Student Name:_____ Student ID:_____

Cairo University
Faculty of Computer Science and Artificial Intelligence
Subject: Object Oriented Programming
Subject Code: **CS213**
Examiner(s): **Cherry Ahmed**

Mid-term exam
Semester: 1st
Date: 2/1/2022
Duration: **1 hour**

/40
/20

Question 1: Choose the correct answer: **[10 marks]**

1- The header of the subscript operator[] of the class MyArray with an array of doubles should be:
   a. int MyArray::operator[](double& param)
   b. double& MyArray::operator[](int param)
   c. MyArray MyArray::operator[](int param)
   d. double& MyArray::operator[](double param)

2- To initialize a static member variable "pi" of class Circle to value "3.14", you write:
   a. static double Circle::pi = 3.14;
   b. static double pi = 3.14;
   c. double Circle::pi = 3.14;
   d. double static Circle::pi = 3.14;

3- Both aggregation and friend classes represent a relation between 2 classes. Assume that class A is either "aggregated" or "a friend of" class B, which of the following is true:
   a. In aggregation, class A is not a member of class B, while as a friend, class A is a member of class B.
   b. In aggregation, class A is a member of class B, while as a friend, class A is a not member of class B.
   c. In aggregation, a class A is a member of class B, and as a friend, class A is also a member of class B.
   d. In aggregation, a class A is not a member of another class, and as a friend, class A is not a member of class B.

4- The method that moves the write pointer of an fstream file is:
   a. seekg( )
   b. seekp( )
   c. tellg( )
   d. tellp( )

5- Which of the following is an example of multiple inheritance:
   a. Class A inherits from both classes B and C
   b. Class A has 2 derived classes B and C
   c. Class A inherits from class B, and class B inherits from class C
   d. Classes A and B are the children of class C

Question 2: True or False: **[10 marks]**

1. Constructors may not have a return type. (T)
2. If a class has at least one pure virtual function then it is an abstract class. (T)
3. Static member variables cannot be accessed by non-static member functions. (F)
4. The statement **`obj1 = obj2;`** calls the copy constructor. (F)
5. One must add a dummy parameter when overloading the postfix increment operator ++ (T)

Question 3: Answer the following [20 marks]

Write a class **MyCharArray** that stores a *dynamic character array* **(str)** and its *size* **(s) (1.5 marks)**.
The following should be implemented:

1- A parameterized constructor that receives the size of the character array. **(2 marks)**
2- A member function setArray that takes a cstring as a parmater and copies it to the class's array. Make sure there is space for the parameter cstring to be copied into the class array. The function returns true if the copy was successful and false otherwise. **(4 marks)**
3- A copy constructor. **(3 marks)**
4- A member function *display* that outputs the array to the screen. **(1 marks)**
5- A member function *replaceAll* which takes a character ***cOld*** and a character ***cNew*** as parameters and replaces all occurrences of the character ***cOld*** by the new character ***cNew***. **(3 marks)**
6- Overload the operator + to concatenate two **MyCharArray** objects and return the result in a third **MyCharArray** object. **(4.5 marks)**
7- A destructor to free memory allocated by the class. **(1 marks)**

hint: you may use the ready-made ***strlen*** and ***strcpy*** functions in string library in your code.

```
int main()
{
  char s1[]="Adel|Ahmed|";
  MyCharArray obj1(30);
  //outputs "Adel|Ahmed|"
  if(obj1.setArray(s1))
     obj1.display();
  else
     cout<<"Unsuccessful copy";

  MyCharArray obj2 = obj1;
  obj2.replaceAll('|','#');

  //outputs "Adel#Ahmed#"
  obj2.display();
  char s2[]="Khaled|";
  MyCharArray obj3(10);
   //outputs "Khaled|"
   if(obj3.setArray(s2))
      obj3.display();
   else
      cout<<"Unsuccessful copy";
  MyCharArray obj4 = obj1+obj3;

  //outputs "Adel|Ahmed|Khaled|"
  obj4.display();


  return 0;
  }
```

```
class MyCharArray
{
    char* str;
    int _size;
public:
    MyCharArray(int s){
       _size = s;
       str = new char[s];
    }
    bool setArray(char * s)
    {
        if((strlen(s)+1) > _size)
            return false;
        strcpy(str,s);
        return true;
    }
    MyCharArray(const MyCharArray& ms)
    {
        _size = ms._size;
        str = new char[_size+1];

        for(int i=0;i<_size;i++)
            str[i]=ms.str[i];
        str[_size]='\0';
    }
    void display()
    {
        cout<<str<<endl;
    }

    void replaceAll(char oldC, char newC)
    {
        for(int i=0;i<_size;i++)
            if(str[i]==oldC)
                str[i]= newC;
    }
    MyCharArray operator+(const  MyCharArray& s2)
    {
        int resSize = strlen(str)+strlen(s2.str)+1;
        MyCharArray res(resSize);
        char * resStr= new char[resSize];
        for(int i=0;i<strlen(str);i++)
            resStr[i]=str[i];
        for(int i=strlen(str);i<resSize;i++)
            resStr[i]=s2.str[i - strlen(str)];
        resStr[resSize-1]='\0';
        res.setArray(resStr);
        return res;
    }

    ~MyCharArray()
    {
        delete[] str;
    }
};
```