Jaypee Institute Of Information Technology

Test-1 Examination- 2020-2021
B.Tech ., Odd Semester
Course Title: Data Structures
Course Code: 15B11CI311
Maximum Marks: 20
Maximum Time: 01 hr + (10 minutes for uploading)

* Required

1. Email address *

2. Name *

3. Enrollment Number *

4. Batch *

Mark only one oval.

- B1
- B2
- B3
- B4
- B5
- B6
- B7
- B8
- B9
- B10
- B13

5. Question 1

Q1) [1 mark] [CO2]	Fig 1

Write a one line instruction in main() to create 5 objects of class T1_Questions (defined in Fig 1) as per following requirement:
 	
For first three objects parameterized constructor should be invoked, for fourth object default constructor should be invoked, and fifth object should be a copy of the first object and hence invokes copy constructor.
 	
int main ()
{
// One line Instruction
}
	class T1_Questions
	{ int marks;
	int co;
	public:
	T1_Questions()
	{ marks=co=0;
	}
	T1_Questions(int m, int c)
	{ marks=m;
	co=c;
	}
	T1_Questions(T1_Questions &obj)
	{ marks=obj.marks;
	co=obj.co;
	}
	};

6. Question 2 & 3

Consider the code shown in Fig 2. When this code is run, compiler throws an error. Answer Q2) and Q3) based on this code:	Fig 2

Q2) [1 mark][CO2] What might be reason for the error? Justify in your own words (and not compilers words)
 	
Q3) [1 mark][CO2] How can we create a smallerbox inside a box object?
	class Box
	{ int length;
	int width;
	Box smallerbox;
	public:
	Box()
	{ cout<<"size of this object is "<<sizeof(*this);
	}
	};
	int main()
	{
	Box box1;
	}

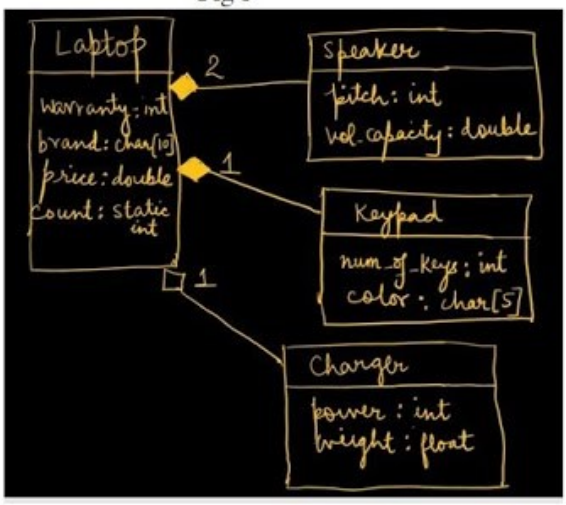
7. Question 4

Q4) [1 mark] [CO2] Justify the Truth or False claim of the following statement with a very small example.
Statement: "In composition, order of component constructor call depends on the order of initialization in composite class constructor."

8. Question 5

Q5) [1 mark] [CO2] Assume that the size of a double variable is d bytes, size of integer variable is i bytes, size of a char variable is c bytes and size of a float variable is f bytes. For the class diagram given in Fig 3, what would be the size of an object of Laptop class. (Assume no alignment is done by compiler).

Fig 3



9. Question 6

Q6) [1 mark] [CO2] Identify and explain the error in following code (Fig 4).

Fig 4

```
class class1
{ public:
   virtual void fun() = 0;
};
class class2 : public class1
 { int x;
 };
int main()
{ class2 obj;
   return 0;
}
```

10. Question 7

Q7) [1 mark] [CO2] Identify the error in following code(Fig 5). How can it be resolved?

Fig 5

```
template<class P = int, class Q, class R = int, class S>
class DS2020
{
   P p;
   Q q;
   R r;
   S s;
};
```

```
int main()
{
   DS2020<char, char> obj1;
   DS2020<int, int, int, int > obj2;
   cout<<sizeof(obj1)<<endl;
   cout<<sizeof(obj2)<<endl;
   return 0;
}
```

11. Question 8 and Question 9

Consider the code given in Figure 6 and answer Q8) and Q9) based on it.

Q8) [1 mark] Predict the output of the given code. (with one-two line explanation)

Q9) [1 mark] Ybu have to modify the given code by changing only one or two statements in such a way that the output of the given code comes out to be: 12341213-4-3-2-1-3-1-2-1 (Note: You can't add or remove any *cout* statement. Just specify the line numbers in which you are making any modification.)

Fig 6

```
1.  class One {
2.  public:
3.  One()
4.  {  cout<<"1";
5.  }
6.  ~One()
7.  {  cout<<"-1";
8.  }
9.  };
10. class Two: virtual public One {
11. public:
12. Two()
13. {   cout<<"2";
14. }
15. ~Two()
16. {   cout<<"-2";
17. }
18. };
19. class Three: virtual public One {
20. public:
21. Three()
22. {   cout<<"3";
23. }
24. ~Three()
25. {   cout<<"-3";
26. }
27. };
```

```
28. class Four: public Two, Three {
29. public:
30. Four()
31. {   cout<<"4";
32. }
33. ~Four()
34. {   cout<<"-4";
35. }
36. };
37. int main()
38. {  One *oneptr=new Four();
39.    Two a;
40.    Three b;
41.    delete oneptr;
42.    return 0;
43. }
```

12. Question 10

Q10) [1 mark] [CO2] Is it possible to inherit from a template class? If yes, will the derived class also b generic? Justify.

---

13. Question 11

Q11) [3 marks] [CO3] Given an array Student_List [n] of size 'n' which contains objects of student class. Student class has following attributes: enrollment number (non-zero and unique), name and marks. The first 'm' student objects in Student_List[] are sorted according to enrollment numbers and remaining objects contain negative/null values in all attributes (representing no students details are stored there till now). **The value of 'm' is not known**. But it is given that m<<n (m is very less than n). Now, given an enrollment number E, propose an efficient logic (write logic/pseudocode) to search and print the index of Student_List[] at which enrollment number E exists. If E does not exists in Student_List[] then return -1.
[Expected time complexity: O ($\log_2$m)].

---

14. Question 12

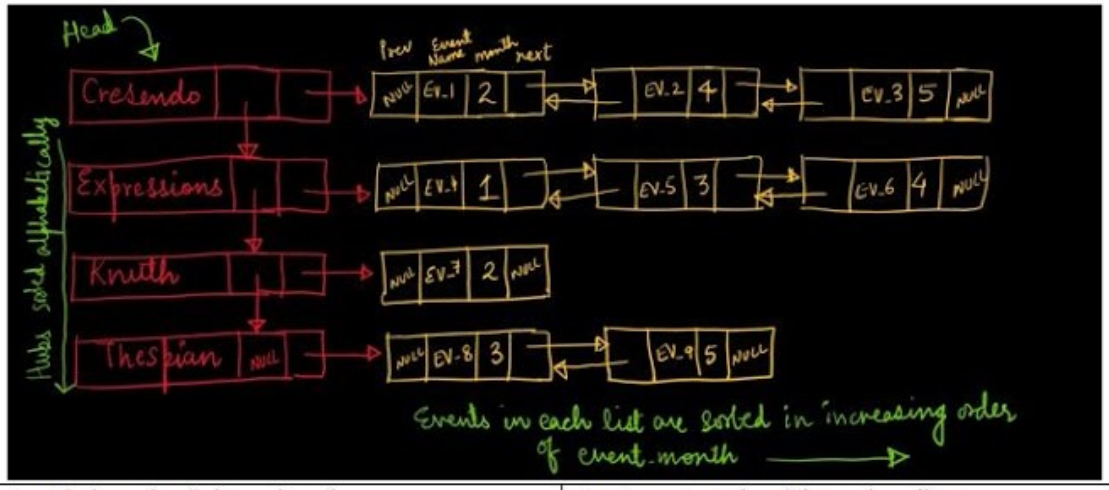
Q12) [4 marks] [CO2] A grocery shop stores various products (minimum zero and maximum unknown number of products) which are identified through product_id, price, name, and manufacturing date. A product can be consumable or non consumable. A consumable product in addition to other attributes is also having an expiry date. Both manufacturing date and expiry date have attributes day, month and year. The grocery shop has a name, location and store ID. Minimum 5 customers are associated with each grocery store and each customer is identified by his name and address. The customers can be members & non-members. The members are having unique membership id while the non-members are having a unique mobile no. The discount calculation method on bills of customers and non-customers is different.

a) [1.5 marks] Draw the complete class diagram showing relationships (along with multiplicity and roles) among classes for the above scenario.
b) [2.5 marks] Implement the class diagram created in part a) in C++. (Write code for creating classes, their attributes, relationships among these classes, and mention virtual functions declaration or abstract classes where applicable). **You don't need to write definitions of the functions or main method.**

---

15. Question 13

Q13) [3 marks] [CO1] There are various hubs in JIIT and in each hub various events takes place throughout the year. All the events occurring in different hubs in a particular year are stored in a list of lists Data Structure as shown below. The structure of two types of nodes in the shown list of lists is also given.




```
struct hub_node  // shown in red
{
        String hub_name;
        hub_node *next_hub;
        event_node *head_event;
};
```

```
struct event_node   //shown in yellow
{
        event_node *prev_event;
        String event_name;
        int event_month;
        event_node *next_event;
};
```

The Red nodes in figure represent hub_node. The linked list of hub_nodes is sorted alphabetically according to hub_names. The head_event pointer of each hub_node points to the doubly linked list of events (shown in yellow nodes) occurring in that hub in a particular year. These events are stored in increasing order of the event_month in which they occur.

Assume above list is created and "hub_node *head" is the head pointer pointing to first node of hub_node list. Complete the following C++ function:

```
hub_node* insert_hub_event ( hub_node* head, String hub_name, String event_name, int event_month)
{
        // write the code in this function to insert a given event (event name and event month) of a particular hub (hub_name given) at correct place. If the hub_name doesnot exist already, add a new hub_node for the new hub and then insert the given event in that list.

}
```

16. Upload answer sheet (scanned single pdf file only) Nomenclature: batch_enrollmentnumber_name_DS_T1.pdf *

Files submitted: