Printed Pages – 13 Roll No. : .....

322455(22)

B. E. (Fourth Semester) Examination, Nov.-Dec. 2021

(New Scheme)

(CSE Branch)

OBJECT ORIENTED CONCEPTS & PROGRAMMING using C++

Time Allowed : Three hours

Maximum Marks : 80

Minimum Pass Marks : 28

Note : *Attempt all questions. Part (a) of each question is compulsory and carries 2 marks. Attempt any* ***two*** *parts from (b), (c) and (d) which carry 7 marks each.*

Unit-I

1. (a) Explain with neat diagram the 4-step compilation process in C++.

(b) Explain 7 points of comparison between procedure oriented and object oriented programming.

(c) Explain using appropriate code the use of pointer, pointer to array, reference and reference to array.

(d) Following is a 25 line C++ code. Evaluate each statement in the following program and mention your observations for each. Mention errors if found and solution to those errors.

Line No.	C++ code
1.	#include<iostream.h>
2.	#include<conio.h>
3.	main()
4.	{
5.	clrscr();
6.	int a=10;
7.	int b=20;
8.	int c=30;
9.	int* const p=&a;
10.	cout<<"\n*p="<<*p;

11.	*p=100;
12.	cout<<"\na="<<a;
13.	p=&b;
14.	const int *q=&b;
15.	q=&a;
16.	cout<<"\n*q="<<*q;
17.	q=&b;
18.	*q=200;
19.	cout<<"\n*q="<<*q;
20.	const int* const z=&c;
21.	z=&b;
22.	*z=100;
23.	cout<<"\n*z="<<*z;
24.	getch();
25.	}

Unit-II

2. (a) Find the errors in the following code and explain the reason for the errors :

Line No.	C++ code
1.	#include<iostream.h>
2.	#include<conio.h>
3.	class A
4.	{
5.	in i;
6.	public:
7.	int j;
8.	protected:
9.	int k;
10.	private:
11.	int m;
12.	};
13.	void main()
14.	{
15.	clrscr();
16.	A ob;
17.	ob.i=10;
18.	ob.j=20;

19.	ob.k=30;
20.	ob.m=40;
21.	getch();
22.	}

(b) Evaluate the following code and explain the output of every statement in main function with appropriate diagram depicting the contents of the code segment and data segment :

Line No.	C++ code
1.	#include<iostream.h>
2.	#include<conio.h>
3.	class A
4.	{
5.	static int s;
6.	int n;
7.	public:
8.	void fill()
9.	{
10.	cin>>n;

```
11.     }
12.     void show()
13.     {
14.     cout<<"\n s="<<this->s;
15.     cout<<"\n n="<<this->n;
16.     }
17.     static void test()
18.     {
19.     cout<<"\n s="<<s;
20.     }
21.     };
22.     int A::s;
23.     void main()
24.     {
25.     A obl;
26.     ob1.fill();//let input be 100 for n
27.     ob1.show();//o/p: s=0 n=100
28.     A *p=new A();
29.     p->fill();//let input be 200 for n
```



```
30.     p->show();// o/p: s=0 n=200
31.     A::test();// o/p: s=0;
32.     getch();
33.     }
```

(c) Give below are two programs, complete them separately. Both the programs should be able to swap the data member of objects ob1 and ob2. The data members should private. (hint: use friend function/class).

C++ code-1

```
main()
{
clrscr();
A ob1;
B ob2;
ob1.fill(10);
ob2.fill(20);
swapAB(ob1, ob2);
ob1.show();
```

C++ code-2

```
main()
{
clrscr();
A ob1;
B ob2;
ob1.fill(10);
ob2.fill(20);
ob1.swapAB(ob2);
ob1.show();
```

```
ob2.show();                    ob2.show();
getch();                       getch();
}                              }
```

(d) Following program implements a linked list. Complete the following program :

```
void main()
{
clrscr();
Node head; //Enter node data: 10
Node n1; //Enter node data: 20
Node n2; //Enter node data: 30
Node n3;
//Enter node data: 40
head.add(n1); //n1 node gets added to head node
n1.add(n2); //n2 node gets added to n1 node
n2.add(n3); //n3 node gets added to n2 node
head.showAll();//displays data present in every
node 10 20 30 40
getch();
}
```


Unit-III

3. (a) Explain various types of construcrtors in C++.

(b) Analyze the following program code. The statement in line number 20 is printing garbage value. Explain the reason behind this error and complete the code so that the error gets corrected. (Hint : shallow copy/deep copy).

Line No.	C++ code
1.	#include<iostream.h>
2.	#include<conio.h>
3.	class A
4.	{
5.	int *p;
6.	public:
7.	A()
8.	{p=new int(0);}
9.	void setP(){cin>>*p;}
10.	void setP(){cout<<*p;}
11.	void removeP(){delete p;}

```
12.   };
13.   void main()
14.   {
15.   A ob1;
16.   ob1.setP();
17.   ob1.getP();//output: 100
18.   A ob2=ob1;
19.   ob1.removeP();
20.   ob2.getP();
      //output: garbage value;
      //the correct output should be
      100
21.   getch();
22.   }
```

(c) Complete the following program without using global variables and without changing code in the main function. The data members should be private. Justify the reason behind the output.


```
void main()                    The output of the
                               program should be
{                              Created object id: 1
  {                            Created object id: 2
My class ob1, ob2, ob3         Created object id: 3
}                              Destructed object id: 3
getch();                       Destructed object id: 4
}                              Destructed object id: 2
```

(d) Explain using appropriate code the implementation of three types of type casting operations :

(i) Class type to Class type,

(ii) Class type to Basic type and

(iii) Basic type to Class type can be performed

Unit-IV

4. (a) Explain with proper code examples public inheritance, protected inheritance and private inheritance.

(b) Implement a multilevel inheritance structure using three classes A, B, C. Calss A is the grand parent

of Class C. Class A has a public member function test 1(). Arrange the inheritance in such a manner that the member function test 1() becomes protected in Class B and private in Class C. The program should be designed in such a manner that the above arrangement can be proven.

(c) Explain with appropriate code the use of virtual base class and the reason of its use.

(d) What is dynamic binding? Explain with proper code the implementation of dynamic binding using virtual functions.

Unit-V

5. (a) Write down the basic steps for performing disk file operations in C++.

(b) Write a program to store every character typed on the keyboard into a file. When the user presses the ESC key (ASCII code 27) then the program should stop. (hint : use getche() to read characters from keyboard).

(c) What is the need of templates? Explain with proper program code the implementation of function templates and class templates.

(d) What is the need of exception handling? Explain with proper program code the implementation of exception handling in C++.