İ.T.Ü.
Computer Engineering Department
Sanem SARIEL TALAY
Gülşen CEBİROĞLU ERYİĞİT

14.03.2011

Object Oriented Programming 1st Midterm Examination

Duration: 2 hours

Question1 (50 pts):

Analyze run-time behavior of the following code and **fill in the table** given below with **the output** generated by each line of the code. If there is **no output** for a given line, **leave** the related cell **empty**.

```
class Aclass{
  int i;
public:
  Aclass(){
          i = 0;
          cout << "Aclass() default" << endl;}

  Aclass(int new_i){
          i= new_i;
          cout << "Aclass(int)" << endl;}

  ~Aclass(){
          cout  << "~Aclass()" << "i: " << i << endl;}

  Aclass(const Aclass &in_c) {
          i = in_c.i;
          cout << "Aclass(const &)"<< "i: " << i << endl;}
};
```

```
class Bclass{
  int j;
public:
  Bclass(){cout << "Bclass() default" << endl;}

  ~Bclass(){
          cout << "~Bclass()" << endl;}

  void func1(Aclass obj){
          cout << "func1" << endl;}

  void func2(){
          cout << "func2" << endl;
          Aclass obj(7);}

  Bclass(const Bclass &in_b) {
          cout << "Bclass(const &)" << endl;}
};
```

```
int main(){

Line 1   :      Aclass obj1;
Line 2   :      Aclass *obj2_ptr;
Line 3   :      obj2_ptr = new Aclass(5);
Line 4   :      Bclass obj3;
Line 5   :      obj3.func1(*obj2_ptr);
Line 6   :      obj3.func2();
Line 7   :      Bclass obj4 = obj3;
Line 8   :      delete obj2_ptr;
Line 9   :      return 0;
}
```

Line 4:

Line 5:

Line 6:

Line 1:	**Line 7:**
Line 2:	**Line 8:**
Line 3:	**Line 9:**

Question2 (50 pts):

```
1: #include <iostream>
2: using namespace std;
3:
4: class myNode{
5: private:
6:      int data;
7:      myNode *next;
8: public:
9:      myNode(int value):data(value)
10:         {next=NULL;}
11:  };
12:  class myLinkedList{
13: private:
14:     myNode *head;
15: public:
16:     myLinkedList(int [],int);
17:     ~myLinkedList();
18:     void add(int);
19:  };
20: myLinkedList::
21:         myLinkedList(int array[],int size){
22:     head=NULL;
23:     for(int i=0;i<size;i++)
24:             add(array[i]);
25: }
26: void myLinkedList::add(int data){
27:     myNode *n=new myNode(data);
28:     if(head==NULL){
29:             head=n;
30:             return;
31:     }
32:     n->next=head;
33:     head=n;
34: }
35: myLinkedList::~myLinkedList(){
36:     while(head){
37:             cout<<head->data<<endl;
38:             myNode *temp=head;
39:             head=head->next;
40:             delete temp;
41:     }
42: }
43: int main(){
44:     int array[3]={1,2,3};
45:     myLinkedList mylist (array,3);
46:     mylist.add(4);
47:     return EXIT_SUCCESS;
48: }
```

a) (5 pts) The above code gives compiler error at the lines 32 and 39. State the reason and add a **single line of code** in order to make it work. (Please give the line number where your code will be inserted.)

b) (15 pts) We want to change myNode class so that each node carries an id which represents its creation rank. (e.g., the first created myNode object will have an id=1, the second id=2 and so on.) myNode class will be responsible of the assignment of these ids. Please **write the new myNode class** which implements the defined behaviour. (State everything that should be added to the code)

c) (5 pts) **Write the output** which will be printed on the screen when the following print method is invoked.

```
void myLinkedList::print() const{
        myNode *ptr=head;
        while(ptr){
                cout<<ptr->data<<" creation
              id="<<ptr->id<<endl;
                ptr=ptr->next;
        }}
```

d) (10 pts) When we create a new object mylist2 as shown below, mylist2.print() prints the data correctly but we get a runtime error at the last line when the destructors for the two objects are executed. **Give the output** which will be printed on the screen and **explain the reason** of the error.

```
int main(){
        int array[3]={1,2,3};
        myLinkedList mylist(array,3);
        mylist.add(4);
        myLinkedList mylist2(mylist);
        mylist2.print();
        return EXIT_SUCCESS;
}
```

Reason:

e) (15 pts) **Write the method** which should be added to the program in order to eliminate the error you explained in Question 2.d. Please also give the output of the method call "mylist2.print()" in this case. (**Hint:** You may just assume that myLinkedList class also has an add_totheend() method which allocates and adds nodes at the end of the linked list.)