Exercise 5

Structures and Records: Pearson's Correlation Coefficient

Introduction:

Until this point, the programs that you've created would only deal with processing information sent by the source code or the keyboard input. Yet, a program may also acquire its information from a file on a disk. For this exercise, the source of your information come from a text file that is accessed sequentially.

Note that when reading a string data, the `getline()` function would be used, and it has the newline as the default delimiter. For example,

```
string str;
getline(inRec, str, '\t ');//reading a string with a tab delimiter
```

The correlation coefficient is used to compute the intensity and direction of the linear relationship between numerical variables X and Y. It is denoted by r. Although different correlation measures is available; the Pearson correlation coefficient would be used for this exercise. The formula for the correlation (r) is as follows:

$$r = \frac{1}{n-1}\sum \frac{(x-\bar{x})(y-\bar{y})}{S_x S_y}$$

Where:

n - the size of the population

x, y – sample data

$\bar{x}, \bar{y}$ - sample means

$S_x S_y$ - sample standard deviations

The **standard deviation** is the most commonly used measure of spread (dispersion). It indicates how compactly the values of a data set are clustered all around the mean. Given the equation

$$\sigma^2 = \frac{\sum(x-\mu)^2}{n}$$

Where:

x - element in a population
μ – mean
n – size of the population

Learning Outcomes:

- Read data on a sequential file and place it on a vector of structures.
- Compute the Pearson correlation result and store it on another sequential file.

Problem background

1. Create a text file named "StudentScore1.txt" and on the notepad with the following content.

```
John        99
Caleb       79
Timothy     82
Ruth        85
Hannah      78
```

   and "StudentScore2.txt" on the notepad with the following content:

```
Peter       89
Gabriel     67
Joshua      80
Grace       77
James       63
```

   Note: The delimiter between the student name and the score is a tab stop (\t) while the delimiter between the score and the student name is a new line (\n). Be sure that there is no stray character at the end of your record.

2. On your C++ source code, define a structure as follows.

```
struct student
{
   string studentName;
   double score;
};
```

3. Implement the functions below:

```
/* get the data from the text file and store it on the array of student
structure. Read a single record first then use a looping structure controlled
by an  ifStreamObj.eof() function to read all of the content of the text
file. The string filename parameter will be used as parameter on what record
to read on text file
*/
vector<student> readFromRecord (string fileName);

/*
 to compute the average score on the studScoreRec[] array with size s.
*/
double recAverage (vector<student> S);

/*
to compute the standard deviation of the record.
*/
double recSTDev(vector<student> S);

/* Compute the correlation coefficient of between vector <student> S1, and
vector <student> S2
*/
double recPearCorr(vector<student> S1, vector <student> S2);
```

```
/*
to write to a sequential text file named “scoreDescStat.txt”  with the
average, standard deviation, Pearson Correlation Coefficient as content.
*/
void writeResultToFile (double ave1, double ave2 double stDev1,double stDev2
double PearCorr);
```

4. Call all the functions created to function main