Bangladesh University of Engineering and Technology

Department of Computer Science and Engineering

CSE 107: Object Oriented Programming

Class Test 2, Exam date: 27 May 2025

Time: 25 minutes Full Mark: 20

Student ID: Name:

- √ **There are 10 (ten) questions in this question paper. Answer all of them by writing the minimum codes. Use ellipses (……..) where appropriate.**
- √ **Answer must lie inside the given space. No extra paper sheet can be used.**

Question	Answer

Consider program codes:
class Rectangle{
int length, width;
};
Write down **only one** constructor method that can execute the following code:
Rectangle r1, r2(8), r3(5, 4);
	Rectangle(int m = 0, int n = 0){
	length = m;
	width = n;
	}

Consider the following Java code:
Class Main{
public static void print(Double m, int n){
System.out.println(m + “,” + n);
}
}
Among *m* and *n*, which one is ***call by value*** and which one is ***call by reference***?
	m → call by value (Don’t worry if you wrote call by reference, although it’s incorrect)
	
	n → call by value

Consider following program codes:
void display(string str){
count << str << endl;
}
void display(int m, int n){
count << m+n << endl;
}
Write down necessary program codes to obtain the starting addresses of above two functions in memory location of ***fp*1** and ***fp*2**, respectively.
	void (*fp1)(string);
	void (*fp2)(int, int);
	fp1 = display;
	fp2 = display;

Consider the following Java code inside main method:
Point pointArray = new Point[5];
The class Point has following constructor:
Point(int px, int py) { x = px; y = py;}
Write down an array statement to insert the following points into pointArray. (1, 2), (2, 3), (3, 4), (4, 5), (5, 6).
	for (int i = 0; i < pointArray.length; ++i){
	pointArray[i] = new Point(i+1, i+2);
	}

Consider the following program codes:
class Point{
int x, y;
public:
…….
};
Assume that ***p*** is an object of ***Point*** class. Write own a member method that allows ***p***++.
	Coord operator ++ (int notused) {
	Coord temp = *this;
	x++;
	y++;
	return temp;
	}

Consider the following program codes:
class Rect{
int length, width;
public:
…….
};
int main(){
Rect r = new Rect(10, 5);
double val = r;
cout << val << endl;
}
Write down necessary ***conversion function*** for the above codes.
	operator double(){
	return length * width;
	}

Write down the name of the problem generated under the following scenario:
• class D11 and class D12 inherits class B.
• class D2 inherits class D11 and class D12.
Write down class definitions where you can use ***virtual*** keyword to solve the above problem.
	Diamond problem.
	
	class D11: virtual public B{
	……..
	}
	class D11: virtual public B{
	……..
	}

Normally, an ***interface*** does allow body of any method. Write down the generic **names of two type methods**, whose complete body can be written inside the body of an interface.	Any two from the three below: (1) private; (2) default; and (3) static
Consider the following program code: 1 int main(){ 2 float length = 4.26; 3 int num = static_cast<int> length; 4 double d = static_cast<double> num; 5 cout << typeid(num).name() << endl; 6 double *p = static_cast<double *> (&d); 7 } (i) One of the above statements contains upcasting where explicit casting is not mandatory. Rewrite down only that statement with implicit casting. (ii) Write down the statement number that will generate error?	(i) double d = num; (ii) 6

Consider the following Java Codes:
public class Main{
public static void main(String args[]){
try{
throw new MyException(”User-defined.");
}
catch (MyException e){
System.*out*.println(e.getMessage());
}
}
}
Write down ***MyException*** class that inherits ***Exception*** class.
	class MyException extends Exception {
	public MyException(String str){
	super(str);
	}
	}