EXAM 3 (RASM-5)

Re-submit Assignment

Due Wednesday by 5:50pm **Points** 200 **Submitting** a file upload
Available until May 20 at 5:50pm

For this assignment, you will be creating a Menu driver program in C that executes the following menu. The purpose of the assignment is demonstrate the calling Assembly language macros from C. Do not load automatically, only via the menu.

input file "input.txt" (200,000 random integers that you create). I recommend making a C++ file that randomly generates the 200,000 integers and outputs to a file.

Finally. Every time you sort, you need to save the array to its respective text file (i.e. "c_output.txt" and "a_output.txt") so that I can verify that your sorting algorithms work.

```
               MASM5 C vs Assembly
              File Element Count: 200000
-------------------------------------------------
C        Bubblesort Time: 0 secs
Assembly Bubblesort Time: 0 secs
-------------------------------------------------
<1> Load input file (integers)
<2> Sort using C Bubblesort algorithm
<3> Sort using Assembly Bubblesort algorithm
<4> Quit
```

Testing: If you read in the input file and then immediately save, the output file should be identical to the input file in every way (i.e. # of bytes).

Upload your .s file and a screen shot running the code.

```
void BubbleSort (int a[], int length)
{
        int i, j, temp;

    for (i = 0; i < length; i++)
    {
        for (j = 0; j < length - i - 1; j++)
        {
            if (a[j + 1] < a[j])
            {
                temp = a[j];
                a[j] = a[j + 1];
```

```
                a[j + 1] = temp;
            }
        }
    }
}
```

You will need to link example.....

```
> g++ -c RASM5.cpp
> as -g bubbleSort.o bubbleSort.s
> g++ -o myprog.exe RASM5.o bubbleSort.o
```

s/

Prof. B.

****** Extra Credit 20pt *******

Add the insertionSort fn's to the menu set.

```
                MASM5 C vs Assembly
File Element Count: 200000
-----------------------------------------------
C bubbleSort Time: 0 secs
Assembly bubbleSort Time: 0 secs

C insertionSort Time: 0 secs
Assembly insertionSort Time: 0 secs
-----------------------------------------------
<1> Load input file (integers)
<2> Sort using C bubbleSort algorithm
<3> Sort using Assembly bubbleSort algorithm
<4> Sort using C insertionSort algorithm
<5> Sort using Assembly insertionSort algorithm
<6> Quit
```