Name:

Enrolment No:

UNIVERSITY OF PETROLEUM AND ENERGY STUDIES
End Semester Examination, May 2022

Program Name : B. Tech. ECE **Semester : VI**
Course Name : Programming Techniques **Time : 03 hrs**
Course Code : ECEG 3035 **Max. Marks: 100**

SECTION A [5x4]

S. No.		Marks	CO
Q 1.	Choose the correct option: i. Which special symbol is allowed in a function name? a) ! b) | c) * d) _ ii. Which is the only 'left bit shift' operator in C++? a) ?: b) && c) *= d) << iii. Which of the following statement is true about the function func? void func(int x, int y) { x++; ++y; return (x+y); } a) The sum of x and y b) The sum of the decremented value of x and y c) returns a pointer to the sum of the decremented value of x and y d) Compilation Error: return value type does not match the function type	[2+1+1]	CO1
Q 2.	Look at the following code segment and decide which statement(s) is/are correct: int main()	4	CO2

	{
	char m = 4;
	const char n = 5;
	
	const char * p = &n;
	char * const q = &m;
	
	n = 65; // stmt-1
	*p = 17; // stmt-2
	p = &m; // stmt-3
	*q = 84; // stmt-4
	
	return 0;
	}
	
	a. stmt-1
	b. stmt-2
	c. stmt-3
	d. stmt-4		

Q 3.	#include <iostream>
	using namespace std;
	
	inline int SQR(int x)
	{ return x * x; }
	
	int main()
	{
	int a , b, c;
	a = 15, b = 17; b = SQR(x);
	cout << b << endl;
	c = SQR(x++);
	cout << c << endl;
	return 0;
	}
	
	a. 100 121
	b. Compilation Error: invalid function definition
	c. 100 132
	d. Compilation Error: invalid function parameter
		4	CO1

Q 4.	**Guess the expected output:** #include <bits/stdc++.h> using namespace std; class Geeks { public:	4	CO3

```
        int id;

        //Default Constructor
        Geeks()
        {
                cout << "Default Constructor called" << endl;
                id=-1;
        }

        //Parameterized Constructor
        Geeks(int x)
        {
                cout << "Parameterized Constructor called" << endl;
                id=x;
        }
};

int main() {

        // obj1 will call Default Constructor
        Geeks obj1;
        cout << "Geek id is: " <<obj1.id << endl;

        // obj1 will call Parameterized Constructor
        Geeks obj2(21);
        cout << "Geek id is: " <<obj2.id << endl;
        return 0;
}
```

Q5.	What will be the output of the following code?
	
	#include <iostream>
	using namespace std;
	
	class B
	{ int id;
	public: static int count;
	B() { count- -;
	id = count;
	cout << id << " ";
	}
	};
	
	class D : public B
	{ int n;
	public: D(){count++;
	n = count;
	cout << n << " ";
	}
	};
	
	int B::count = 9;
	
	int main()
	{
	B *basePtr = new D[2];
	delete [] basePtr;
	return 0;
	}
		4	CO3

	SECTION B [4x10]		

Q 6.	Consider the following code segment:
	
	class A
	{
	public:
	virtual void f(int) { }
	virtual void g(double) { }
	virtual void d(char) { }
	int h(A *) { }
	};
	
	class B: public A
	{
	public:
		10	CO3

	void f(int) { }
	virtual int h(B *) { }
	};
	
	class C: public B
	{
	public:
	void g(double) { }
	void d(char) { }
	int h(B *) { }
	};
	
	What will be the content of the virtual function table (VFT) for the instance of class C in correct order?
	
	a. C::g(C*const, double)
	C::d(C*const, char)
	C::h(C*const, B*)
	B::f(B*const, int)
	
	b. B::f(B*const, int)
	C::g(C*const, double)
	C::d(C*const, char)
	C::h(C*const, B*)
	
	c. C::d(C*const, char)
	C::h(C*const, B*)
	C::g(C*const, double)
	B::f(B*const, int)
	
	d. B::f(B*const, int)
	C::d(C*const, char)
	C::h(C*const, B*)
	C::g(C*const, double)		

Q 7.	#include<iostream>
	using namespace std;
	
	class pointer
	{
	
	public:
	int eight;
	
	pointer (int six)
	{
	cout << "A constructor is called" << endl;
	eight=six;
	}
	pointer PTR ()
	{
		10	CO2

	cout << "A default constructor is called " << endl;
	}
	
	~ pointer ()
	{
	cout << "Destructing " << eight<< endl;
	}
	
	int add()
	{
	return(eight+six);
	}
	};
	
	int main()
	{
	pointer myobj1(7);
	pointer myobj2;
	
	cout << myobj1.eight << endl;
	cout << "Enter a number : " ;
	
	cin >> myobj2.eight;
	cout << myobj2.add() << endl;
	
	return(0);
	}
	
	Go through the aforementioned code and endeavor to suggest the expected results to be obtained at Output screen.		

Q 8.	Write the prototype/syntax for the following in C++: (a) Ambiguity in Inheritance (b) virtual function (c) new operator (d) Run time binding for an array of size [16] (e) A reference for a variable	10	CO1
Q 9.	Write a program in C++ to implement the friend function 'print()' which is a member of class QWERTY and accesses the private data members a and b of class QWERTY. **OR** Write a program in C++ to illustrate the top-down order of Constructor execution and Bottom-up order of Destructor execution. [**Hint**: Inheritance may help]	10	CO2
SECTION-C [20x2]			

Q 10.			
(a)	Differentiate between the Multiple Inheritance and Multipath Inheritance.	4	CO3

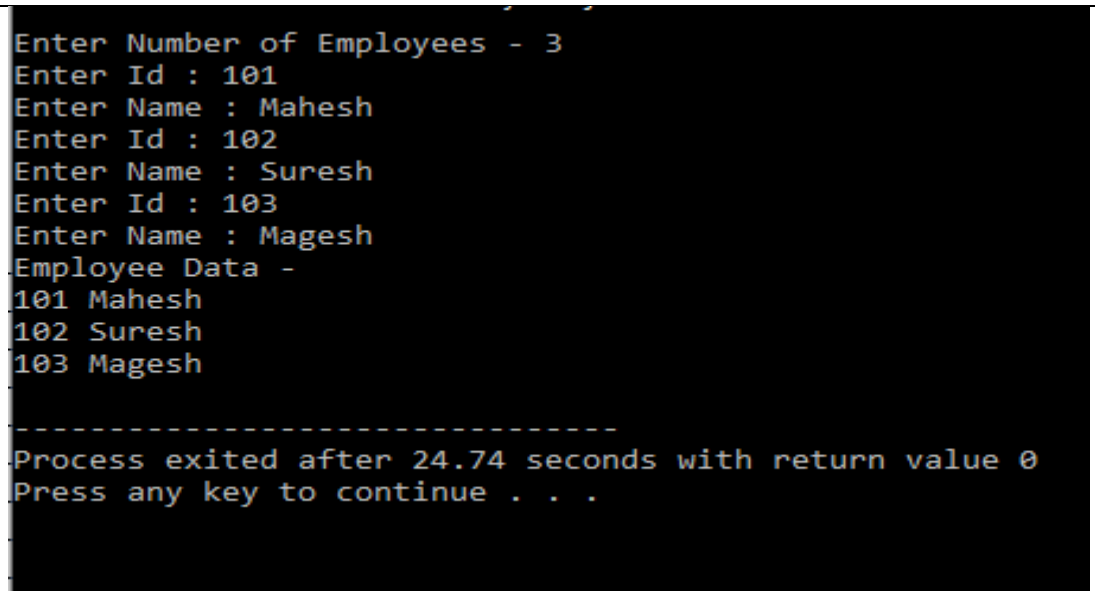
(b)	Define class & object with the help of appropriate example.
	
	OR
	
	This program is all about the implementation of Pre/Post Decrementer. Fill the blank by keeping this in mind so that the given test cases will satisfy:
	
	#include <iostream>
	using namespace std;
	
	class DClass
	
	{ int data;
	public:
	_____{ } **// Define Constructor**
	
	DClass& operator++ ()
	{ ++ data;
	return _____;
	}
	
	_____ {
	DClass t(data);
	++ data;
	return _____; }
	
	void disp()
	{ cout << " " << data ; }
	};
	
	int main()
	{
	int x;
	cin >> x;
	
	DClass obj1(x);
	obj1.disp();
	
	DClass obj2 = obj1- -;
	obj2.disp();
	
	obj2 = - -obj1;
	obj2.disp();
		4	CO1

	return 0; }		
(c)	*A class can be called as polymorphic if it contains:(Multiple Choice Question)* a. Virtual functions of its own b. Virtual functions of base classes c. Pure virtual functions d. Member functions and Virtual functions	4	CO2

(d)	**Go through the following code in C++ and comment upon the problem associated with this code. Also, fetch the expected output:**
	
	#include<iostream>
	
	class ClassA {
	public:
	int a;
	};
	
	class ClassB : public ClassA {
	public:
	int b;
	};
	
	class ClassC : public ClassA {
	public:
	int c;
	};
	
	class ClassD : public ClassB, public ClassC {
	public:
	int d;
	};
	
	int main()
	{
	ClassD obj;
	
	// obj.a = 10; // Statement 1, Error
	// obj.a = 100; // Statement 2, Error
	
	obj.ClassB::a = 10; // Statement 3
	obj.ClassC::a = 100; // Statement 4
	
	obj.b = 20;
	obj.c = 30;
	obj.d = 40;
		8	CO3

	cout << " a from ClassB : " << obj.ClassB::a; cout << "\n a from ClassC : " << obj.ClassC::a; cout << "\n b : " << obj.b; cout << "\n c : " << obj.c; cout << "\n d : " << obj.d << '\n'; }		

Q11. (A)	**Fill in the blanks with one of the following choices:**
	"public/private/protected/not accessible**" against asked comments:**
	
	class A
	{
	public:
	int x;
	protected:
	int y;
	private:
	int z;
	};
	
	class B : public A
	{
	// x is _____
	// y is _____
	// z is _____
	};
	
	class C : protected A
	{
	// x is _____
	// y is _____
	// z is _____
	};
	
	class D : private A // 'private' is default for classes
	{
	// x is _____
	// y is _____
	// z is _____
	};
		10+10	CO3

(B)	 Write a code in C++ using the Array of Objects owing to display the projected output.		