COSC 1320, Programming Assignment 1 (Deadline: 06/08/2016)

In this assignment, you are going to define a class called **Student**, which will be used to store student information at a university.

Before you go further with this programming exercise, please notice that you must follow all object oriented programming approaches and conventions. In other words, you program must not a sequential C++ code having the extension .cpp. The file extension cpp does not make the code an object oriented program.

First of all, please write down the behavior you expect from a student. What attributes (i.e., member variables of the class **Student**) and actions (i.e., member functions of the class **Student**) may a university student have? In a text file called **predesign.txt**, please write all the member variable and function names you think of. You **must** turn in this file with the program code.

Now, you are going to define the class **Student** based on the specifications below.

1. A student must have first name (type of string), last name (type of string) and ID (type of int, and ID does NOT start with 0). In other words, the class **Student** must have 3 member variables. All variables **MUST** be **private**. You **MUST NOT** initialize these variables.
2. The student class must have a **public static** variable called **numberOfStudents** that counts the number of **Student** objects that are created. Please initialize **numberOfStudents** to 0 (zero).
3. Please define a no-argument constructor (the default constructor).
    - Please increment (plus 1) the value of the **public static** variable **numberOfStudents.**
    - You **must not** do anything else.
4. Please define a constructor that takes three arguments: first name (type of string), last name (type of string) and ID (type of int).
    - Please increment (plus 1) the value of the **public static** variable **numberOfStudents.**
    - Please assign the constructor argument values to the member variables of the class **Student**.
5. Please define 2 **public** member functions called
    - **getName**, which returns the student first and last name together (e.g., **John Doe, must return a single string.**).
    - **getID**, which returns the student's ID (e.g., **1234**).
    - What happens if we make these 2 functions **private**? Please discuss this in the file **predesign.txt**.

You must define the class **Student** in 2 separate files called:

1. **Student.h**, which is the interface file. You must first create this file. The interface file consists of member variable and function declarations (i.e., class definition). Interface file **Student.h** template:

```
#include <string>
using namespace std;

class Student
{
.
.
.
};
```

2. **Student.cpp**, which is the implementation file. You must include class **Student** interface file (**NOT THE CLASS INTERFACE**) in the implementation file. The implementation file consists of the function definitions. Please notice that constructors are also functions. Implementation file **Student. cpp** template:

```
#include "Student.h"

.
.
.
```

Finally, you are going to use the driver file (i.e., **main.cpp**) provided in order to test your program.

```
#include <iostream>
#include <string>
using namespace std;
#include "Student.h"

main()
{
    Student st1("John", "Smith", 1234), st2("Eric", "Evans", 2345), st3("Gina", "Kim", 5442), st4;  cout <<
  "We created " << Student::numberOfStudents << " student objects." << endl;
    cout << st1.getID() << " " << st1.getName() << endl;
    cout << st2.getID() << " " << st2.getName() << endl;
    cout << st3.getID() << " " << st3.getName() << endl;
    cout << st4.getID() << " " << st4.getName() << endl;

system("pause");
}
```

If you define and implement the class student properly, when you execute the file main.c, the program output should be (note that the last line can produce a random number instead of 0):

```
We created 4 students
Objects 1234 John Smith
2345 Eric Evans
5442 Gina Kim
0
Press any key to continue......
```

Grading Criteria

1. The assignment is due **June 8, Wednesday 2016 11.59 pm** Houston, Texas local time.
2. You must submit your assignment through Blackboard Learn assignment system. Submissions via message are not going to be accepted.
3. You must add comment lines (**//explain the statements briefly**) in your program code. If you do not add comment lines, TAs are going to **deduct 10 points**.
4. You must use indentation for the blocks in your program. Please read http://en.wikipedia.org/wiki/Indent_style for more information.

```
// GOOD EXAMPLE
int main()
{
   int x;
   cin >> x;
   // Check the value of x
   if ( x == 1)
   {
      cout << "x is ...";
   }
   else
   {
      cout << "x is ...";
   }
   . . .
   return 0;
}
```

```
// BAD EXAMPLE
int main()
{
int x;
cin >> x;
// Check the value of x
if ( x == 1)
{
cout << "x is ...";
}
else
{cout << "x is ...";}
. . .
return 0;
}
```

If you do not use the indentation, TAs are going to **deduct 10 points**.

5. Please select meaningful variable names (e.g., **x323wsw**is a very bad variable name). Otherwise, TAs are going to **deduct 5 points** for each improper variable name.
6. If your program **cannot be compiled**, TAs are going to **deduct at least 40 points**.
7. If your program **terminates at run time**, TAs are going to **deduct at least 40 points**.
8. You must turn in 3 files: **predesign.txt**, **Student.h**, and **Student.cpp**. When you are ready to turn in all your files, please put them **IN ONE FILE** called **PeopleSoftIDFirstNameLastNamePA1.zip**. (e.g., 1234567JohnDoePA1.zip) Please submit .zip file only. Please notice that if the files are uploaded individually, or the file extension is **not** .zip your program will **NOT** be graded and your submission will be returned.
9. **If your file cannot be opened or your file is corrupted, your grade will be 0 (zero).**
10. **predesign.txt**: **10 points**
    a. **5 points** for the correct answer of the question about public/private member functions. Please read the instructions above carefully in order not to lose points.
    b. **2 points** for the member variables you suggested. We expect you to suggest at least 2 more variables.
    c. **3 points** for the member functions you suggested. We expect you to suggest at least 3 more member functions.
11. **Student.h**: **40 points**
    a. **5 points** for each proper member variable declaration (**4x5 points =20 points**). Please read the instructions above carefully in order not to lose points. You must **NOT** initialize any variables except the static variable.
    b. **5 points** for each proper member function declaration (**4x5 points =20 points**). Please read the instructions above carefully in order not to lose points.

12. **Student.cpp**: **50 points**
    a. **15 points** for proper no-argument constructor implementation. Please read the instructions above carefully in order not to lose points. Please notice that you modify **the static variable** only.
    b. **20 points** for proper 3-argument constructor implementation. Please read the instructions above carefully in order not to lose points.
    c. **10 points** for proper implementation of the member function getName. Please notice that there is a space between the first and last name in the output.
    d. **5 points** for proper implementation of the member function getID.
13. If you define and implement the Student class in the same file, TAs are going to **deduct 25 points**.
14. If you have a question, please **email the TAs your question**.
15. Deductions for the late submission: (One day, 20%), (Two days, 30%), (Three days, 40%), and so forth…

☺ Good Luck and Have Fun ☺