Computer Programming Language

【Fall, 2018】

Homework 7

Program A：Streams and File I/O (50%)

The text file **words.txt**, which is provided on our CEIBA course website, contains an alphabetically sorted list of English words. Note that the words are in mixed upper and lowercase.

Write a program that reads this file and finds the longest word that reverses to a different word. For example, "stun" reverses to make the word "nuts" but is only four letters long. Find the longest such word and show it on the screen.

■ *Web-Cat Submission Check:*

int answer1;   // Store the total number of words in the **words.txt** file.

int answer2;   // Store the number of characters of the longest word you find.

Program B：Classes (50%)

In this program you need to use Object-Oriented Programming approach to read a series of names and ages from a text file called **persons.txt**. You should not assume you know how many rows are in the file.

Define a class call **person** that will contain 3 private variables to hold the three data strFirstName, strLastName, and intAge. The **person** class should have public accessor and mutator functions, getFirstName, getLastName, getAge, setFirstName, setLastName, and setAge.

The main program should have a function called readData that reads in the data into an array of objects that contain firstname, lastname, and age. Since you don't know how many rows are there in the file, you need to apply dynamic memory allocation method to store the data read for later processing. The main program should also have functions called calculateAvgAge, calculateMaxAge, calculateMinAge, that have an array of objects passed to it, and the number of objects in the array. The functions calculate the integer average age, oldest age, youngest age, respectively. Show the calculated results on the screen.

■ *Web-Cat Submission Check:*

int answer1;   // Store the total number of persons list in the **persons.txt** file.

int answer2;   // Store the calculated average age.

int answer3;   // Store the calculated oldest age.

int answer4;   // Store the calculated youngest age.

Challenge Program C (Optional) : N-Queen Problem (Bonus Points 50%)

The N-queen is the problem of placing N chess queens on an N×N chessboard so that no two queens attack each other. For example, the following figure is a solution for a 8-queen problem.
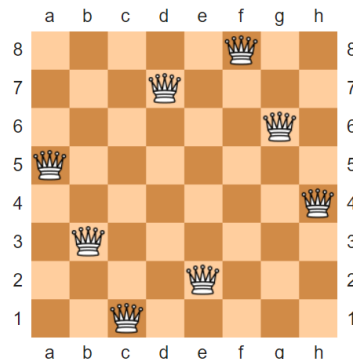
The N-queen problem can be solved by brute force or by a recursive backtracking algorithm. In backtracking algorithms you try to build a solution one step at a time. If at some step it becomes clear that the current path that you are on cannot lead to a solution, you go back to the previous step (backtrack) and choose a different path. Briefly, once you exhaust all your options at a certain step you go back. Here is a summary of how it works:

1.   Starting from the first column, check every row for a threat from queens above, below, left, right, or diagonal to it. Once you find a safe row for a given column, place a queen there.
2.   Recursively incrementing one column until you have either placed a queen in every column or you are unable to safely place a queen in a column. If latter occurs, you must backtrack to the last known good solution.
3.   Annotate the location of the last queen's row, remove the queen, and recursively calling on the previous column beginning one row passed where the last queen was.

Write a program to solve the 8-queen problem. Your program should save the solution in a file named **eight_queen.txt** as well as displaying it on the screen.

■ *Web-Cat Submission Check:*

   int answer1;   // Store the row number of the cell where the last queen is placed by the
                          program

Challenge Program D (Optional) : K-Means Clustering Algorithm (Bonus Points 50%)

K-means clustering is a type of unsupervised learning, which is used when you have unlabeled data (i.e., data without defined categories or groups). The goal of this algorithm is to find groups (clusters) in the data, with the number of groups represented by the variable K. The algorithm works iteratively to assign each data point to one of K groups based on the features that are provided. Data points are clustered based on feature similarity. The results of the K-means clustering algorithm are: the centroids of the K clusters (which can be used to label new data), and labels for the training data (each data point is assigned to a single cluster).

The pseudocode of the K-means clustering is summarized as follows:

1. Clusters the data into K groups where K is predefined.
2. Select K points at random as cluster centers.
3. Assign objects to their closest cluster center according to the Euclidean distance function.
4. Calculate the centroid or mean of all objects in each cluster.
5. Repeat steps 2, 3 and 4 until the same points are assigned to each cluster in consecutive rounds.

Write a program that reads the **Iris.data** text data file which contains the petal length and petal width of three speices of Iris flower in the first and second column. Let K = 3, use the K-means algorithm to label the data and saved them into a new text file named **Iris.out** with three labels A, B, and C, in the first column, respectively. Output the labeled data and the 3 cluster centroids on the screen.

■ *Web-Cat Submission Check:*

    int answer1;   // Store the number of data in cluster A.

    int answer2;   // Store the number of data in cluster B.

    int answer3;   // Store the number of data in cluster C.

Notes:

1. Please submit your programs (source codes) to the Web-CAT grading system website (http://140.112.94.129:8080/Web-CAT_1.4.0/WebObjects/Web-CAT.woa/)   before   **Jan.   3**. (3:30PM)

2. Late submission will have a penalty of 10% discount per day of your grade toward a minimum score of 60. No late submission over a week will be accepted.

3. Criteria of grading include: (1) Program functionality; (2). User interface; (3). Structure of the program; (4). Suitable comments; (5). Programming style; (6). Creativity. The proper use of functions in building a modular program is encouraged and will be part of the grading criteria in this homework.