

BANGLADESH UNIVERSITY OF PROFESSIONALS
Military Institute of Science and Technology
B.Sc. in Computer Science and Engineering, Term Final (Spring) Examination 2021: July-Aug 2021
Student Group: 36< Earned Credit Hours ≤ 72
Subject: CSE-205, Object Oriented Programming

Total	: 2.00 hours (Online Examination for Section-A + Section-B)	Full Marks	180
Section B	: 1.00 hour (Online Examination)	Section B	90

INSTRUCTIONS - ONLINE EXAMINATION (IF PHYSICAL EXAMINATION IS NOT HELD)

- a. **Question - 5** and **Question – 8 (Viva Voce)** in **Section B** are compulsory.
- b. Answer any **OTHER ONE** question from **Q-6 & Q-7** of **Section-B**.
- c. Figures in the margin indicate full **marks**.
- d. Assume reasonable data if necessary.
- e. **Symbols** and **abbreviations** used have their usual meanings.

SECTION-B

Question – 5 (Compulsory) This Question was set to achieve CO-3
 Take a look at figure 1. It contains 6 boxes representing the details of 6 classes.

Legend

Class Name
Private Members
Protected Members
Public Members

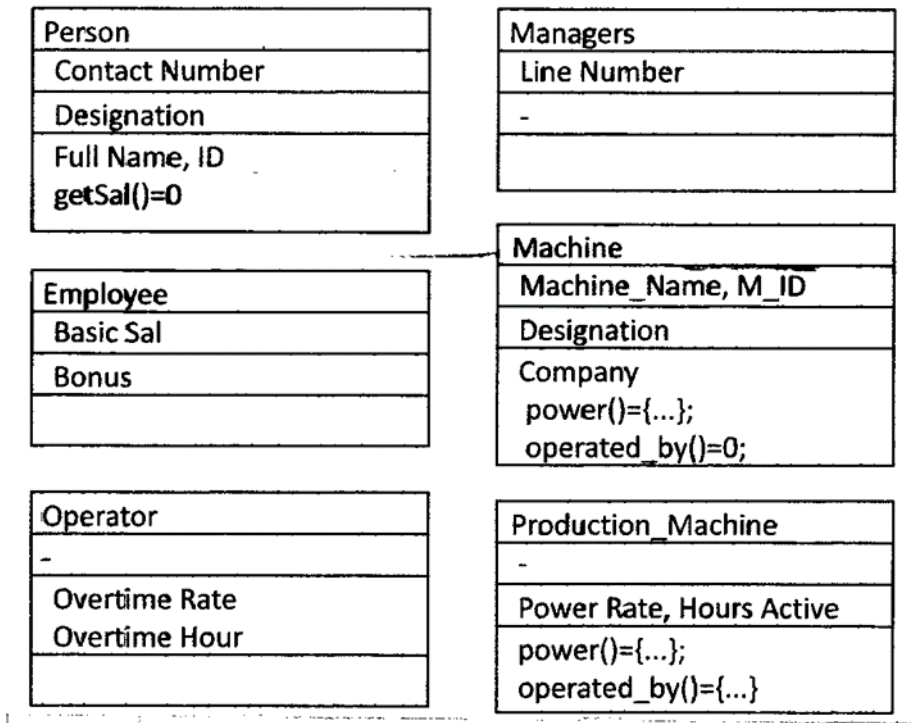


Figure 1

The inheritance relation among these classes is as follows –

- “Employee” is a “Person” (Public inheritance)
- “Operator “ is a Person (Public inheritance)
- “Managers” is an “Employee” (Protected inheritance)
- “Managers” is an “Operator” (Protected inheritance)
- “Production_Machine” is a “Machine” (Public inheritance)
- “Production_Machine” has an “Operator” (Private inheritance)

Now, based on this inheritance scenario, answer question **1(a) – 1(d)**.

- a. Based on the given relation, connect the boxes in figure 1 to complete the inheritance scenario. Only drawing boxes with class name is enough. **10**

Now, in your redrawn diagram, identify the following by analyzing the scenario –

- i) Multilevel inheritance
- ii) Multiple inheritance
- iii) Interface(s), Abstract & Concrete Class(es).

- b. Briefly differentiate between function over-riding and function overloading. Which one do you think is present in the given scenario? **5**

- c. Analyze the scenario and write down the access specification (or visibility) of “getsal()” function in the classes Employee and Manager. **5**

How can you modify the visibility of “getsal()” function in “managers” to public? Determine with code snippet.

- d. Determine which portion of the inheritance scenario causes the “Diamond Problem”. **10**
What can we use to solve it in C++ (no need to write any code to demonstrate)?

Give only pictorial example to demonstrate how it’s implemented internally.

Question – 6

- a. Consider the inheritance scenario in Question 1. Now, take a look at the following code – **15**

```
int main()
{
    Machine M1;
    Production_Machine P1;

    Machine * ptr;

    ptr = &M1;
    ptr-> power();

    ptr = &P1;
    ptr-> power();
}
```

Which specific function will get called when –

- i) Compiler early binds
- ii) Compiler late binds

Use this scenario as an example and explain how late binding is implemented internally in C++.

- b. What is operator overloading? Why do you think it’s an example of the concept of polymorphism? **15**

Now, take a look at the following C++ code and write only the appropriate overloading functions so that the operations in **line 24-34** can be executed as intended.

What the specific operator is intended to do is given as comments in the code. Determine whether to use member or non-member functions in each case.

```

1  #include<iostream>
2  using namespace std;
3
4  class Point
5  {
6  private:
7      int x;
8      int y;
9
10 public:
11     Point(int x, int y)
12     {
13         this->x = x;
14         this->y = y;
15     }
16 };
17 int main()
18 {
19     Point P1(1,2);
20     Point P2(3,4);
21
22     Point p3;
23
24     p3= 5 - p2; ///p3= -2, -1
25
26     P2-=p1; /// p2 = p2-p1, p2= 2,2
27
28     p3!=p2; ///returns true if p3.x!=p2.x and
29     p3.y!=p2.y, false otherwise
30
31     int a = p1; ///a = p1.x+p1.y, so a = 3
32
33     Point * pt = new Point (3,4);
34     ///allocates memory needed for Point(3,4)
35     and prints "Mem Allocated"
36 }

```

Question – 7

- a. What are “Functors” or function objects in operator overloading? Give an example scenario with code snippet. 10
- b. How can we do I/O formatting using the following in C++? You may give short code snippets as examples. 5
 - i) “precision()” function
 - ii) manipulators
- c. Which operators need to be overloaded to create a stream enabled class in C++? Write **only the function signatures** of the overloaded functions for a sample class named “Person”. 5
- d. Assume you want to save the errors generated by “stderr” in a “.txt” file in local storage instead of the standard stream. 10
How can you achieve that? Explain with pictorial example or code snippet.

Question – 8

VIVA VOCE (Compulsory in case of Online Examination)

30