

# **SAURASHTRA UNIVERSITY**

**RAJKOT – INDIA**



**Accredited Grade A by NAAC (CGPA 3.05)**

**CURRICULAM**

**FOR**

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

**Bachelor of Science (Computer Science)**

**(Semester - 1 and Semester - 2)**

**Effective From June – 2016**

**B.Sc. (C.S.) (Semester – 1)**

<b>CS-101: Programming Fundamentals using C and C++</b>		
<b>Objective:</b> Through this subject students will learn about the fundamental of C programming language (Procedure Oriented Programming - POP) and will learn in detail about C++ (Object Oriented Programming - OOP), Inheritance, Polymorphism and Exception Handling.		
<b>Unit No.</b>	<b>Topic</b>	<b>Details</b>
<b>1</b>	<b>Introduction to C and C++</b>	<ul style="list-style-type: none"> <li>• History of C and C++</li> <li>• Overview of Procedural Programming and Object-Orientation Programming</li> <li>• Using main() function, Structure of c and C++ program</li> <li>• Compiling and Executing Simple Programs in C and C++.</li> </ul>
	<b>Data Types, Variables, Constants, Operators and Basic I/O</b>	<ul style="list-style-type: none"> <li>• Data Types, Casting of Data Types</li> <li>• Declaring, Defining and Initializing Variables, Scope of Variables</li> <li>• Using Named Constants, Keywords</li> <li>• Operators (Arithmetic, Logical and Bitwise)</li> <li>• Using Comments in programs</li> <li>• Character I/O (getc, getchar, putc, putchar etc), Formatted and Console I/O (printf(), scanf(), cin, cout)</li> <li>• Using Basic Header Files (stdio.h, iostream.h, conio.h etc).creating user defined header file ,use of user defined header file</li> </ul>
<b>2</b>	<b>Expressions, Conditional Statements and Iterative Statements</b>	<ul style="list-style-type: none"> <li>• Simple Expressions in C++ (including Unary Operator Expressions, Binary Operator Expressions), Understanding Operators Precedence in Expressions</li> <li>• Conditional Statements (if construct, switch-case construct)</li> <li>• Understanding syntax and utility of Iterative Statements (while, do-while, and for loops</li> <li>• Use of break and continue in Loops</li> <li>• Using Nested Statements (Conditional as well as Iterative)</li> </ul>
	<b>Functions and Arrays</b>	<ul style="list-style-type: none"> <li>• Utility of functions, Call by Value, Call by Reference, Functions returning value, Void functions, Inline Functions, Return data type of functions, Functions parameters, Differentiating between Declaration and Definition of Functions, Command Line Arguments/Parameters in Functions, Functions with variable number of Arguments.</li> <li>• Creating and Using One Dimensional Arrays ( Declaring and Defining an Array, Initializing an Array, Accessing individual elements in an Array, Manipulating array elements using loops), Use Various types of arrays (integer, float and character arrays / Strings)</li> <li>• Two-dimensional Arrays (Declaring, Defining and Initializing Two Dimensional Array, Working with Rows and Columns), Introduction to Multi-dimensional arrays</li> </ul>

**Bachelor of Science (Computer Science) (Semester - 1 and Semester - 2)**

**Saurashtra University**

**Effective from June – 2016**

3	<b>Derived Data Types (Structures and Unions)</b>	<ul style="list-style-type: none"><li>• Understanding utility of structures and unions, Declaring, initializing and using simple structures and unions, Manipulating individual members of structures and unions,</li><li>• Array of Structures, Individual data members as structures, Passing and returning structures from functions, Structure with union as members, Union with structures as members.</li></ul>
	<b>Memory Allocation</b>	<ul style="list-style-type: none"><li>• Differentiating between static and dynamic memory allocation, use of malloc, calloc and free functions</li><li>• use of new and delete operators, storage of variables in static and dynamic memory allocation</li></ul>
	<b>File I/O, Preprocessor Directives</b>	<ul style="list-style-type: none"><li>• Opening and closing a file (use of fstream header file, ifstream, ofstream and fstream classes)</li><li>• Reading and writing Text Files, Using put(), get(), read() and write() functions, Random access in files</li><li>• Understanding the Preprocessor Directives (#include, #define, #error, #if, #else, #elif, #endif, #ifdef, #ifndef and #undef), Macros, command line argument.</li></ul>
4	<b>Pointers and References in C++</b>	<ul style="list-style-type: none"><li>• Understanding a Pointer Variable, Simple use of Pointers (Declaring and Dereferencing Pointers to simple variables), Pointers to Pointers, Pointers to structures, Problems with Pointers</li><li>• Passing pointers as function arguments, Returning a pointer from a function</li><li>• using arrays as pointers, Passing arrays to functions.</li><li>• Pointers vs. References, Declaring and initializing references, Using references as function arguments and function return values</li></ul>
	<b>Overview of Function Overloading and Operator Overloading</b>	<ul style="list-style-type: none"><li>• Need of Overloading functions and operators, Overloading functions by number and type of arguments, Looking at an operator as a function call, Overloading Operators (including assignment operators, unary operators)</li></ul>
5	<b>Using Classes in C++</b>	<ul style="list-style-type: none"><li>• Principles of Object-Oriented Programming</li><li>• Defining &amp; Using Classes, Class Constructors, Characteristics of constructor, Constructor Overloading, Destructors</li><li>• Class Variables &amp; Functions, Friendly functions, Const member, Objects as parameters</li><li>• Specifying the Protected and Private Access, Copy Constructors, Overview of Template classes and their use.</li></ul>
	<b>Inheritance, Polymorphism and</b>	<ul style="list-style-type: none"><li>• Introduction to Inheritance (Multi-Level Inheritance, Multiple Inheritance), Polymorphism (Virtual Functions, Pure Virtual Functions), Basics Exceptional Handling (using catch and throw, multiple catch</li></ul>

**Bachelor of Science (Computer Science) (Semester - 1 and Semester - 2)**  
**Saurashtra University**  
**Effective from June – 2016**

<b>Exception Handling</b>	statements), Catching all exceptions, Restricting exceptions, Rethrowing exceptions.
---------------------------	--

**References Books**

1. BjarneStroustrup, "The C++ Programming Language", 4th Edition, Addison-Wesley , 2013.
2. BjarneStroustrup, "Programming -- Principles and Practice using C++", 2nd Edition, Addison-Wesley 2014.
3. E Balaguruswamy, "Object Oriented Programming with C++", Tata McGraw-Hill Education, 2008.
4. Paul Deitel, Harvey Deitel, "C++ How to Program", 8th Edition, Prentice Hall, 2011.
5. Programming in ANSI C Author : E. Balaguruswami.
6. Stanley B. Lippman, JoseeLajoie, Barbara E. Moo, "C++ Primer", Published by Addison-Wesley, 5th Edition, 2012

**B.Sc. (C.S.) (Semester – 2)**

<b>CS-201: Data Structures</b>		
<p><b>Objective:</b> Through this subject students will learn about data structure - basic techniques of algorithm analysis, writing recursive methods, several sub-quadratic sorting algorithms including quick sort, insertion sort, bucket sort, the implementation of linked data structures such as linked lists and binary trees.</p>		
<b>Unit No.</b>	<b>Topic</b>	<b>Details</b>
<b>1</b>	<b>Arrays and pointers</b>	<ul style="list-style-type: none"> <li>• Single and Multi-dimensional Arrays, Sparse Matrices (Array and Linked Representation)</li> <li>• Pointers : Advantages and disadvantages of pointers, Declaring and initializing pointers, pointer arithmetic, Array of pointers</li> </ul>
	<b>Introduction</b>	<ul style="list-style-type: none"> <li>• Introduction</li> <li>• Primitive and simple structures</li> <li>• Linear and nonlinear structures file organization.</li> </ul>
<b>2</b>	<b>Stacks</b>	<ul style="list-style-type: none"> <li>• Implementing single / multiple stack/s in an Array</li> <li>• Prefix, Infix and Postfix expressions, Utility and conversion of these expressions from one to another</li> <li>• Applications of stack; Limitations of Array representation of stack</li> </ul>
	<b>Linked Lists</b>	<ul style="list-style-type: none"> <li>• Singly, Doubly and Circular Lists (Array and Linked representation)</li> <li>• Normal and Circular representation of Stack in Lists; Self Organizing Lists; Skip Lists</li> </ul>
<b>3</b>	<b>Queues</b>	<ul style="list-style-type: none"> <li>• Array and Linked representation of Queue, De-queue, Priority Queues , Circular queue, Function to insert an element into the queue</li> <li>• Function for deletion from circular queue, Circular queue with array implementation</li> </ul>
	<b>Recursion</b>	<ul style="list-style-type: none"> <li>• Developing Recursive Definition of Simple Problems and their implementation</li> <li>• Advantages and Limitations of Recursion</li> <li>• Understanding what goes behind Recursion (Internal Stack Implementation)</li> </ul>
<b>4</b>	<b>Graph</b>	<ul style="list-style-type: none"> <li>• Introduction</li> <li>• Adjacency matrix and adjacency lists</li> <li>• Graph traversal Depth first search (dfs) Implementation, Breadth first search (bfs) Implementation</li> <li>• Shortest path problem</li> <li>• Minimal spanning tree</li> </ul>
	<b>Trees</b>	<ul style="list-style-type: none"> <li>• Introduction to Tree as a data structure; Binary Trees (Insertion, Deletion , Recursive and Iterative Traversals on Binary Search Trees); Threaded Binary Trees (Insertion, Deletion, Traversals); Height-Balanced Trees (Various operations on AVL)</li> </ul>

**Bachelor of Science (Computer Science) (Semester - 1 and Semester - 2)**

**Saurashtra University**

**Effective from June – 2016**

<b>5</b>	<b>Searching and Sorting</b>	<ul style="list-style-type: none"><li>• Linear Search, Binary Search, Index Search, Comparison of Linear and Binary Search</li><li>• Selection Sort, Insertion Sort, Bucket sorting, Shell Sort, Merge Sort, Bubble Sort, Quick Sort, Comparison of Sorting Techniques</li></ul>
----------	------------------------------	--

**References Books**

1. Adam Drozdek, "Data Structures and algorithm in C++", Third Edition, Cengage Learning, 2012.
2. SartajSahni, Data Structures, "Algorithms and applications in C++", Second Edition, Universities Press, 2011.
3. Aaron M. Tenenbaum, Moshe J. Augenstein, YedidyahLangsam, "Data Structures Using C and C++", Second edition, PHI, 2009.
4. Pointer in C Author : Kanitkar.
5. D.S Malik, Data Structure using C++, Second edition, Cengage Learning, 2010.
6. Data Structure through C/C++ Author : Tennaunbuam.
7. Data and File Structure Author : Trembley & Sorrenson.