

CSE 410 Homework 3

Spring 2008

Due: Midnight, Wednesday, 4/23/2008

Use electronic submission via the Catalyst tool:

<https://catalysttools.washington.edu/collectit/dropbox/telmas/2218>

Write a MIPS assembly program that sorts an array in memory. The sorting algorithm should be a simple insertion sort. You may search the web for an in-depth description (and even animations) of insertion sort, but here is the algorithm in pseudo code.

Pseudo code:

```
Given an array of numbers: array, and an array length: length

for i from 1 to length

    set A be array[i]

    set j be i-1

    while j>=0 and array[j] > A

        set array[j+1] equal to array[j]

        set j be j-1

    set array[j+1] be A
```

Put the following declarations at the beginning of your code. These declarations will initialize memory to contain the array and length at startup. The array and length can then be read and modified using loads and stores.

```
        .data
array:   .word   1 7 2 8 3 6 4 5 10 9
length:  .word   10
```

You will want to use SPIM to test your program to make sure it works properly. Your code should be well-organized so that it is clear what each section of your code does. You do not need to optimize your assembly code, but make sure it is properly commented.

What you need to do:

1. Download the java skeleton code here:
<http://www.cs.washington.edu/education/courses/410/08sp/hw/skeleton.java>
2. Rename the java file and implement the insertion sort.
3. Now, you should have a better understanding on insertion sort.
Download the skeleton code here:
<http://www.cs.washington.edu/education/courses/410/08sp/hw/skeleton.s>
4. Implement the sorting algorithm in assembly code
5. **(Extra Credit)** Allow the sort function to accept array size as the parameter. Hence, you do not need to load the length of the array from the data section.