Project 2 – Grade Report STL (Part A)

For **project 2**, you will be using the files you created in **project 1**. You will need to work with the **same team members** and you will need to keep the **same team name**.

For this part of the project, you will change the way the student list is implemented. Instead of using a singly-linked list, you will be using the **STL vector class**.

IMPLEMENTATION

All files, except for the **StudentList.h** and **StudentList.cpp**, will remain the same. Below there is a list of modifications you need to implement.

- **StudentList.h**
  - Include the **STL vector class**
  - Remove the **Node** class (completely)
  - Delete all member variables of the **StudentList** class and replace them with a single member variable: a **pointer** named **studentList** that will point to an **STL vector** of **Student objects**.
  - All functions declarations will remain the same.

- **StudentList.cpp**
  - All member functions will need to be re-implemented, because they are all using an **STL vector** instead of a singly-linked list.

  - **Default constructor**
    - Initialize your pointer so that it points to a **dynamic** **STL vector** of objects of type **Student**.

  - **Function addStudent**
    - Insert the student object passed by the parameter into the **vector** using the function **push_back**.
    - This function should have only **one statement**.

  - **Function getNoOfStudents**
    - The **size** of the vector determines how many students are in the list.
    - Make sure you **cast** the return value of the function **size** before returning the value as an **int**.
    - This function should have only one statement.

  - **Function printStudentByID**
    - The overall format of the function stays the same (printing error messages if the list is empty, and if the student was not found).
    - Use an **iterator** and a **WHILE** loop to traverse the vector, making sure you **stop** the loop once the student is found.

- Which iterator should you use? You are not modifying the object to which the iterator is pointing; therefore, you should use a **constant iterator**.

- **Function printStudentsByCourse**
  - The overall format of the function stays the same (printing error messages if the list is empty, and if there are no students enrolled in the course).
  - Use an **iterator** and a **FOR** loop to traverse the vector.
  - Which iterator should you use?

- **Function printStudentsByName**
  - The overall format of the function stays the same (printing error messages if the list is empty, and if there are no students with the given last name).
  - Use an **iterator** and a **FOR** loop to traverse the vector (we will consider the possibility that there is more than one student with the given last name).
  - Which iterator should you use?

- **Function printStudentsOnHold**
  - The overall format of the function stays the same (printing an error message if the list is empty).
  - Consider also the case when there are no students on hold and print out the message, "There are no students on hold."
  - Use an **iterator** and a **FOR** loop to traverse the vector.
  - Which iterator should you use?

- **Function printAllStudents**
  - The overall format of the function stays the same (printing an error message if the list is empty).
  - Use an **iterator** and a **FOR** loop to traverse the vector.
  - Which iterator should you use?

- **Function printStudentsToFile**
  - The overall format of the function stays the same (printing an error message if the list is empty).
  - Use an **iterator** and a **FOR** loop to traverse the vector.
  - Which iterator should you use?

- **Function destroyStudentList**
  - Your **StudentList** class has only a pointer that points to a dynamic vector. Use the delete statement to delete the object (in this case, the vector) to which the pointer is pointing. The vector will use its own destructor to complete the job.
  - Make sure you NULL the pointer.

- **Destructor**
  - As before, this function will call the function **destroyStudentList**.

DATA FILE

You will use a new text file, **student_data_2.txt**, to test your project. Remove all **.txt** files from your project and update the name of the file where necessary.

Test your project and compare your output with the output produced by the file **output.exe**. As usual, pay attention to details (spacing, case, punctuation, spelling, etc.).

NAME HEADER

Remove all name headers and include a **name header** only in the **Main.cpp** file.

EXPECTED COMPLETION

Due:

- **MW class: Monday, November 16th**, at the beginning of class.
- **TTh class: Tuesday, November 17th**, at the beginning of class.