Object Oriented Programming & Design - Monsoon 2023

Midterm Examination Questions

Name of the Student:

Roll Number:

Stream:

Question Structure and Instructions

Please write your roll number **on each page of the question paper** above the top margin. This paper consists of two sections:

1. The 1st section consists of 10 short conceptual questions each carrying 3 points.
2. The 2nd section consists of 4 longer questions each carrying 5 points.

1 Short Conceptual Questions (3x10=30 points)

A1 Which of the above functions have smaller execution time and why?

```
void modify_char(char arr[], int len) {
    for (int i = 0; i < len; i++) { arr[i] += 1; }
}
void modify_int(int arr[], int len) {
    for (int i = 0; i < len; i++) { arr[i] += 1; }
}
```

The first (char) function takes lower time, as char consumes less memory (or has smaller size) than int's. This leads to fewer cache misses in case of char.

(a) 1 mark – char function takes lower time (NO marks should be awarded if this is wrong)

(b) 1 mark – less memory/smaller size

(c) 1 mark – fewer cache misses

A2 Consider two functions that sorts an array of integers and floats respectively. If the arrays are of same length, which of these functions is likely to be faster and why?

The function that sorts integers is going to be faster. This is because (i) integers have smaller lengths, thus leading to fewer cache misses, and (ii) operations on integers take less amount of time than on floats.

(a) 1 mark – integer function takes lower time (NO marks should be awarded if this is wrong)

(b) 1 mark – less memory/smaller size

(c) 1 mark – a single operation on integer takes less amount of time

A3 Let us consider the following code, which designs the class of OOPD and then updates the marks and grade of a student. Is this code properly designed? Justify either way.

```
class OOPDStudent {
    public:
    int marks;
    int grade; };
int main(void) {
    OOPDStudent s;
    s.marks += 10; s.grade += 1;
    return 0;  }
```

No, this code is not properly designed, as it contains all public data members in the class. This violates the principle of data hiding. Furthermore, it updates the object outside the class, thus also violating the principle of encapsulation.

(a) 1 mark – Not correctly designed AND has public data members (NO marks should be awarded if this is wrong)

(b) 1 mark – violates the principle of data hiding

(c) 1 mark – updates the object outside the class / violates encapsulation

A4 The Government of India's Council of Ministers consists of a Prime Minister, a set of Cabinet Ministers and a set of Minister of States. All of them are either members of the Lok Sabha, in which case they represent a constituency, or Rajya Sabha, in which case they represent a state. Furthermore, within the Lok Sabha, there are committees in-charge of various topics. Design an appropriate class structure to represent the above.

```
class House { };
class LokSabha: public House { string constituency; };
class RajyaSabha: public House { string state; //can be int too };

class Minister {
    House *house;
    Committee *committee; //can be string too
};

class CabinetMinister: public Minister { };
class MinisterOfState: public Minister { };
```

(a) Defining at least 4 classes – 1.5 marks (0.75 marks for 2 classes; no marks below that)

(b) Creating the right data members – 0.5 marks

(c) Using inheritance from two base classes – 0.5 marks / inheritance

A5 Suppose you have the following code fragment:

```
class Base {
    public:
    void fun(int a=5) { cout << "Base Function" << a; }
};
class Derived: public Base {
    public:
    void fun() { cout << "Derived Function"; }
};
int main(void) {
    Base *b = NULL;
    b = new Derived; b->fun();
    return 0; }
```

What will be the output of the above code?

Base Function5

(a) Base Function – 1 mark

(b) 5 – 1 mark

(c) No virtual function is used / base class function is invoked – 1 mark

A6 Write the output of the following code:

```
class Base {
        public:
         Base() { cout << "Base Function Constructor"; }
    };
    class Derived: public Base {
        public:
         Derived() { cout << "Derived Function Constructor"; }
        ~Derived() { cout << "Derived Function Destructor"; }
    };
    int main(void) {
        Derived *d = NULL;
        d = new Derived;
        delete d;
        return 0;
    }
```

Base Function ConstructorDerived Function ConstructorDerived Function Destructor

(a) Base Function Constructor – 1 mark

(b) Derived Function Constructor – 1 mark

(c) Derived Function Destructor – 1 mark

(d) -1.5 if sequence is different

(e) -0.5 if line break is used

A7 What will be the output of the following code?

```
class Base {
        public:
        int a;
    };
    class Derived: private Base {
        Derived() { a = 0; cout << a << endl; }
    };
    int main(void) {
        Derived *d = NULL;
        d = new Derived;
        cout << d->a << endl;
        delete d;
        return 0;
    }
```

Compilation error, because 'a' within 'd' is a private data member, as inheritance is private.

Alternatively, because the constructor is private.

(a) Compilation error – 1.5 marks (No marks if this is wrong)

(b) 'a' is private member / private inheritance is used / constructor is private – 1.5 marks

A8 What will be the output of the following code?

File p1.cpp

```
extern int i;
void fun() { cout << i << endl; i+=10; }
```

File p2.cpp

```
int i = 100;
void fun();
int main(void) {
    cout << i << endl;
    fun();
    return 0;
}
```

100

100

The global variable's value is retained and can be accessed from any other file.

(a) First 100 – 1 mark

(b) Second 100 – 1 mark

(c) Explanation about accessing global variable – 1 mark

(d) Note: No marks for claiming compilation error

A9 Is it possible to utilize multiple inheritance in C++, where both the base classes have a public method with the same name and arguments?

Yes, it is possible. The common method needs to be disambiguated by explictly specifying the base class name and using the scope operator.

(a) Yes, possible – 1 mark

(b) Common method can be used – 1 mark

(c) Disambiguation by specifying the base class name and/or scope operator – 1 mark (Note that if this point is specified, so it implicitly includes the 2nd point, so full marks should be given)

A10 Suppose you find that a program runs correctly in most cases, but you are getting a segmentation fault occasionally. How would you diagnose this problem? Explain if any tool can help you with it.

This looks like a case of accessing memory outside allocated region. One tool to identify it is valgrind's memcheck.

(a) Accessing memory outside allocated region – 1.5 marks

(b) Valgrind / memcheck (any one is fine) – 1.5 marks

2 Long Questions (5 x 4 = 20 points)

B1 The Republic of India consists of a number of states and union territories, with each state headed by a governor and a chief minister, whereas each union territory is headed by a Lieutenant Governor. Design a class structure to keep track of the states and union territories, their heads and their neighboring states/union territories.

```
class Region {
    Region *neighbor;
};
class State: public Region {
    string governor;
    string chiefMinister;
};
class UnionTerritory: public Region {
    string ltgovernor;
};
```

(a) 1 mark for each class – 3 marks

(b) 1 mark for the proper data members – 1 mark

(c) 1 mark for proper inheritance – 1 mark

B2 Consider an electronic store keeping track of different types of items. Each item has an associated price, date of manufacturing. The price can be modified by calling a function. Suppose there is a specific type of electronic item called air conditioner with additional characteristics such as tonnage and power rating. How would you model the items?

```
class Item {
    float price; // can be int
    Date dateOfManufacturing;
    void modifyPrice(float price) { this->price = price; }
};
class AirConditioner: public Item {
    int tonnage;
    int powerRating;
};
class UnionTerritory: public Region {
    string ltgovernor;
};
```

(a) 1 mark for each class – 2 marks

(b) 1 mark for the proper data members – 1 mark

(c) 1 mark for proper inheritance – 1 mark

(d) 1 mark for adding modifyPrice function

B3 (i) Consider a situation where you are managing a very large codebase. You find that the values of member variables within the objects are inconsistent. What can you do to improve this situation?

Need to create set functions within the corresponding class, and ensure that the data members are made private. This would allow easier setting of variables.

(a) Creation of set function – 1 mark

(b) Data members are made private – 1 mark

(ii) Now consider a situation where you find that the RAM utilization of program keeps increasing even if you are not allocating a lot of new memory. The objects in this program are often dynamically allocated. How can you improve this situation?

This is a probable case of memory leak. This needs to be handled by using an appropriate delete operator. Can be checked using valgrind/memchecker.

(a) Memory leak – 1 mark

(b) Need to use appropriate delete operator – 1 mark

(c) Checked using valgrind/memchecker – 1 mark

B4 (i) What will be the output of the following program? Justify.

```
int main(void) {
    int arr[10] = {0, 0, 0, 0, 0, 0, 0, 0, 0, 0};
    arr[10] = 5; cout << arr << endl;
    return 0;
}
```

This can give two possible outputs – segmentation fault and/or the location of the array.

(a) segmentation fault – 1 mark

(b) location of array – 1 mark

(ii) Which among the following is faster and why?

```
File prog.py
    def __main__:
        arr = [0] * 10
        for i in range(1, 10):
            arr[i] += arr[i-1] + i

File prog.cpp
    int main(void) {
        int arr[10] = {0, 0, 0, 0, 0, 0, 0, 0, 0, 0};
        for (int i = 1; i < 10; i++) { arr[i] += arr[i-1] + i; }
        return 0;
    }
```

prog.cpp is faster because (i) it is a compiled language, and (ii) it has no garbage collection.

(a) 1.5 marks for each reason – 3 marks