Assignment 7
CSIS 113A – Duffie (F2016)

Use two array declarations in main() of your own making. One array should be *string*, and the second array should be of type *int* or *double*. Each array should have a minimum of 10 elements.

For example -

```
string myNames[] = {"Bob","Tina","Wes","Geoff","Jan",
"Bob","Mitch","Susan","James","Grace"};

int myScores[] = { 100,102,95,89,103,78,98,102,65,73};
```

Data concept for this assignment is up to you. Try and think of something unique (definitely do not use names and scores). You may be able to find some comma delimited data that you can use.

Here are a few ideas –

- o Cities with Populations
- o Movies with Year
- o Teams with Number of Wins (and Losses)
- o Items with Prices
- o Snacks Description with Calories
- o Occupations with Salaries

Create a simple C++ application that utilizes the specified functions to 1) display the contents of both arrays, 2) display the average numeric value associated with the numeric array, 3) display the maximum and minimum numeric values (along with associated string items for each), and 4) find a numeric value based on keyboard input of a string value (keyboard entry of which name to find for example). This function should return an index value which you can use to display the associated numeric value for this item. In addition to the functions described above, your application should implement one additional function of your own design that passes at least one array argument.\

Have fun with this!

[sample function prototypes next page]

Your program should implement and use functions similar to the following (obviously your array names will be different) –

```
Void displayAll(int score[], string names[], int size);
```

      displays all names and scores in the two arrays

```
int avgScore(int score[], int size);
```

      returns the average of the scores

```
int maxScore(int score[], int size);
int minScore(int score[], int size);
```

      returns the high and low scores

```
int findNameIndex(string names[], string find, int size);
```

      returns the index of the array item find. This index can then be used to find the corresponding score. For example -


```
cout<<myScores[findNameIndex(myNames,"Tina",10)]<<endl;
```

Write <u>one additional function</u> of your own design that passes at least one array argument.

Notes: *The only function that should utilize cout for output is displayAll. All other cin and cout statements should be in main(). Note that findNameIndex return the index value of the significant item, and not the item value. The index can then be used in main() to display the corresponding pair item.*