Kadi Sarva Vishwavidyalaya

Faculty of Engineering & Technology

First Year Bachelor of Engineering (CE / IT /EC)

(With Effect From: Academic Year 2017-18)

Subject Code: CC111-N	Subject Title: OBJECT ORIENTED PROGRAMMING USING 'C++'

Teaching scheme				Total Credit	Evaluation Scheme					
L	T	P	Total		Theory		Mid Sem Exam	CIA	Pract.	Total
Hrs	Hrs	Hrs	Hrs		Hrs	Marks	Marks	Marks	Marks	Marks
03	00	02	05	04	03	70	30	20	30	150

Course Objective:

- To make the students think in the direction of providing alternative option to procedural programming languages
- To understand the importance of data over process
- To learn the concepts of Object Oriented Programming (OOP)
- To understand the real word issues and map them using OOP
- To learn the syntax and semantics of C++ programming language

Outline of the Course:

Sr. No	Title of the Unit	Minimum Hour
1	Elaborated understanding of Essentials C Programming	6
2	Fundamental Concepts of OOP with C++	4
3	C++ Programming Syntactical Basics	4
4	C++ Functions	6
5	Objects and Class	6
6	Operator Overloading	5
7	Inheritance	5
8	Polymorphism & Virtual Functions	5
9	Templates and Exception Handling	5
10	Introduction to Streams and Files	2

Total hours (Theory): 48
Total hours (Lab): 32
Total hours: 80

Kadi Sarva Vishwavidyalaya

Faculty of Engineering & Technology

First Year Bachelor of Engineering (CE / IT /EC)

(With Effect From: Academic Year 2017-18)

Detailed Syllabus:

Sr. No	Topic	Lecture Hours	Weightage(%)
1	Elaborated understanding of Essentials C Programming: Structure, Pointers, File Management.	6	**13**
2	**Fundamental Concepts of OOP with C++:** Procedural Vs. OOP Approach, Basics concepts of OOP. Understanding the terminology: Objects, Classes, Inheritance, Reusability, Polymorphism, and Overloading. Benefits and Applications of OOP.	4	**8**
3	**C++ Programming Syntactical Basics**: Input/output statements, Directives, Comments, Manipulators, Type conversion, Library Functions.	4	**8**
4	**C++ Functions**: Functions with objects, Overloaded functions, Inline functions.	6	**13**
5	**Objects and Class**: Introduction and definition, Access specifies (private, protected and public) and their scope, Constructors, Destructors, Objects as function arguments, Friend function, *this* pointer.	6	**13**
6	**Operator Overloading**: Overloading unary operator, Overloading binary operator, Data conversion, Pitfalls of Operator Overloading and Conversion.	5	**10**
7	**Inheritance**: Concepts of base class and derived class, Overriding member functions, Scope resolution and overridden functions, Abstract class, Various categories of Inheritance.	5	**10**
8	**Polymorphism & Virtual Functions**: Virtual and Pure virtual function, Late binding, Polymorphism implementation using examples.	5	**10**
9	**Templates and Exception:** Function templates, class templates, Exceptions handling	5	**10**
10	**Introduction to Streams and Files:** Stream classes, Stream errors, File I/O with stream.	2	**5**
	Total	**48**	**100**

Instructional Method and Pedagogy:

- At the start of course, the course delivery pattern, prerequisite of the subject will be discussed.
- Lectures will be conducted with the aid of multi-media projector, black board, OHP etc.
- Attendance is compulsory in lecture and laboratory which carries 10 marks in overall evaluation.
- One internal exam will be conducted as a part of internal theory evaluation.
- Assignments based on the course content will be given to the students for each unit and will be evaluated at regular interval evaluation.
- Surprise tests/Quizzes/Seminar/tutorial will be conducted having a share of five marks in the overall internal evaluation.
- The course includes a laboratory, where students have an opportunity to build an appreciation for the concepts being taught in lectures.
- Experiments shall be performed in the laboratory related to course contents.

Kadi Sarva Vishwavidyalaya

Faculty of Engineering & Technology

First Year Bachelor of Engineering (CE / IT /EC)

(With Effect From: Academic Year 2017-18)

Learning Outcome:

At the end of this course, the student would be able

- To differentiate the ideas and utilizations of Procedural languages and Object Oriented Programming
- Learn significance of data over the process
- Apply the concepts of OOP to solve real time applications
- Code the OOP using C++ syntax.

TEXT BOOKS:

1. Object Oriented Programming in Turbo C++, Rahul Lafore, Galgotia
2. Object Oriented Programming With C++, E Balagurusamy, TMH
3. C++ : How to Program, 9th Edition, Deitel and Deitel, PHI
4. Object Oriented Programming with ANSI and Turbo C++, Ashok Kamthane, Pearson

REFERENCE BOOKS:

1. The Compete Reference C++, Herbert Schlitz, TMH
2. C++ Programming, Black Book, Steven Holzner, dreamtech

List of experiments (Not limited to following. Subject teacher may modify the same)**:**

Sr.No	List of Experimentation
0	**C-Programming** • Write a program to create an employee structure having member's name, salary, Get data in employee structure through one function and display data using another function. Use concept of structure and function. • Write a program to find length of string using pointer and without using string functions. • Write a program that creates the structure of student and Scan the data of n students and store this information into the file. Again read the data of n students from file and display on the standard output. • Write a program that copies the contents of one file into another. • Write a program that appends the content of file at the end of the other.
1	Write a function power to raise a number m to power n. The function takes a double value for m and int value for n. Use default value for n to make the function to calculate squares when this argument is omitted
2	Write a program to perform addition of two complex numbers using constructor overloading. The first constructor which takes no argument is used to create objects which are not initialized, second which takes one argument is used to initialize real and imaginary parts to equal values and third which takes two argument is used to initialized real and imaginary to two different values
3	Create a class TIME with members hours, minutes, and seconds. Read values from keyboard and add two TIME objects (hint: by passing objects to function) and display result.

4	Declare a class Account to represent bank account of customers with the following data members. Name of the depositor, account no., type of account(S for saving and C for current), balance amount. Write a program using above class to do the following: • To initialize data members. • To deposit money. • To withdraw money after checking the balance (minimum is Rs.1000) • To display the Details of particular account.
5	Imagine a publishing company that markets both book and audiocassette versions of its works. Create a class publication that stores the title (a string) and price (type float) of a publication. From this class derive two classes: book, which adds a page count (type int), and tape, which adds a playing time in minutes (type float). Each of these three classes should have a getdata() function to get its data from the user at the keyboard, and a putdata() function to display its data.Write amain() program to test the book and tape classes by creating instances of them, asking the user to fill in data with getdata(), and then displaying the data with putdata().
6	Rahul is starting a new cosmetic and clothing business and would like to make a net profit of approximately 10% after paying all the expenses, which include merchandise cost, store rent, employees' salary, and electricity cost for the store. He would like to know how much the merchandise should be marked up so that after paying all the expenses at the end of the year he gets approximately 10% net profit on the merchandise cost. Note that after marking up the price of an item he would like to put the item on 15% sale. Write a program that prompts Rahul to enter the total cost of the merchandise, the salary of the employees (including his own salary), the yearly rent, and the estimated electricity cost. The program then outputs how much the merchandise should be marked up so that Rahul gets the desired profit.
7	A book shop maintains the inventory of books that are being sold at the shop. The list includes details such as author, title, price, publisher and stock position. Whenever a customer wants a book, the sales person inputs the title and author and the system searches the list and displays whether it is available or not. If it is not, an appropriate message is displayed. If it is, then the system displays the book details and requests for the number of copies required. If the requested copies book details and requests for the number of copies required. If the requested copies are available, the total cost of the requested copies is displayed; otherwise the message "Required copies not in stock" is displayed. Design a system using a class called books with suitable member functions and Constructors. Use new operator in constructors to allocate memory space required.
8	Implement a class string containing the following functions: • Overload + operator to carry out the concatenation of strings. • Overload = operator to carry out string copy. • Overload <= operator to carry out the comparison of strings. • Function to display the length of a string. • Function tolower( ) to convert upper case letters to lower case. • Function toupper( ) to convert lower case letters to upper case.
9	Write a C++ Program to implement following inheritance structure. Write input/output functions to test your code.
10	Design three classes STUDENT, EXAM and RESULT. The STUDENT class has data members such as rollno, name. EXAM is created by inheriting STUDENT. EXAM class adds data members representing the marks scored in six subjects. Derive RESULT from EXAM and has its own data members such as total marks. Write a program to model this relationship.

11	Create a base class called SHAPE. Use this class to store two double type values. Derive two specific classes called TRIANGLE and RECTANGLE from the base class. Add to the base class, a member function *getdata* to initialize base class data members and another member function *display* to compute and display the area of figures. Make *display* a virtual function and redefine this function in the derived classes to suit their requirements. Using these three classes design a program that will accept driven of a TRINGLE or RECTANGLE interactively and display the area.
12	Create a function using the concept pointers that swaps the private data values of two objects of the same class type.
13	Write a program to find the larger of two given numbers in two different classes using friend function.
14	Create a class called LIST with two pure virtual function store() and retrieve().To store a value call store and to retrieve call retrieve function. Derive two classes stack and queue from it and override store and retrieve.
15	Write a program to define the function template for calculating the square of given numbers with different data types.