

**KAKATIYA UNIVERSITY**  
**FACULTY OF SCIENCE**  
**B.Sc. (Computer Science)**  
**CBCS Pattern with Effect from the Academic Year 2019-2020**

**Structure of Curriculum**

Course Title	Hours/Week		Credits
	Theory	Practical	
<b>Semester –I</b>			
Programming in C	4	3	4+1 = 5
<b>Semester –II</b>			
Programming in C++	4	3	4+1 = 5
<b>Semester –III</b>			
Data Structures using C++	4	3	4+1 = 5
<b>Semester –IV</b>			
Data Base Management Systems (DBMS)	4	3	4+1 = 5
<b>Semester –V</b>			
Programme in Java	4	3	4+1 = 5
<b>Semester –VI</b>			
Web Technologies	4	3	4+1 = 5

<b>AECC</b>			
Semester -I	Hours/Week		Credits
	Theory		
Fundamentals of Computer	2		2
<b>Semester -II</b>			
Hour/Week			
Office Automation	2		2
<b>SEC</b>			
<b>Semester -III</b>			
Python –I (Sec –I)	2		2
Operating Systems (Sec –II)	2		2
<b>Semester -IV</b>			
Python –II (Sec –III)	2		2
Operating Systems (Sec –IV)	2		2
<b>Generic Elective (GE)</b>			
<b>Semester -IV</b>			
Information Technologies	4		4
<b>Project/Optional</b>			
<b>Semester -VI</b>			
PHP with MY SQL	Theory 3	Practical 3	3+1=4



D. Ramesh

Chairperson Board of Studies in Computer Science, KU

**KAKATIYA UNIVERSITY**  
**FACULTY OF SCIENCE**  
**B.Sc. (Computer Science)**  
**SEMESTER – I**  
**Programming in C**

<b>Theory</b>	<b>4 Hours/Week</b>	<b>4 Credit</b>	<b>Internal marks = 20</b>
<b>Practical</b>	<b>3 Hours/Week</b>	<b>1 Credit</b>	<b>External Marks = 80</b>

### **Unit – I**

Computer Fundamentals: Introduction of Computers, Classification of Computers, Anatomy of a Computer, Memory Hierarchy, Introduction to OS, Operational Overview of a CPU.

Program Fundamentals: Generation and Classification of Programming Languages, Compiling, Interpreting, Loading, Linking of a Program, Developing Program, Software Development.

Algorithms: Definitions, Different Ways of Stating Algorithms (Step-form, Pseudo-code, Flowchart), Strategy for Designing Algorithms, Structured Programming Concept.

Basics of C: Overview of C, Developing Programs in C, Parts of Simple C Program, Structure of a C Program, Comments, Program Statements, C Tokens, Keywords, Identifiers, Data Types, Variables, Constants, Operators and Expressions, Expression Evaluation–precedence and associativity, Type Conversions.

### **Unit – II**

Input-Output: Non-formatted and Formatted Input and Output Functions, Escape Sequences,

Control Statements: Selection Statements – if, if-else, nested if, nested if-else, comma operator, conditional operator, switch; Iterative Statements–while, for, do-while; Special Control Statement–goto, break, continue, return, exit.

Arrays and Strings: One-dimensional Arrays, Character Arrays, Functions from ctype.h, string.h, Multidimensional Arrays.

### **Unit – III**

Functions: Concept of Function, Using Functions, Call-by-Value Vs Call-by-reference, Passing Arrays to Functions, Scope of Variables, Storage Classes, Inline Functions, and Recursion.

Pointers: Introduction, Address of Operator (&), Pointer, Uses of Pointers, Arrays and Pointers, Pointers and Strings, Pointers to Pointers, Array of Pointers, Pointer to Array, Dynamic Memory Allocation.

### **Unit – IV**

User-defined Data Types: Declaring a Structure (Union) and its members, Initialization Structure (Union), Accessing members of a Structure (Union), Array of Structures (Union), Structures verses Unions, Enumeration Types.

Files: Introduction, Using Files in C, Working with Text Files, Working with Binary Files, Files of Records, Random Access to Files of Records, Other File Management Functions.

**Textbook:** Pradip Dey, Manas Ghosh, Computer Fundamentals and Programming in C (2e)

### **References:**

1. Ivor Horton, Beginning C
2. Ashok Kamthane, Programming in C
3. Herbert Schildt, The Complete Reference C
4. Paul Deitel, Harvey Deitel, C How to Program
5. Byron S. Gottfried, Theory and Problems of Programming with C
6. Brian W. Kernighan, Dennis M. Ritchie, The C Programming Language
7. B. A. Forouzan, R. F. Gilberg, A Structured Programming Approach Using C

**KAKATIYA UNIVERSITY**  
**FACULTY OF SCIENCE**  
**B.Sc. (Computer Science)**  
**SEMESTER – I**

**Programming in C Lab**

Practical

3 Hours/Week

1 Credit Marks: 50

**Note:**

- Programs of all the Concepts from Text Book including exercises must be practice and execute.
- Faculty must take care about UG Standard Programs.
- In the external lab examination student has to execute two programs with compilation and deployment steps are necessary. Write the Pseudo Code and draw Flow Chart for the programs.
- Recommended to use Open Source Software: GCC on Linux; DevC++ (or) CodeBlocks on Windows10.
- External Vice-Voce is compulsory.

1. Write a program to find the largest two (three) numbers using if and conditional operator.
2. Write a program to print the reverse of a given number.
3. Write a program to print the prime number from 2 to n where n is given by user.
4. Write a program to find the roots of a quadratic equation using switch statement.
5. Write a program to print a triangle of stars as follows (take number of lines from user):

```
*
 * * *
 * * * * *
 * * * * * * *
```

6. Write a program to find largest and smallest elements in a given list of numbers.
7. Write a program to find the product of two matrices.
8. Write a program to find the GCD of two numbers using iteration and recursion.
9. Write a program to illustrate the use of storage classes.
10. Write a program to demonstrate the call by value and the call by reference concepts.
11. Write a program that prints a table indicating the number of occurrences of each alphabet in the text entered as command line arguments.
12. Write a program to illustrate use of data type enum.
13. Write a program to demonstrate use of string functions string.h header file.
14. Write a program that opens a file and counts the number of characters in a file.
15. Write a program to create a structure Student containing fields for Roll No., Name, Class, Year and Total Marks. Create 10 students and store them in a file.
16. Write a program that opens an existing text file and copies it to a new text file with all lowercase letters changed to capital letters and all other characters unchanged.

**KAKATIYA UNIVERSITY**  
**FACULTY OF SCIENCE**  
**B.Sc. (Computer Science)**  
**SEMESTER – II**  
**Programming in C++**

Theory	4 Hours/Week	4 Credit	Internal marks = 20
Practical	3 Hours/Week	1 Credit	External Marks = 80

**Unit – I**

Introduction to C++: Applications, Example Programs, Tokens, Data Types, Operators, Expressions, Control Structures, Arrays, Strings, Pointers, Searching and Sorting Arrays.

Functions: Introduction, Prototype, Passing Data by Value, Reference Variables, Using Reference Variables as Parameters, Inline Functions, Default Arguments, Overloading Functions, Passing Arrays to Functions.

Object Oriented Programming: Procedural and Object-Oriented Programming, Terminology, Benefits, OOP Languages, and OOP Applications.

**Unit – II**

Classes: Introduction, Defining an Instance of a Class, Why Have Private Members? Separating Class Specification from Implementation, Inline Member Functions, Constructors, Passing Arguments to Constructors, Destructors, Overloading Constructors, Private Member Functions, Arrays of Objects, Instance and Static Members, Friends of Classes, Member-wise Assignment, Copy Constructors, Operator Overloading, Object Conversion, Aggregation.

**Unit – III**

Inheritance: Introduction, Protected Members and Class Access, Base Class Access Specification, Constructors and Destructors in Base and Derived Classes, Class Hierarchies, Polymorphism-Function Overloading, Function Overriding and Virtual Member Functions, Abstract Base Classes and Pure Virtual Functions, Multiple Inheritance.

C++ Streams: Stream Classes, Unformatted I/O Operations, Formatted I/O Operations.

**Unit – IV**

Exceptions: Introduction, Throwing an Exception, Handling an Exception, Object-Oriented Exception Handling with Classes, Multiple Exceptions, Extracting Data from the Exception Class, Re-throwing an Exception.

Templates: Function Templates–Introduction, Function Templates with Multiple Type, Overloading with Function Templates, Class Templates – Introduction, Defining Objects of the Class Template, Class Templates and Inheritance, , Introduction to the STL.

**Textbook:** Tony Gaddis, Starting out with C++: from control structures through objects (7e)

**References:**

1. B. Lippman, C++ Primer
2. Bruce Eckel, Thinking in C++
3. K.R. Venugopal, Mastering C++
4. Herbert Schildt, C++: The Complete Reference
5. Bjarne Stroustrup, The C++ Programming Language
6. Sourav Sahay, Object Oriented Programming with C++TEXT BOOK:
7. Object Oriented Programming with C++ Sixth edition, E.Balaguruswamy.
8. A Structured Approach Using C++ By B.A.Forouzan & Rf Gilberg (Thomson Business Information India)
9. Herbert Schilbt, C++ - The Complete Reference, TMH 2002
10. J.P. Cohoon and J.W. Davidson, C++ program design – An Introduction To Programming and Object Oriented Design.- MGH 1999.

**KAKATIYA UNIVERSITY**  
**FACULTY OF SCIENCE**  
**B.Sc. (Computer Science)**  
**SEMESTER – II**  
**Programming in C++ Lab**

Practical      3 Hours/Week      1 Credit      Marks: 50

**Note:**

- Programs of all the Concepts from Text Book including exercises must be practice and execute.
  - Faculty must take care about UG Standard Programs.
  - In the external lab examination student has to execute two programs with compilation and deployment steps are necessary. Write the Pseudo Code and draw Flow Chart for the programs.
  - Recommended to use Open Source Software: GCC on Linux; DevC++ (or) CodeBlocks on Windows10.
  - External Vice-Voce is compulsory.
1. Write a program to.
    - a. Print the sum of digits of a given number.
    - b. Check whether the given number is Armstrong or not
    - c. Print the prime number from 2 to n where n is natural number given.
  2. Write a program to find largest and smallest elements in a given list of numbers and sort the given list.
  3. Write a program to read the student name, roll no, marks and display the same using class and object.
  4. Write a program to implement the dynamic memory allocation and de-allocation using new and delete operators using class and object.
  5. Write a program to find area of a rectangle, circle, and square using constructors.
  6. Write a program to implement copy constructor.
  7. Write a program using friend functions and friend class.
  8. Write a program to implement constructors
    - a. Default Constructor, Parameterized Constructor, Copy Constructor
    - b. Define the constructor inside/outside of the class
    - c. Implement all three constructors within a single class as well as use multiple classes( individual classes)
  9. Write a program to implement the following concepts using class and object
    - a. Function overloading
    - b. Operator overloading (unary/binary(+ and -))
  10. Write a program to demonstrate single inheritance, multilevel inheritance and multiple inheritances.
  11. Write a program to implement the overloaded constructors in inheritance.
  12. Write a program to implement the polymorphism and the following concepts using class and object.
    - a. Virtual functions
    - b. Pure virtual functions
  13. Write a program to implement the virtual concepts for following concepts
    - a. Constructor (not applied)
    - b. Destructor (applied)
  14. Write a program to demonstrate static polymorphism using method overloading.
  15. Write a program to demonstrate dynamic polymorphism using method overriding and dynamic method dispatch.
  16. Write a program to implement the template (generic) concepts
    - a. Without template class and object
    - b. With template class and object

**KAKATIYA UNIVERSITY**  
**FACULTY OF SCIENCE**  
**B.Sc. (Computer Science)**

**Model Question Paper**

3 Hours

Max Marks -80 Credits -4

**PART -A**                    **Answer any eight questions in part –A    8X4 M = 32 Marks**

UNIT- I    1  
              2  
              3

UNIT- II    4  
              5  
              6

UNIT- III    7  
              8  
              9

UNIT- IV    10  
              11  
              12

**Part – B**                    **Answer all Questions 12MX4 = 48 Marks**

UNIT- I    13  
              Or  
              14

UNIT- II    15  
              Or  
              16

UNIT- III    17  
              Or  
              18

UNIT- IV    19  
              Or  
              20

**KAKATIYA UNIVERSITY**  
**FACULTY OF SCIENCE**  
**B.Sc. (Computer Science)**

**Practical Question Paper**

3 Hours

Max Marks -50

Credits -1

Answer any Two

15MX2 = 30 MARKS

UNIT – I	1 Program
UNIT- II	1 Program
UNIT-III	1 Program
UNIT -IV	1 Program

Viva - 10 Marks

Record – 10 Marks