Project 1

Chase Farmer

Bubble Sort Algorithm

First Pass:

(**1 32** 10 8 33) -> (**1 32** 10 8 33)

(1 **32 10** 8 33) -> (1 **10 32** 8 33) 32 > 10 swap

(1 10 **32 8** 33) -> (1 10 **8 32** 33) 32 > 8 swap

(1 10 8 **32 33**) -> (1 10 8 **32 33**)

Second Pass:

(**1 10** 8 32 33) -> (**1 10** 8 32 33)

(1 **10 8** 32 33) -> (1 **8 10** 32 33) 10 > 8 swap

(1 8 **10 32** 33) -> (1 8 **10 32** 33)

(1 8 10 **32 33**) -> (1 8 10 **32 33**)

Third Pass: (must do a full pass just to check if it’s all in order)

(**1 8** 10 32 33) -> (**1 8** 10 32 33)

(1 **8 10** 32 33) -> (1 **8 10** 32 33)

(1 8 **10 32** 33) -> (1 8 **10 32** 33)

(1 8 10 **32 33**) -> (1 8 10 **32 33**) No Swaps, correctly ordered.

//Bubble Sort Java Code

void bubbleSort(int arr[])

{

int n = arr.length;

for (int i = 0; i < n-1; i++)

for (int j = 0; j < n-i-1; j++)

if (arr[j] > arr[j+1])

{

// swap arr[j+1] and arr[i]

int temp = arr[j];

arr[j] = arr[j+1];

arr[j+1] = temp;

}

}

Register Map

r0 – Saved for Returning the Array

r1 – Saved for Returning the Array

r2 – Array Size

r3 – Current Element Value (ex. array[j])

r4 – Offset Element Value (ex. array[j+1])

r5 – Current Element # (Position in the Array for Current Element)

r6 – Offset Element # (Position in the Array for Offset Element)

r7 - # of Swaps taken place in the current iteration of the inner loop

r8 – Array is Loaded and Stored