## Project 2 – Grade Report (Final Testing)

Now that your project is completed, you have another chance to check if there are any items that should be corrected. If so, make the necessary corrections.

**Make sure you understand why you are correcting your code. If not sure, ask me.**

**NOTE: Replace** your **student\_data.txt** file with the new one.

|  |
| --- |
| **Person.h** |
| **#include** string, iostream(any order, but they need to be at the top of the file) |
| **Functions** are in the **order** specified in the instructions. |
| Overloaded constructor has 2 parameters:  **const** **string&**  **const** **string&** |
| Function **setName** has 2 parameters:  **const** **string&**  **const** **string&** |
| Function **getLastName** is a **const** function**.** |
| Function **getName** is a **const** function**.** |
| Function **getName** is a void function and has 2 parameters:  **string&**  **string&** |
| Function **printName** is a **const** function**.** |
| These are the **ONLY** functions declared:  Default constructor  Overloaded constructor  Destructor  setName  getLastName  getName  printName |
| **Person.cpp** |
| (No need for name header in this file) |
| (No need to include any libraries and/or namespaces. They are already included in the .h file**)** |
| **Functions** are in the **order** specified in the instructions. |
| **Default constructor** sets both member variables to **N/A.** |
| Overloaded constructor |
| Function **setName** sets both member variables to given values passed by the parameters. |
| Function **getLastName** has **only one statement** that returns a **string**, the last name. |
| Function **getName** sets both parameters to values stored in the member variables. |
| Function **printName** prints both last name and first name in the **format** specified in the instructions. |
| **Destructor** is empty. |
| **Course.h** |
| **Functions** are in the **order** specified in the instructions. |
| setCourseInfo |
| Function **printCourseInfo** passes a **Boolean** as a parameter. |
| Function **printCourseInfo** is a **const** function**.** |
| Function **getCourseCredits** is a **const** function**.** |
| Function **getCourseNumber** is a **const** function**.** |
| Function **getCourseName** is a **const** function**.** |
| Function **getCourseGrade** is a **const** function**.** |
| These are the **ONLY** functions declared:  Default constructor  Destructor  setCourseInfo  printCourseInfo  getCourseCredits  getCourseNumber  getCourseName  getCourseGrade |
| **Course.cpp** |
| (No need to include any libraries and/or namespaces. They are already included in the .h file**)** |
| **Functions** are in the **order** specified in the instructions. |
| **Default constructor** sets the course grade to an asterisk and the course credits to **0**; it does **NOT** set the **strings** to empty strings (can be set to “N/A” or similar, or not set at all). |
| Function **setCourseInfo** sets all member variables to given values passed by the parameters. |
| Function **printCourseInfo** has an **IF/ELSE** statement that does **NOT** contain a comparison operator. |
| Function **getCourseCredits** contains one statement only. |
| Function **getCourseNumber** contains one statement only. |
| Function **getCourseName** contains one statement only. |
| Function **getCourseGrade** contains one statement only. |
| **Destructor** is empty. |
| **Student.h** |
| There are **ONLY** 4 private member variables:  int studentID  int numberOfCourses bool tuitionWasPaid  vector<Course> coursesEnrolled |
| These are the **ONLY** functions declared:  Default constructor  Destructor  setStudentInfo  getID  getNumberOfCourses  getCreditsEnrolled  isTuitionPaid  isEnrolledInCourse  getGpa  billingAmount  printStudentInfo  printStudentInfo *(overloaded)*  getCoursesEnrolled  printStudentInfoToFile |
| **Student.cpp** |
| Only inclusion: “Student.h” |
| **Default constructor** sets member variables as follows:  numberOfCourses = 0  studentID = 0  tuitionWasPaid = false |
| Function **setStudentInfo** parameters (in this order):  **const string&**  **const string&**  **int**  **bool**  **const vector<Course>&** |
| Function **setStudentInfo** calls function **setName** to set the first name and the last name to the values of the parameters passed. |
| Function **setStudentInfo** sets **studentID**, **tuitionWasPaid**, and **coursesEnrolled** to the values of the parameters passed. |
| Function **setStudentInfo** sets **numberOfCourses** to the value of the parameter passes after **casting** the type returned by the vector size to an integer. |
| Function **getID** is a **const** function. |
| Function **getID** has only one statement returning **studentID**. |
| Function **getNumberOfCourses** is a **const** function. |
| Function **getNumberOfCourses** has only one statement returning **numberOfCourses**. |
| Function **getCreditsEnrolled** is a **const** function. |
| Function **getCreditsEnrolled** contains a **LOOP** that calls the function **getCourseCredits** to sum up the total credits. |
| Function **isTuitionPaid** is a **const** function. |
| Function **isTuitionPaid** has only one statement returning **tuitionWasPaid**. |
| Function **isEnrolledInCourse** is a **const** function. |
| Function **isEnrolledInCourse**  uses an **IF/ELSE** statement to check if the student is enrolled in any courses, before going through the **loop**.  (not really necessary) |
| Function **isEnrolledInCourse**  calls function **getCourseNumber** to access the course number. |
| Function **isEnrolledInCourse**: All paths return a value. |
| Function **getGpa** is a **const** function. |
| Function **getGpa** uses a **SWITCH** statement inside the **LOOP**. |
| Function **getGpa** calls the function **getCourseCredits** to access the credits in the vector. |
| Function **billingAmount** is a **const** function. |
| Function **billingAmount** uses function **getCreditsEnrolled** to get the credits. |
| Function **printStudentInfo()** is a **const** function. |
| Function **printStudentInfo** calls the function **printName** to print the first and last name. |
| **Overloaded** function **printStudentInfo(double)** is a **const** function. |
| **Overloaded** function **printStudentInfo(double)** uses **setw** to format output. |
| **Overloaded** function **printStudentInfo(double)** calls the following functions:  printName()  printCourseInfo()  getCreditsEnrolled()  getGpa()  billingAmount() |
| Function **getCoursesEnrolled()** is a **const** function. |
| Function **getCoursesEnrolled()** has only one statement that returns the courses in which the student is enrolled. |
| Function **printStudentInfoToFile** uses function **getName (first, last)** to retrieve the values of first and last name and store it into two strings. |
| Function **printStudentInfoToFile** calls functions:  getCourseNumber  getCourseName  getCourseCredits  getCourseGrade  getCreditsEnrolled  getGpa  billingAmount |
| **Destructor** is empty. |
| **StudentList.h** |
| There are **ONLY** 3 private member variables:  int count  Node\* first  Node\* last |
| These are the **ONLY** functions declared:  Default constructor  Destructor  addStudent  getNoOfStudents  printStudentByID  printStudentsByCourse  printStudentsByName  printStudentsOnHold  printAllStudents  printStudentsToFile  destroyStudentList |
| **STudentList.cpp** |
| Only inclusion: “StudentList.h” |
| **Default constructor** sets member variables as follows:  first = NULL  last = NULL  count = 0 |
| Function **addStudent** has only one parameter:  **const Student&** |
| Function **addStudent** considers 2 cases: when the list is empty and when the list has more than one node. |
| Function **addStudent** should increment the count. |
| Function **getNoOfStudents** is a **const** function. |
| Function **getNoOfStudents** has only one statement returning **count**. |
| Function **printStudentByID** is a **const** function. |
| Function **printStudentByID** has 2 parameters:  **int**  **double** |
| Function **printStudentByID** considers when the list is empty and outputs an error message using **cerr**. |
| Function **printStudentByID** uses a **WHILE** loop (**NOT** a FOR loop). |
| Function **printStudentByID** stops the **WHILE** loop when the student is found using a **Boolean** value. |
| Function **printStudentByID** uses function **printStudentdInfo** to print the information. |
| Function **printStudentByID** considers the case when the student is not in the list and prints an error message (either cout or cerr). |
| Function **printStudentsByCourse** is a **const** function. |
| Function **printStudentsByCourse** has one parameter:  **const string&** |
| Function **printStudentsByCourse** considers when the list is empty and outputs an error message using **cerr**. |
| Function **printStudentsByCourse** calls functions **isEnrolledInCourse** and **printStudentInfo**. |
| Function **printStudentsByCourse** considers the case when no students are enrolled in the class and prints an error message (either cout or cerr). |
| Function **printStudentByName** is a **const** function. |
| Function **printStudentByName** has one parameter:  **const string&** |
| Function **printStudentByName** considers when the list is empty and outputs an error message using **cerr**. |
| Function **printStudentByName** calls functions **getLastName** and **printStudentInfo**. |
| Function **printStudentByName** considers the case when the student is not in the list (either cout or cerr). |
| Function **printStudentByName** considers the case when there is more than one student with the same last name (either cout or cerr) 🡺 error message is not necessary. |
| Function **printStudentsOnHold** is a **const** function. |
| Function **printStudentsOnHold** has one parameter:  **double** |
| Function **printStudentsOnHold** considers when the list is empty and outputs an error message using **cerr**. |
| Function **printStudentsOnHold** considers the case when there are no students on hold (either cout or cerr). |
| Function **printStudentsOnHold** calls functions **isTuitionPaid**, **printStudentInfo** and **billingAmount**. |
| Function **printAllStudents** is a **const** function. |
| Function **printAllStudents** has one parameter:  **double** |
| Function **printAllStudents** considers when the list is empty and outputs an error message using **cerr**. |
| Function **printAllStudents** calls function **printStudentInfo**. |
| Function **printStudentsToFile** is a **const** function. |
| Function **printStudentsToFile** has 2 parameters:  **ofstream&**  **double** |
| Function **printStudentsToFile** considers when the list is empty and outputs an error message using **cerr**. |
| Function **printStudentsToFile** calls function **printStudentInfoToFile**. |
| Function **destroyStudentList** uses a **LOOP** to delete each node in the list. |
| Function **destroyStudentList** re-sets the following variables:  first = NULL  last = NULL  count = 0 |
| **Destructor** calls function **destroyStudentList**. |
| **OTHER** |
| Format of **name header** is as shown on the syllabus (check the syllabus if not sure). The name header should be on the **Main.cpp** file, because you worked on the whole project. |
| There is **NO** **horizontal scrolling**. All statements are short enough. |
| There is a **space** before and after operators. |
| All code is properly indented. |
| There is a **line** in between functions (both declarations and definitions). |
| All **variables** and **objects** have a **descriptive** identifier. |
| All **variables** and **objects** use the **camelCase** convention. |

The next section of the test deals with the output produced by your application. Replace the **student\_data.txt** file with the one provided.

Execute the program and check that the output is as expected—this includes lines, spaces, upper- and lower-case letters, and punctuation.

|  |
| --- |
| **Menu** |
| **\*\*\* MAIN MENU \*\*\***  **Select one of the following:**  **1: Print all students**  **2: Print student information**  **3: Search student by last name**  **4: Print students by course**  **5: Print students on hold**  **6: Print students to file**  **7: To exit**  **Enter your choice:** |
| **Choice 1** |
| **Enter your choice: 1**  **Student Name: Parker, Peter**  **Student ID: 654321**  **Number of courses enrolled: 4**  **Course No Course Name Credits Grade**  **CSC150 ComputerSci 4 A**  **CHM348 Chemistry 4 A**  **ENG101 English 3 B**  **BIO234 Biology 4 A**  **Total number of credit hours: 15**  **Current Term GPA: 3.80**  **-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-**  ***(there should be 7 students in between)***  **-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-**  **Student Name: Wayne, Bruce**  **Student ID: 123456**  **Number of courses enrolled: 5**  **Course No Course Name Credits Grade**  **CHM348 Chemistry 4 B**  **CSC201 ComputerSci 4 B**  **MTH345 Mathematics 4 A**  **ENG101 English 3 A**  **HIS101 History 3 A**  **Total number of credit hours: 18**  **Current Term GPA: 3.56**  **-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-**  **Press any key to continue . . .** |
| **Choice 2** |
| **Enter your choice: 2**  **Please enter student's ID: 123456**  **Student Name: Wayne, Bruce**  **Student ID: 123456**  **Number of courses enrolled: 5**  **Course No Course Name Credits Grade**  **CHM348 Chemistry 4 B**  **CSC201 ComputerSci 4 B**  **MTH345 Mathematics 4 A**  **ENG101 English 3 A**  **HIS101 History 3 A**  **Total number of credit hours: 18**  **Current Term GPA: 3.56**  **-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-**  **Press any key to continue . . .** |
| **Enter your choice: 2**  **Please enter student's ID: 159753**  **Student Name: Richards, Reed**  **Student ID: 159753**  **Number of courses enrolled: 3**  **Course No Course Name Credits Grade**  **CHM348 Chemistry 4 B**  **MTH345 Mathematics 4 B**  **ENG101 English 3 B**  **Total number of credit hours: 11**  **Current Term GPA: 3.00**  **-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-**  **Press any key to continue . . .** |
| **Enter your choice: 2**  **Please enter student's ID: 654321**  **Student Name: Parker, Peter**  **Student ID: 654321**  **Number of courses enrolled: 4**  **Course No Course Name Credits Grade**  **CSC150 ComputerSci 4 A**  **CHM348 Chemistry 4 A**  **ENG101 English 3 B**  **BIO234 Biology 4 A**  **Total number of credit hours: 15**  **Current Term GPA: 3.80**  **-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-**  **Press any key to continue . . .** |
| **Enter your choice: 2**  **Please enter student's ID: 345678**  **Student Name: Luthor, Lex**  **Student ID: 345678**  **Number of courses enrolled: 3**  **Course No Course Name Credits Grade**  **CSC201 ComputerSci 4 \*\*\***  **HIS101 History 3 \*\*\***  **BIO234 Biology 4 \*\*\***  **Total number of credit hours: 11**  **\*\*\* Grades are being held for not paying the tuition. \*\*\***  **Amount Due: $3795.00**  **-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-\*-**  **Press any key to continue . . .** |
| **Enter your choice: 2**  **Please enter student's ID: 111111**  **No students with ID 111111 found in the list.**  **Press any key to continue . . .** |
| **Choice 3** |
| **Enter your choice: 3**  **Please enter the student's last name: Wayne**  **123456 - Wayne, Bruce**  **Press any key to continue . . .** |
| **Enter your choice: 3**  **Please enter the student's last name: Rogers**  **789456 - Rogers, Steven**  **Press any key to continue . . .** |
| **Enter your choice: 3**  **Please enter the student's last name: Parker**  **654321 - Parker, Peter**  **Press any key to continue . . .** |
| **Enter your choice: 3**  **Please enter the student's last name: Johnson**  **No students with last name Johnson is in the list.**  **Press any key to continue . . .** |
| **Choice 4** |
| **Enter your choice: 4**  **Please enter the course number: BUS128**  **No students enrolled in BUS128.**  **Press any key to continue . . .** |
| **Enter your choice: 4**  **Please enter the course number: CHM348**  **654321 - Parker, Peter**  **645915 - Grimm, Benjamin**  **159753 - Richards, Reed**  **789456 - Rogers, Steven**  **123456 - Wayne, Bruce**  **Press any key to continue . . .** |
| **Enter your choice: 4**  **Please enter the course number: CSC201**  **765345 - Gordon, Barbara**  **345678 - Luthor, Lex**  **145632 - Kent, Clark**  **789456 - Rogers, Steven**  **123456 - Wayne, Bruce**  **Press any key to continue . . .** |
| **Enter your choice: 4**  **Please enter the course number: MTH345**  **324598 - Holland, Alec**  **765345 - Gordon, Barbara**  **159753 - Richards, Reed**  **123456 - Wayne, Bruce**  **Press any key to continue . . .** |
| **Enter your choice: 4**  **Please enter the course number: ENG101**  **654321 - Parker, Peter**  **159753 - Richards, Reed**  **123456 - Wayne, Bruce**  **Press any key to continue . . .** |
| **Enter your choice: 4**  **Please enter the course number: HIS101**  **345678 - Luthor, Lex**  **145632 - Kent, Clark**  **789456 - Rogers, Steven**  **123456 - Wayne, Bruce**  **Press any key to continue . . .** |
| **Enter your choice: 4**  **Please enter the course number: BIO234**  **654321 - Parker, Peter**  **324598 - Holland, Alec**  **765345 - Gordon, Barbara**  **345678 - Luthor, Lex**  **145632 - Kent, Clark**  **Press any key to continue . . .** |
| **Choice 5** |
| **Enter your choice: 5**  **765345 - Gordon, Barbara**  **Amount Due: $4140.00**  **345678 - Luthor, Lex**  **Amount Due: $3795.00**  **145632 - Kent, Clark**  **Amount Due: $3795.00**  **Press any key to continue . . .** |
| **Choice 6** |
| **Enter your choice: 6**  **Enter a name for a new file to write on (include extension .txt):**  **new\_file.txt**  **Press any key to continue . . .** |
| File new\_file.txt should be inside the project and formatted as requested. |
| **Choice 7** |
| Exiting the program does not produce any errors and there is no delay after hitting the ENTER key or clicking on the (X) closing icon. |

**Turn in your project named:** **A250\_P1\_TeamName**