**C/C++ STUDENT 'CLASS' LAB REPORT**

**1) Enter your name, student ID, platform (Mac or PC) and date**

Name and Student ID: Samuel Indurkar, 0888068

Class: CIS054 C/C++ Programming

Platform (Mac or PC): gcc and eclipse  
Date: 7/15/2017

**DESCRIPTION:**

Modify the Student 'class' files as follows:  
1) Add a **Major** field to the Student class definition. Update the student.h and student.cpp files to include the additional field. Be sure to include a get and set method for the **Major** field.

2) Add several majors such as CIS, Math, English, etc. to the students in the array of students located in the main program.

3) When displaying the list of student ID numbers and student names, also display their major. For C++ use the setw( ) function to set the widths of each column so that the columns line up on the screen when displayed. For the C-language, set the parameters for **%s** in the format string.

4) Change some of the ID numbers and GPAs to illegal values. Have the program detect and display an error message when the illegal values are present.

**EXTRA CREDIT:** 3 points – must be submitted at the same time as the original lab report

Display a header at the top of the table similar to:  
 **ID # NAME MAJOR**

**============================================**

**LAB REPORT:  
2) Determine the Inputs, Processing and Outputs before creating the program**

|  |  |  |
| --- | --- | --- |
| **INPUTS** | **PROCESSING** | **OUTPUTS** |
| Pre-existing input already coded inside the program. No user input taken from cin | use the constructor to initialize, then use the get/set access methods to access the private variables and display using cout | Display StudentID, student name, and the major. |

**3) Fill in the EXPECTED & ACTUAL RESULTS**

|  |  |  |
| --- | --- | --- |
| **TEST DATA VALUES**  Provide different examples including an illegal Student ID | **EXPECTED RESULT**  Computed values before the program is run | **ACTUAL RESULT**  Fill in the output displayed  by the program |
| ("Joe Williams", "CIS", 44536, 3.4),  ("Sally Washington", "Biology", 55458, 3.7),  ("Fred MacIntosh", "Math", 66587, 2.9),  ("Jose De La Cruz", "Chemistry", 67892, 3.5),  ("777 Dan McElroy", "CIS", 77777, 4.0),  ("Thinh Nguyen", "CIS", 73657, 3.6) |  | ID # NAME MAJOR  ============================================  44536 Joe Williams CIS  55458 Sally Washington Biology  66587 Fred MacIntosh Math  67892 Jose De La Cruz Chemistry  77777 --Bad name entered CIS  73657 Thinh Nguyen CIS |

**DISCUSSION:**

**4) Complete the DISCUSSION section. It does not need to be long, but it needs to be complete.**4a) What did you do to develop the program? ("Followed the Directions" is not a complete description)

Pre-existing input already coded inside the program. No user input taken from cin. Use the constructor to initialize, then, use the get/set access methods to access the private variables and display Display StudentID, student name, and the major using cout

4b) What problems did you have and how did you overcome the problems?

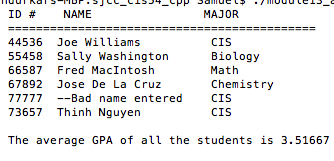
To get the extra credit, I added the header row. That was easy and simple. But for the Major, you asked us to use setw(). Initially it was not working properly, so I switched to “printf %20s” but printf with getName() was giving compilation error so I went backk to cout. Finally after studying set, and “left” I was able to overcome the problem.

**PROGRAM OUTPUT:**

**5) Show a screen of the program execution that includes an illegal selection for name.**For extra credit, display a header row and a ==== separator line.



Refer to previous lab assignments for instructions on how to capture a screen or portions of a screen for either the PC or a Mac



**PROGRAM LISTING:**

**6) Copy and paste the code that YOU typed to make the program work. Your program should include a comment block at the top that shows the name of the program, date, version and your name.**

/\*

\* module13\_assignment\_on\_classes.cpp

\*

\* Created on: Jul 15, 2017

\* Author: Samuel

\*/

// Defines the entry point for the console application.

//

#include <iostream>

#include <cstdio>

#include <iomanip>

#include "Student.h"

using namespace std;

// define an array of students

Student CIS054[] = {

Student ("Joe Williams", "CIS", 44536, 3.4),

Student ("Sally Washington", "Biology", 55458, 3.7),

Student ("Fred MacIntosh", "Math", 66587, 2.9),

Student ("Jose De La Cruz", "Chemistry", 67892, 3.5),

Student ("777 Dan McElroy", "CIS", 77777, 4.0),

Student ("Thinh Nguyen", "CIS", 73657, 3.6)

};

int main(int argc, char\* argv[])

{

int NumberOfStudents = sizeof(CIS054)/sizeof(Student);

// Display the header line

// List all the students in the course

cout << " ID # NAME MAJOR" <<endl;

cout << " ============================================" <<endl;

for (int i=0; i<NumberOfStudents; i++)

{

cout << " " << CIS054[i].getIdNumber() << " " <<setw(20)<< left<< CIS054[i].getName() <<" " << CIS054[i].getMajor()<< setw(0)<<endl;

}

cout << endl; // blank line after displaying the student names

// compute the average gpa of all the students

double total=0;

for (int i=0; i<NumberOfStudents; i++)

total += CIS054[i].getGpa();

double average = total / NumberOfStudents;

cout << " " << "The average GPA of all the students is " << average << endl << endl;

return 0;

}