Prog-07 (Take Home) **Solving Problems using 300 Knowledge** **Due: before midnight Oct/31**

It is extremely important that you understand the programming assignments; please participate in the classroom discussion. You are expected to solve the programming assignments individually. You are encouraged to discuss with your friends. However, **do not copy**. Follow the submission and grading policies for full points.

***Sorting* in C**

The primary focus is to understand and implement *Bubble Sort and Insertion Sort* in C.

Q1 (Algorithm – 10%): For the given array show the items after every pass along with the number of operations for insertion sort. Calculate the number of operations to complete the insertion sort. Original given array: {5, 9, 6, 7, 3, 2, 1}

> Sample (insertion sort):
> Pass 2: items {5, 9, 6, 7, 3, 2, 1} operations 0
> …

Q2 (Code – 90%): An array A of N integers (where, N = 1, 2, 3, …, 100000) should be used. Write the following functions in C.

Mandatory functions:

```
char UserMenu( char *opt ); /* display menu and receive option */
void PopulateArrayDes( int *A ); /* populate array A in descending order */
void CopyArrays( int *A1, int *A2 ); /* copy array A1 into array A2 */
void DisplayArray10Items( int *A ); /* display the first 10 items in array A */
void BubbleSortAsc( int *B ); /* sort array B in ascending order using bubble sort algorithm */
void InsertionSortAsc( int *I ); /* sort array I in ascending order using insertion sort algorithm */
int CalculateTimeBS( int *A ); /* return the time needed to bubble sort an unsorted array A */
int CalculateTimeIS( int *A ); /* return the time needed to insertion sort an unsorted array A */
void ExitProg( void ); /* normal exit */
```

Function “main”

Driver program

Function “UserMenu”

Display user friendly menu

Sample (menu):

```
Welcome to CS460 (Fall 2013) HW-07 Solution!
Please see your options:
1) Exit
2) Populate array
3) Copy array
4) Display array (first 10 items)
5) Bubble Sort Ascending
6) Insertion Sort Ascending
7) Calculate Time for Bubble Sort
8) Calculate Time for Insertion Sort
Please enter your option >
```

Function "PopulateArrayDes"

Populate array A in descending order. For N = 100000, the initial array should be populated such that A[0] = 0, A[1] = 100000, A[2] = 99999, …, A[100000] = 1.

Function "CopyArray"

Copy the first array into the second array.

Function "DisplayArray10Items"

Display the first 10 items in the given array.

Sample (initial array in descending order):

```
Item bbbbbbb1 Value bb100000
Item bbbbbbb2 Value bbb99999
Item bbbbbbb3 Value bbb99998
…
```

Sample (sorted array in ascending order):

```
Item bbbbbbb1 Value bbbbbbb1
Item bbbbbbb2 Value bbbbbbb2
Item bbbbbbb3 Value bbbbbbb3
…
```

Function "BubbleSortAsc"

Sort the given array in ascending order using bubble sort algorithm.

Function "InsertionSortAsc"

Sort the given array in ascending order using insertion sort algorithm.

Function "CalculateTimeBS"

Calculate and return the time needed to bubble sort the given unsorted array.

Sample (sorting time):

```
Start date/time: 2013-10-23 17:33:48
End date/time: 2013-10-23 17:33:57
Time elapsed for the bubble sort: 9 sec
```

Function "CalculateTimeIS"

Calculate and return the time needed to insertion sort the given unsorted array.

Sample (sorting time):

```
Start date/time: 2013-10-23 17:35:56
End date/time: 2013-10-23 17:36:03
Time elapsed for the insertion sort: 7 sec
```

Function "ExitProg"

Normal exit.

NOTE: The following sample code is available to print "time".

/usr/users/User11/drzaman/CS460/Samplecodes/date_time_upto_sec.c