CSC 2111 Lab 22

Objectives:

- Explore how to sort an array using insertion sort algorithm.
- Learn how to implement the binary search algorithm.

Part1:

Download **Lab22.cpp**. This program reads the candidate names and the number of votes received by each candidate of a local election from a file named **input.txt**. There will be eight lines in the input file. The program then outputs each candidate's name and the number of votes received, in Ascending order. (The sort algorithm uses insertion sort algorithm). Show your output in the program console (check the output format from sample output).

You need to implement the following function prototype:

```
void insertionSort(int vote[],string name[], int listLength);
```

Precondition: The function takes an integer array of vote[], a string array of candidate name[], and an integer listLength which represents the size of both arrays.

Post-condition: Integer array vote[], is sorted in ascending order and string array name[], is updated based on elements in sorted array vote[]. (Check the given sample output)

Part2:

After showing the output of Part 1 in the program console, the program asks the user to input a number as received votes to be searched and outputs corresponding candidate's name. (The binary search is used for searching purpose)

Complete the `binarySearch` function which accepts a sorted array resulted from insertion sort function in Part A, and returns the search result for the searched item.

You need to implement the following function prototype:

```
string binarySearch(const int vote[], const string name[], int
listLength, int searchItem)
```

Precondition: The function takes an array of sorted integer vote[], an array of string name[], an integer listLength which represents the size of the arrays, and an integer searchItem which to be searched.

Post-condition: If the searchitem is found in the vote[] its corresponding candidate's name from the name[] is returned else the message "There is no candidate with the inputted votes!" is returned. (Check the given sample output)

Submit one single cpp file.

Sample Output: