Kadi Sarva Vishwavidyalaya

Master of Computer Application (MCA)

Year – I (Semester – II) (W.E.F. August 2014)

Subject Name: Object Oriented Concepts and Programming – MCA 203

Sub Total Credit	Teaching scheme		Examination scheme				
	(per week)		MID	CEC	External		Total Marks
	Th	Pr	Th	Th	Th.	Pr.	
5	3	4	25	25	50	50	150

Rationale (Course Objective) :

- To introduce Object oriented concepts and programming so that the student can work on any object oriented language in the future.
- To give hands on knowledge of visual object oriented programming.

Learning Outcome:

Students will learn the object oriented concepts. They will understand the different features of C++. They would be able to model real world problems through C++ programming.

Instructional Strategies:

To fulfill the aim of the subject, theory as well as practical sessions will be conducted. The act of learning can be improved by using audio-visual aids like OHP and LCD Projector. Apart from these regular seminars and case studies will also be conducted.

Course Content:

UNIT – 1 Object Oriented Concepts [20%]

Object Oriented Development; Objects and Classes; Generalization and Inheritance; Polymorphism and Virtual Functions

UNIT – 2 Classes, Constructors and Destructors [20%]

A Simple Class; Objects as Physical Objects & as Data Types; Constructors; Objects as Function Arguments; returning Objects from Functions; Arrays of Objects.

UNIT – 3 Operator Overloading and Inheritance [20%]

Overloading Unary Operators; Overloading Binary operators; Data conversion; Inheritance: Derived Class and Base Class; Derived Class Constructors; Overriding Member Functions; Public and Private Inheritance; Levels of Inheritance; Multiple Inheritance; Containership: Classes within Classes.

UNIT – 4 Pointers, Virtual Functions and File Handling [20%]

Addresses and Pointers; Pointers and Arrays; Pointers and Functions; Pointers and Strings; Memory Management : new and delete; Pointers to Objects; Pointers to Pointers; Virtual Function; Friend Functions; Static Functions; The this Pointer; Streams; String I/O; character I/O; Object I/O; I/O with Multiple Objects; File Pointers; Disk I/O with Member Functions; Multi File Programs.

UNIT – 5 Templates, Exception Handling and Namespaces [20%]

Generic Classes, Creation of Generic Classes, Multiple Arguments with Template, Generic Functions, Multiple Arguments with Generic Functions, Overloading of Template Functions, Exception Handling Mechanism, Multiple Catch Handler, Creation of Namespaces, Nesting of Namespaces, RTTI usage.

Text Books:
1. Object Oriented Programming with C++ by E. Balagurusamy

Chapter – 1.2,3,4,5,6,7,8,9,10,11,12.13,15 (Full Chapters)

Reference Books:
1. Object Oriented Programming in Turbo C++ by Robert Lafore (Galgotia - 1994)
2. Complete Reference C++ by Herbert Schildt - forth edition

Practical: Any 'C++' compiler will be used for practical programs

List of Practicals:

1. Write a C++ program that will ask for a temperature in Fahrenheit and display it in Celsius.
2. Define a class for Student. Enter data - roll no., name, age, semester and percentage for **five** students and display it.
3. Write a program to store values in one array, sort it and store it in another array. Display both the arrays. (Write a function for sorting)
4. Construct m x n matrix using class and find the (1, 2) element of the matrix. (Take the choice from user to display the element).
5. Write a program using a friend function to add two values defined in different classes.
6. Write a program using a friend function to exchange two values defined in different classes.
7. Write down a class shape which has three member functions with same name which calculates the area of three shapes.

    a. Triangle – ½ x l x b

    b. Rectangle – l x b

    c. Circle – 3.14 x r2

    (Use function over loading)
8. Write a program to add and subtract two complex numbers. (Use all type of constructors and use destructor to destroy them.)
9. 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 available then system displays the book details otherwise the message – "Book not available." Design a class ***book*** with suitable member functions and constructors. Use ***new*** operator to allocate memory space required.
10. Extend the above program. If the book is available then system displays the book details and requests for the number of copies required. If the required copies are available, the total cost of copies is displayed otherwise the message – "Requested copies not in stock."
11. Write a C++ program to overload '>=' operator using member functions. Also overload '*' operator using only friend function.
12. Create a Class – '**Student**' with appropriate data members and member functions. Derive the classes –

    '**Test**' PUBLICLY from the above class. Derive another classes – '**Result**' PRIVATELY from the

‘**Test**’ class. Enter data of students. Also enter data for Test conducted.

a. Ask the name of the Student from the user and display his / her result.

b. Display the records of all the students who failed in the test.

Use File Management for following programs:-

13. Write a program that stores all ASCII values in file and display it on the screen.
14. Create a class inventory with data member name, code and cost. Store this data in file and display it on console. Use manipulators like ‘setw’ and ‘setprecision’ for displaying data.
15. Write a program for implementing Employee Management System through the use of Virtual Functions. Create class Company, Employee and Salary. Use functions – **show( )** for displaying data on console and **search( )** to search the details of a particular employee. The program should be menu driven.
16. Write a program for implementing Airline Reservation System through the use of Virtual Functions.
17. Create class **Airport**, **Flight** and **Passengers**. Use functions – **show( )** for displaying data on console and **search( )** to search the details of a particular Passenger. The program should be menu driven.
18. Write a program for generic function with specific type of arguments to calculate Addition, Subtraction and Multiplication of the data.
19. Write a class template to represent a generic vector. Include member functions to perform the following tasks:-
    1. To create the vector
    2. To modify the value of a given element
    3. To multiply by a scalar value
    4. To display the vector in the form (10, 20, 30, ...)
20. Write a program to generate different type of Exceptions when checking the value.
21. Write a program with the following:-
    1. A function to read two double type members from keyboard.
    2. A function to calculate the division of these two numbers.
    3. A try block to throw an exception when a wrong type of data is keyed in.
    4. A try block to detect and throw an exception if the condition “divide – by – zero” occurs.
    5. Appropriate catch blocks to handle the exceptions thrown.
22. Define a class Person with data members as name of the person, names of parents of the person, gender, age, and an array-containing list of interests. Provide member functions FindFather, FindMother, FindUncle and FindAunty functions, all of which returns the object of person class. Provide access using function pointer for all these functions. Use Exception Handling techniques to handle errors.
23. For a supermarket, define a bill class. All the bill objects will contain bill number, name of clerk preparing the bill, each item with quantity and price, total amount to be paid. Total items in the bill vary. Define dynamic memory allocation constructor for bill class such that any number of items from 1 to 50 can be accommodated in a single bill. There is an array describing each item with price. Price is to be picked up from that array. Now overload = operator and provide reasons for need of such operator. Use Exception Handling techniques to handle errors.
