code of the Course/Subject | Title of the Course/Subject | (Number of Periods/Week) |
|---|---|---|
| 1CSLAB2 | Data Structure and OOPs | 06 Periods/Batch per week |

**Course name: Data Structure and OOPs lab**
**COs**

Upon completion of this course successfully, Students would be able to demonstrate/perform/accomplish the following

1. Perform various operations Data structure using CPP.
2. Develop the concept of dynamic memory allocation through linked list.
3. Design stack and queue with contiguous and non-contiguous data storage mechanism.
4. Perform the various operations on binary tree.
5. Implement sorting on 1-D array using different techniques.

**Practical List of Data Structure**

1. Write a Data Structure program in C to insert the element into the STACK using PUSH operation.
2. Write a Data Structure program in C to delete the element from the STACK using POP operation.
3. Write a Data Structure program in C to insert the element into the QUEUE.
4. Write a Data Structure program in C to delete the element from the QUEUE.
5. Write a Data Structure program in C to insert the node into the Linked List.
6. Write a Data Structure program in C to delete the node from the Linked List.
7. Write a Data Structure program in C to demonstrate the Linear Search.
8. Write a Data Structure program in C to demonstrate the Binary Search.
9. Write a Data Structure program in C to demonstrate the Bubble Sort.
10. Write a Data Structure program in C to demonstrate the Sorting Algorithms.

**Practical List of Object Oriented Programming language**

1. Write a program in C++ to demonstrate Class and Object.
2. Write a program in C++ to demonstrate constructor and destructor.
3. Write a program in C++ to demonstrate Inline function.
4. Write a program in C++ to demonstrate the use of friend function.
5. Write a program in C++ for default argument.
6. Write a program in C++ for unary operator overloading.
7. Write a program in C++ for Binary operator overloading.
8. Write a program in C++ for function overloading.
9. Write a program in C++ for virtual base class.
10. Write a program in C++ to implement the Inheritance.

Weblink to Equivalent Virtual Lab if relevant:

1. http://cse01-iiith.vlabs.ac.in/

**Distribution of Marks for Practical Examination**

**Time: 4 hours (One Day Examination) Marks: 50**
**Exercise-I .....15**
**Exercise-II..... 15**
**Viva-Voce..... 10**
**Record .....10**
**Total: 50**

**By the end of the Lab/Practical Course, generally students should be able to:**

1. Collect data and revise the experimental procedure iteratively, reflectively, and responsively
2. Design and execute an experimental procedure, work independently, interpret experimental results, and draw a reasonable, accurate conclusion.
3. Evaluate the process and outcomes of an experiment quantitatively and qualitatively,
4. Extend the scope of an investigation whether or not results come out as expected,
5. Communicate the process and outcomes of an experiment, and
6. Conduct an experiment collaboratively and ethically.