Staff Evaluation
Increase percentage

The assignment problem:

You are asked to develop an application with a main goal to calculate the evaluation increase percentage of the staff teaching CS201R over two years. Your application will read input from two files. First file is called "INFO.txt" and it includes a list of current staff with their staff IDs and first and last names. Second file is called "Grades.txt" and it includes a list of staff IDs and their evaluation grades in CS201R course. The grades are listed in the following order "Spring1 Fall1 Spring2 Fall2." (Evaluation grades are graded out of 5) Note that not all staff has grades (some are new). Also note that the staff' IDs in the second file are not necessary consistent (in the same order) with ID's in the first file. Your application starts by providing the following menu to the user:

```
C:\WINDOWS\system32\cmd.exe
Welcome to the Evaluation Increase % Application
---------------------------------------------------------
Please make a selection:
(1) Print staff roster
(2) Print staff roster with their evaluation grades
(3) Print staff roster with staff's increase %
(4) Exit
Enter your choice:
```

The main menu options are as follows:

1. **Print Staff Roster:** this option reads the data from the first file and prints the staff' IDs and full names (Last name, First name) –see following figure:

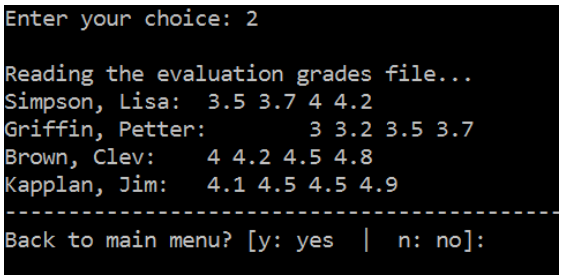
```
Please make a selection:
(1) Print staff roster
(2) Print staff roster with their evaluation grades
(3) Print staff roster with staff's increase %
(4) Exit
Enter your choice: 1

Reading file... The file has 14 staff:
218123  Smith, Steve
218133  Smith, Stan
218134  Flanders, Todd
218131  Simpson, Lisa
218136  Griffin, Petter
218145  Lwanson, Bruce
218146  Swanson, Bonnie
218181  Simpson, Bart
218139  Brown, Clev
218100  Goldman, Neil
218155  Hartman, Elmer
218187  Kapplan, Jim
218218  Machine, Vending
218235  Leonard, Lenny
---------------------------------------------------------
Back to main menu? [y: yes  |  n: no]:
```

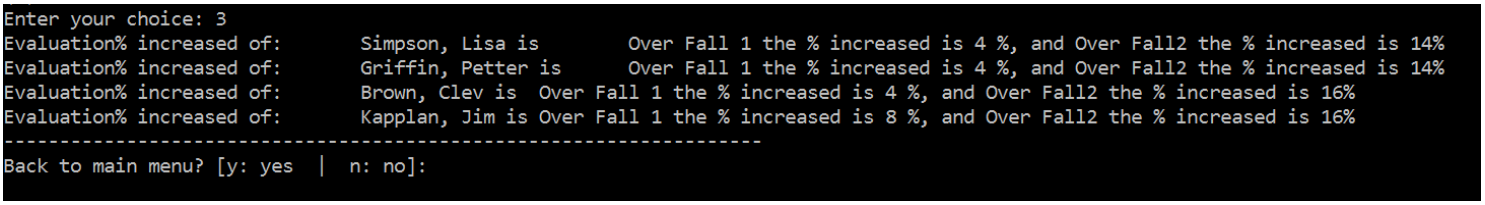
After each menu selection, your application must ask the user if he wants to go back to the main menu or not. If the user inputs "y" for "yes", then your application must print the main menu again.

2. **Print Staff Roster with Evaluation Grades:** this option will (1) read first file to get staffs IDs and their names,

   (2) Read the second file to get staffs IDs with the evaluation grades, and (3) print a list of staff(IDs, names, and evaluation grades) for those who have evaluation grads. See following figure:



3. **Print Staff Roster with Staff % Increase:** this option will find the % increased for each staff for Fall 1 (increase% from Spring 1 to Fall 1) and for Fall 2(increase% from Spring 1 to Fall 2). See the following figure:



4. **Exit:** exists the application

UMKC School of Computing and Engineering

Assignment 2 Due: September 24

Development Instructions

Your application will implement its functionalities using a **class** called "Staff" which has the following:

- A constructor that takes parameters for the Staff ID and staff name. The staff ID must be bigger than 0, otherwise the staff ID must be set to -1
- Getters and Setters for the staff ID, First name, Last name, and others
- A function called *FullName()* that returns a string consisting of the last name, a comma and space, then the first name; a Staff with first name "A" and last name "B" would return "B, A" from this function
- A method called *ReadData()*. It takes an *istream* passed by reference as its only parameter. It reads in an integer (Staff ID), first, and last name (in this order). This method returns a boolean value: true if all data was read successfully, false otherwise. Use this member function to read data from the first file.
- A method called *FindIncreasedpercentage()*that will output the increased evaluation percentage for each staff up till Fall 1 and then up till Fall2. It calculates the increased evaluation percentage of each staff from Spring 1 to Fall 1(((Fall1-Spring1)/5)*100). It will also calculate the increased evaluation percentage of each staff from Spring 1 to Fall 2(((Fall2-Spring1)/5)*100).

Submission:

- Your application must be implemented as a OOP (uses a class, header file, and a main program). Otherwise, it will receive no score.
- Zip up your entire project folder and submit the zip file to Blackboard by the deadline.