Q.6 Attempt any two:

i. Explain the process of opening a file in a programming language of your choice. Describe the steps involved and the different modes in which a file can be opened. Provide a suitable example that demonstrates opening a file, writing data to it, reading data from it, and closing the file. **5**

ii. Discuss the concept of file stream operations in programming. Explain how file stream operations allow reading from and writing to files with example. **5**

iii. Explain how command line arguments affect file handling operations. Give a suitable example that demonstrates the use of command line arguments in file handling. **5**

******

Total No. of Questions: 6 *Total No. of Printed Pages:4*

Enrollment No......................................

Faculty of Science / Engineering

End Sem Examination May-2024

CA3CO05 Object Oriented Programming

Programme: BCA / BCA-MCA (Integrated) Branch/Specialisation: Computer Application

Duration: 3 Hrs. **Maximum Marks: 60**

Note: All questions are compulsory. Internal choices, if any, are indicated. Answers of Q.1 (MCQs) should be written in full instead of only a, b, c or d. Assume suitable data if necessary. Notations and symbols have their usual meaning.

Q.1 i. In object-oriented programming, which of the following best characterises an object? **1**

(a) An instance of a class that contains data and behaviours

(b) A blueprint for constructing objects with characteristics and methods

(c) A single-valued variable, like an integer or string

(d) A data structure for storing numerous values of the same type

ii. What is the main function of data encapsulation in OOP? **1**

(a) Allow numerous classes to inherit from a common base class.

(b) Allowing an object to access methods from another object.

(c) To conceal an object's implementation details.

(d) Defining an object's structure.

iii. Which of the following is a preprocessor directive in programming languages? **1**

(a) import (b) require (c) include (d) load

iv. Which statement is used to terminate the current iteration of a loop and begin the next iteration? **1**

(a) break (b) continue (c) exit (d) next

v. Which of the following is an example of a call by reference in programming? **1**

(a) Passing a value to a function

(b) Passing a pointer to a function

(c) Passing the address of a variable to a function

(d) Passing a copy of a variable to a function

vi. What does the "main" function typically represent? **1**

(a) The starting point of program execution

(b) A function used for mathematical calculations

(c) A function that is called when an error occurs

(d) A function used for string manipulation

vii. Which of the following best describes inheritance in object-oriented programming? **1**

(a) It allows a class to inherit properties and behaviour from another class.

(b) It is used to create multiple instances of a class.

(c) It is used to define the structure of a class.

(d) It allows a class to access private members of another class.

viii. What is the purpose of a virtual function? **1**

(a) To create a function that can be called from any class

(b) To create a function that cannot be overridden in derived classes

(c) To create a function that can be overridden in derived classes

(d) To create a function that is automatically called when an object is created

ix. Which of the following is NOT a file operation? **1**

(a) Opening a file (b) Reading from a file

(c) Writing to a file (d) Compiling a file

x. What is the purpose of exception handling in file operations? **1**

(a) To prevent files from being opened

(b) To handle errors that occur during file operations

(c) To force the program to terminate if a file error occurs

(d) To increase the speed of file operations

Q.2 i. Explain the concept of inheritance in object-oriented programming. **2**

ii. Explain the concept of polymorphism in object-oriented programming and provide an example to illustrate its use. **3**

iii. Discuss the concept of data abstraction in object-oriented programming. Why it is important in software development? Give an example of how data abstraction can be implemented in your preferred programming language. **5**

OR iv. Discuss the concept of static binding in object-oriented programming and compare it with dynamic binding. Illustrate with examples the differences between static and dynamic binding in practice. **5**

Q.3 i. What makes implicit type conversion in computer languages different from explicit type conversion? **2**

ii. Explain the role of variables, data types, and expressions in programming. Provide suitable example for calculating the area of a circle by prompting the user to enter the circle's radius, reading the input, and then calculating and printing the area. **8**

OR iii. Discuss the control structures in programming, including for, while, do-while, if, if-else, and switch statements. Explain the purpose and syntax of each structure and provide examples. **8**

Q.4 i. Explain constructors and destructors in object-oriented programming. Describe types of constructors and their purposes, along with the role of destructors in memory management. **4**

ii. Explain the concept of access modifiers (public, private, protected) in classes. Discuss their significance in encapsulation and data hiding with examples. **6**

OR iii. Discuss the concept of static data members and static member functions in classes. Explain with example their purpose. How they are different from non-static members? **6**

Q.5 i. Explain the concept of function overloading in object-oriented programming. Provide examples to illustrate function overloading in a class. **3**

ii. Create a class hierarchy representing different types of vehicles. Include a base class Vehicle with attributes such as make, model, and year, and methods for setting and getting these attributes. Implement at least two derived classes, such as Car and Motorcycle, each with additional attributes and methods specific to that type of vehicle. Demonstrate the use of these classes by creating instances of Car and Motorcycle, setting their attributes, and calling their methods. **7**

OR iii. Explain the concept of operator overloading in object-oriented programming. Discuss how operator overloading allows operators such as +, -, *, / to be redefined for user-defined classes. Provide examples to illustrate the use of operator overloading in a class. **7**

Marking Scheme
Object Oriented Programming (T) - CA3CO05 (T)

Q.1	i)	(a) An instance of a class that contains data and behaviours.		**1**
	ii)	(c) To conceal an object's implementation details.		**1**
	iii)	(c) include		**1**
	iv)	(b) continue		**1**
	v)	(c) Passing the address of a variable to a function		**1**
	vi)	(a) The starting point of program execution		**1**
	vii)	(a) It allows a class to inherit properties and behaviour from another class.		**1**
	viii)	(c) To create a function that can be overridden in derived classes		**1**
	ix)	(d) Compiling a file		**1**
	x)	(b) To handle errors that occur during file operations		**1**
Q.2	i.	Explanation.	1 Mark	**2**
		Example:	1 Mark	
	ii.	Explanation of Polymorphism	2 Mark	**3**
		Any one Example	1 Mark	
	iii.	Explanation ……… process	2 Marks	**5**
		Explanation of Data …………….. development.	2 Marks	
		Examples.	1 Mark	
OR	iv.	Explanation of ……… binding	2 Marks	**5**
		Comparison of …….binding	2 Marks	
		Examples	1 Mark	
Q.3	i.	Explanation of implicit type conversion	0.5 Mark	**2**
		Example illustrating implicit type conversion	0.5 Mark	
		Explanation of explicit type conversion	0.5 Mark	
		Example illustrating explicit type conversion	0.5 Mark	
	ii.	Explanation of variables, data types, and expressions	4 Marks	**8**
		Examples ……… …………… usage.	4 Marks	
OR	iii.	Explanation of control structures.	4 Marks	**8**
		Examples illustrating control structures	2 Marks	
		Explanation of jump statements	2 Marks	
Q.4	i.	Explanation of constructors and their types	1 Mark	**3**
		Types	1 Mark	
		Example	1 Mark	
	ii.	Explanation of access modifiers	3 Marks	**7**
		encapsulation and data hiding	2 Marks	
		Examples each access modifier	2 Marks	
OR	iii.	Explanation of …….. functions	3 Mark	**7**
		Difference	2 Marks	
		Examples	1 Mark	
Q.5	i.	Explanation of function overloading	2 Marks	**4**
		Examples	2 Marks	
	ii.	Design of class hierarchy	2 Marks	**6**
		Implementation of base class, derived class	2 Marks	
		Demonstrates proper inheritance from the base class	2 Marks	
OR	iii.	Explanation of operator overloading	2 Marks	**6**
		Discussion	2 Marks	
		Example	2 Marks	
Q.6				
	i.	Explanation	2 Marks	**5**
		Example and code	3 Marks	
	ii.	Explanation	3 Marks	**5**
		Example	2 Marks	
OR	iii.	Explanation	3 Marks	**5**
		Example	2 Marks	

******