Total No. of Questions: 6 *Total No. of Printed Pages:3*

Enrollment No.....

Faculty of Engineering / Science
End Sem Examination Dec 2024
CS3CO30 / BC3CO35 Object Oriented Programming
Programme: B.Tech./B.Sc. Branch/Specialisation: CSE All / Computer Science

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.

			Marks	BL	PO	CO	PSO
Q.1	i.	Which OOP concept allows creating multiple instances of a class? (a) Objects (b) Encapsulation (c) Inheritance (d) Methods	**1**	1	1	01	1
	ii.	In OOP, which feature allows data and functions to be bundled together? (a) Abstraction (b) Encapsulation (c) Inheritance (d) Polymorphism	**1**	1	1	01	1
	iii.	Which type of object is created at runtime? (a) Static (b) Global (c) Local (d) Dynamic	**1**	1	1	02	2
	iv.	What is the primary purpose of attributes in a class? (a) Encapsulate methods (b) Define object properties (c) Perform operations (d) Create inheritance	**1**	1	1	02	2
	v.	Which type of relationship represents "has-a"? (a) Inheritance (b) Association (c) Dependency (d) Aggregation	**1**	1	1	03	2

vi.	What is the main advantage of recursive association in OOP? (a) Simplicity (b) Reusability (c) Modeling self-referential objects (d) Memory optimization	**1**	1	1	03	2
vii.	Which type of polymorphism is achieved through method overloading? (a) Static (b) Dynamic (c) Runtime (d) None of these	**1**	1	1	04	3
viii.	What is the main purpose of virtual functions in OOP? (a) Compile-time binding (b) Dynamic binding (c) Abstract classes (d) Constructor chaining	**1**	1	1	04	3
ix.	Which container type allows heterogeneous object storage? (a) Array (b) Vector (c) List (d) Map	**1**	1	2	05	3
x.	In OOP, streams are primarily used for: (a) Memory management (b) File I/O operations (c) Object instantiation (d) Code execution	**1**	1	2	05	3
Q.2	i.	Define object-oriented programming. What are its key characteristics?	**4**	1	1	01	1
	ii.	Explain the concepts of abstraction, encapsulation and information hiding with examples of each.	**6**	3	3	01	1
OR	iii.	Compare object-oriented programming and procedural programming in terms of abstraction, modularity, reusability, and scalability. Give examples of each.	**6**	2	2	01	1

Q.3	i.	Explain the concept of Meta-class and its role in object instantiation.	**4**	3	3	02	2
	ii.	Explain the concept of object lifetime. How does encapsulation ensure data integrity during an object's lifecycle? Provide detailed examples.	**6**	3	3	02	2
OR	iii.	Differentiate between static objects and dynamic objects in object-oriented programming. Explain their memory allocation and lifetime.	**6**	2	1	02	2
Q.4	i.	Define aggregation in OOP. Explain how it differs from association.	**4**	1	1	03	2
	ii.	Discuss multiplicities and navigability in associations. How are they modeled in OOP?	**6**	3	3	03	2
OR	iii.	Explain the concept of association between objects. Differentiate between unidirectional and bidirectional associations with suitable examples.	**6**	2	1	03	2
Q.5	i.	What is dynamic polymorphism? Explain with examples.	**4**	1	1	04	3
	ii.	Explain the concept of multiple inheritance with a suitable example. Highlight the advantages and challenges of multiple inheritance in OOP.	**6**	3	2	04	3
OR	iii.	Explain the difference between public and protected access specifiers in C++. How does inheritance affect their visibility? Give examples.	**6**	2	1	04	3
Q.6		Attempt any two:					
	i.	Discuss the concept of persistent objects in OOP. How are they managed using streams and files?	**5**	3	3	05	3
	ii.	Explain the purpose and usage of container classes in object-oriented programming. Provide an example to demonstrate their importance.	**5**	2	2	05	3
	iii.	What are heterogeneous containers? Explain their purpose and demonstrate with an example.	**5**	3	3	05	3

******

Marking Scheme

Object Oriented Programming CS3CO30-BC3CO35

Q.1	i)	a) Objects	**1**
	ii)	b) Encapsulation	**1**
	iii)	d) Dynamic	**1**
	iv)	b) Define object properties	**1**
	v)	d) Aggregation	**1**
	vi)	c) Modeling self-referential objects	**1**
	vii)	a) Static	**1**
	viii)	b) Dynamic binding	**1**
	ix)	d) Map	**1**
	x)	b) File I/O operations	**1**
Q.2	i.	Define Object-Oriented Programming. **2 Marks** What are its key characteristics?**2 Marks**	**4**
	ii.	Explain the concepts of Abstraction, Encapsulation and Information Hiding with examples of each. **2 Marks Each**	**6**
	iii.	Compare Object-Oriented Programming and Procedural Programming in terms of abstraction, modularity, reusability, and scalability. Give examples of each. **1.5 marks Each topic**	**6**
	iv.		
Q.3	i.	Explain the concept of Meta-class **2 Marks** and its role in object instantiation. **2 Marks**	**4**
	ii.	Explain the concept of object lifetime. **2 Marks** How does encapsulation ensure data integrity during an object's lifecycle? **2 Marks** Provide detailed examples. **2 Marks**	**6**
OR	iii.	Differentiate between static objects and dynamic objects in Object-Oriented Programming. **3 Marks** Explain their memory allocation and lifetime. **3 Marks**	**6**
Q.4	i.	Define Aggregation in OOP. **2 Marks** Explain how it differs from Association. **2 Marks**	**4**
	ii.	Discuss multiplicities and navigability in associations. 3 **Marks** How are they modeled in OOP? 3 **Marks**	**6**
OR	iii.	Explain the concept of association between objects. **2 Marks** Differentiate between unidirectional and bidirectional associations **2 Marks** with suitable examples. **2 Marks**	**6**
Q.5	i.	What is dynamic polymorphism? **2 Marks** Explain with examples. **2 Marks**	**4**
	ii.	Explain the concept of multiple inheritance with a suitable example. 3 **Marks** Highlight the advantages and challenges of multiple inheritance in OOP. 3 **Marks**	**6**
OR	iii.	Explain the difference between public and protected access specifiers in C++. 2 **Marks** How does inheritance affect their visibility? 2 **Marks** Give examples. 2 **Marks**	**6**
Q.6			
	i.	Discuss the concept of persistent objects in OOP. **2.5 Marks** How are they managed using streams and files? **2.5 Marks**	**5**
	ii.	Explain the purpose and usage of container classes in Object-Oriented Programming. **2.5 Marks** Provide an example to demonstrate their importance. **2.5 Marks**	**5**
	iii.	What are heterogeneous containers? **2.5 Marks** Explain their purpose and demonstrate with an example. **2.5 Marks**	**5**

[2] [3]

******

P.T.O.