144

B.Sc. Engg. / HD CSE 3rd Semester (52) 26 February 2012 (Afternoon)

ISLAMIC UNIVERSITY OF TECHNOLOGY (IUT)
ORGANISATION OF ISLAMIC CONFERENCE (OIC)
Department of Computer Science and Engineering (CSE)

MID SEMESTER EXAMINATION **WINTER SEMESTER, 2011-2012**

DURATION: 1 Hour 30 Minutes **FULL MARKS: 75**

CSE 4301: Object Oriented Programming

Programmable calculators are not allowed. Do not write anything on the question paper.

There are **4 (four)** questions. Answer any **3 (three)** of them.

Figures in the right margin indicate marks.

---

1. a) How the real world modeling is done with object oriented approach? Explain with real world examples. 5

   b) Suppose the statement "`cin >> message[i];`" is replaced by the statement "`cin.get(message[i]);`" 4
   Now what will be the output of the code fragment if, in response to the prompt, the interactive user types the line "`Please go away`" and presses Enter?

   c) Predict the output or error(s) for the following code segment with explanation: 6

```
1.  class Sample
2.  {
3.  public:
4.          int *ptr;
5.          Sample(int i){
6.          ptr = new int(i);
7.          }
8.          ~Sample(){
9.            delete ptr;
10.       }
11.       void PrintVal(){
12.         cout « "The value is " « *ptr;
13.       }
14. };
15. void SomeFunc(Sample x){
16.    cout « "Say i am in someFunc " « endl;
17. }
18. int main(){
19.   Sample s1= 10;
20.   SomeFunc(s1);
21.   s1.PrintVal();
22. }
```

   d) Define the following terms: 6
   i. `this` pointer
   ii. Data hiding
   iii. Polymorphism

   e) What is a reference? Write one advantage of using a reference parameter? 4

2. a) What is copy constructor and why is it needed? 5

   b) What are the purpose of `new` and `delete` keyword? Mention some advantages of using them instead of `malloc()` and `free()`. 2+5

   c) Create a `StringType` class that allows the following type of operators: 13
   i. String concatenation using the + operator
   ii. String assignment using the = operator
   iii. String comparisons using <, >, and ==
   Also write a proper main function to test your `StringType` class.

3. a) Explain some ways how ambiguity can be introduced when you are overloading functions. 8

b) Design a `Person` class that will encapsulate some attributes (i.e. `name`, `Address`, `contactNo`). Design 6 member functions to access/modify them. Inclusion of appropriate `Constructor/Destructor` in the class definition is mandatory. Now implement the following `AddressBook` class: 17

```
class AddressBook
{
   private:
        Person  *personList;
        int  personCount;
   public:
        AddressBook(int size); /* allocate memory for personList
        ~ AddressBook(); /* deallocate the memory used in
                            personList*/
        void addPerson(Person * p);
        Person * findPerson(char *name);/*return the person object that
                                        matched with the provided name*/
        Person * findPerson(int contact); /*return the person object
                                        that matched with the provided
                                        contact.*/
        int  getPersonCount(); /* return the number of person present in
                                the Address Book*/
        void showAllPerson()
        {
        /*print all the person details present in the personList.*/
        }
};
```

4. a) Write the output of the following code segment with proper explanation: 6

```
class base{
 public:
        int bval;
        base(){ bval=0;}
};

class deri:public base{
  public:
        int dval;
        deri(){ dval=1;}
};
void SomeFunc(base *arr,int size){
        for(int i=0; i<size; i++,arr++)
           cout«arr->bval;
        cout«endl;
}
int main(){
   base BaseArr[5];
   SomeFunc(BaseArr,5);
   deri DeriArr[5];
   SomeFunc(DeriArr,5);
}
```

+ 146

b) What is a virtual function? What is the significance of using of protected access modifier in the context of inheritance? 2+3

c) Suppose a publishing company that markets both book and audiocassette versions of its works. Now, create a `class` called `publication` that stores the `title` (a `string`) and `price` (type `float`) of a publication. From this class derive two `classes`: `book`, which adds a `pageCount` (type `int`); and `tape`, which adds a playing time in `minutes` (type `float`). Each of the three classes should have a `getdata()` function to get its data from the user at the keyboard, and a `putdata()` function to display the data. 14

Write a `main` () function that creates an array of pointers to publication. In a loop, ask the user for data about a particular `book` or `tape`, and use new to create an object of type `book` or `tape` to hold the data. Put the pointer to the object in the array. When the user has finished entering the data for all `books` and `tapes`, display the resulting data for all the books and tapes entered, using a for loop and a single statement such as

```
pubarr[j]->putdata();
```

to display the data from each object in the array.