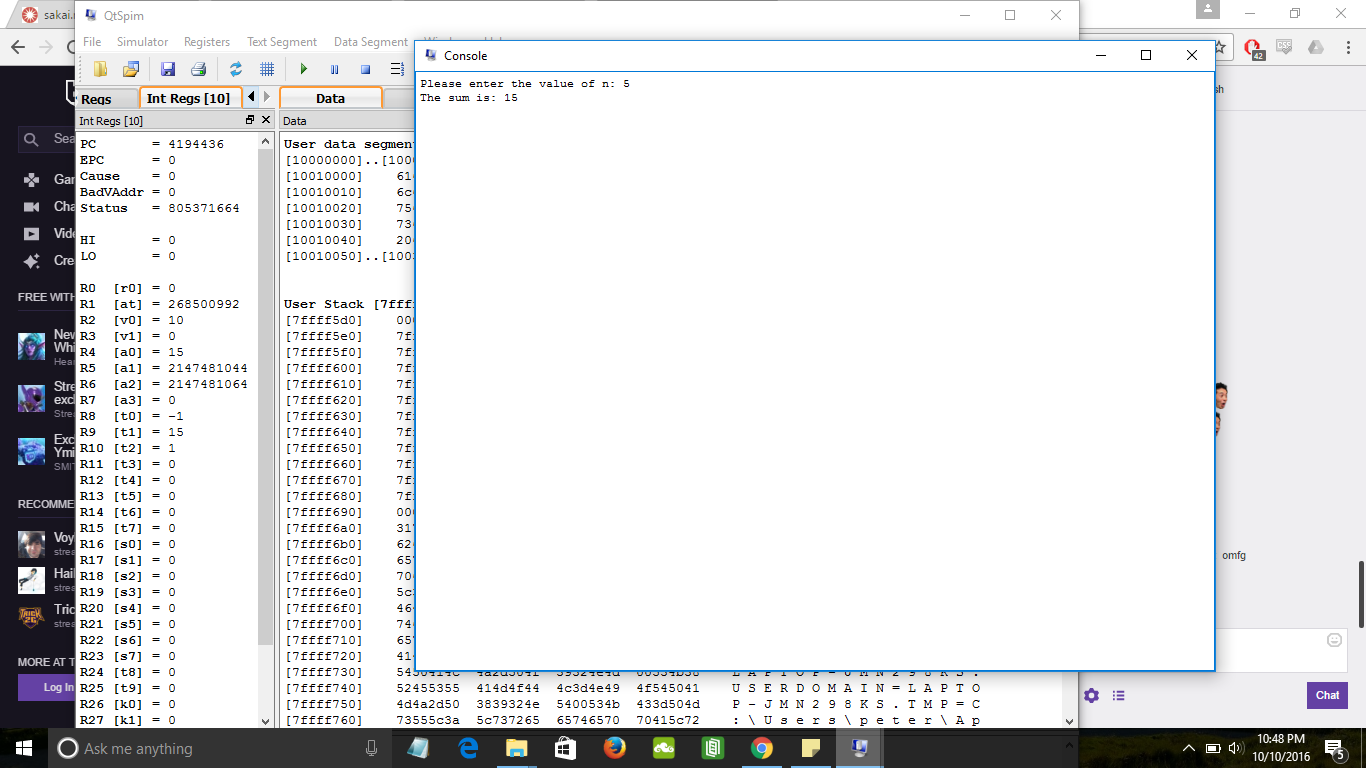
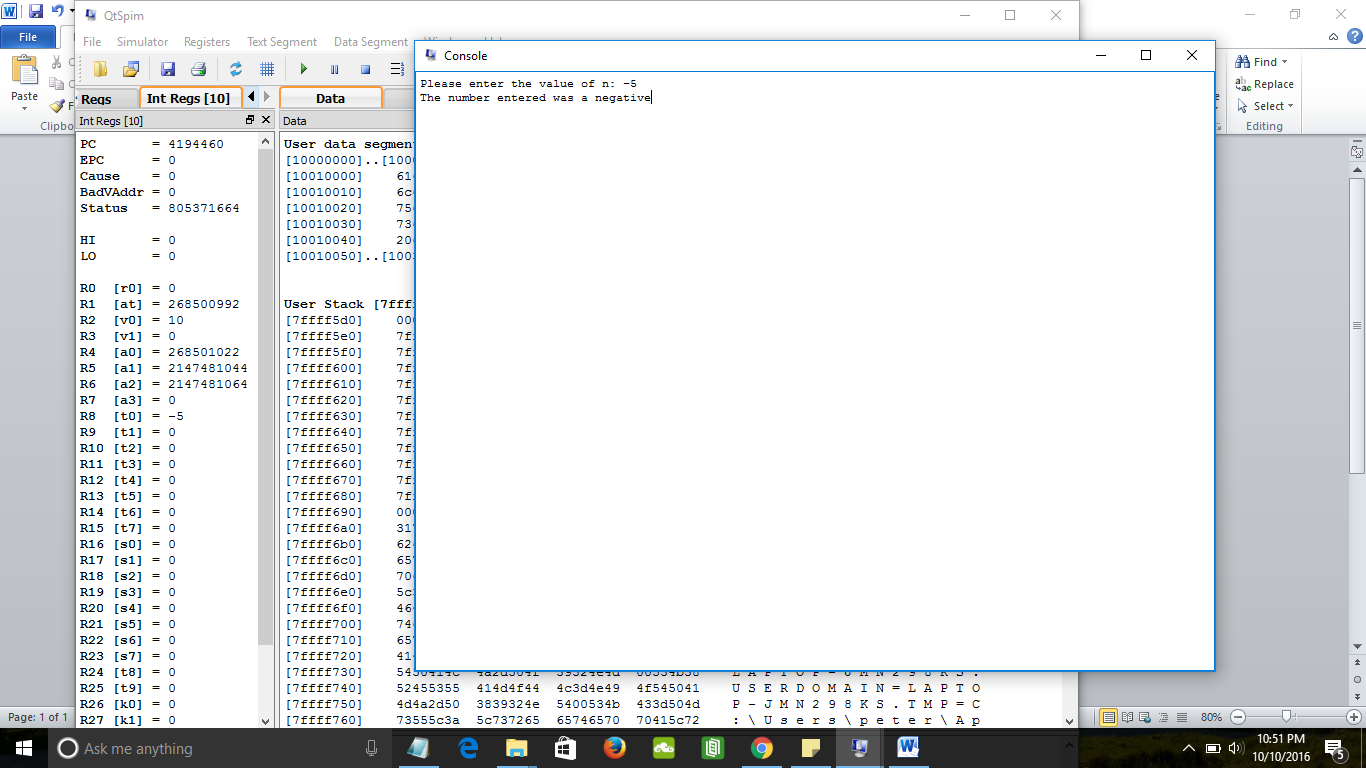
Peter Luo 157004870 Comp Arch Lab 2

Assignment 1:

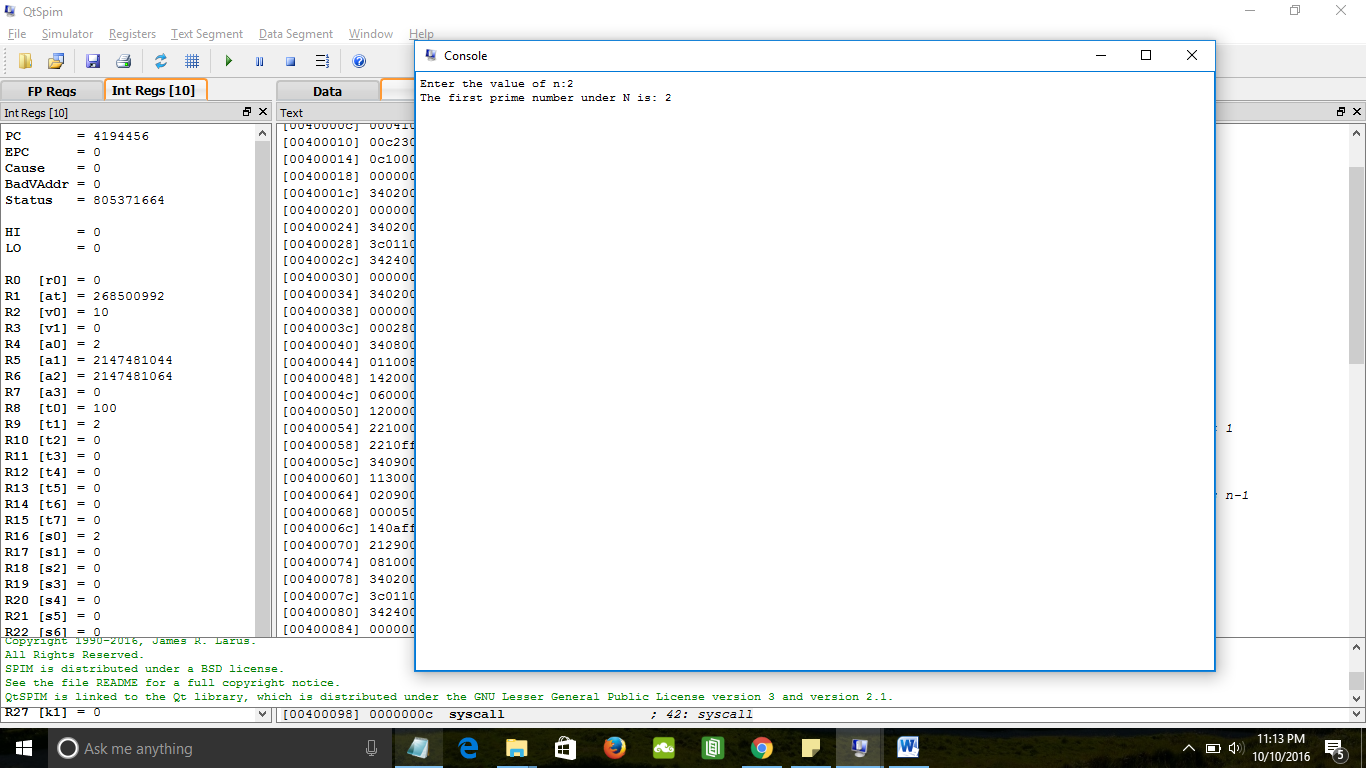
The code works properly



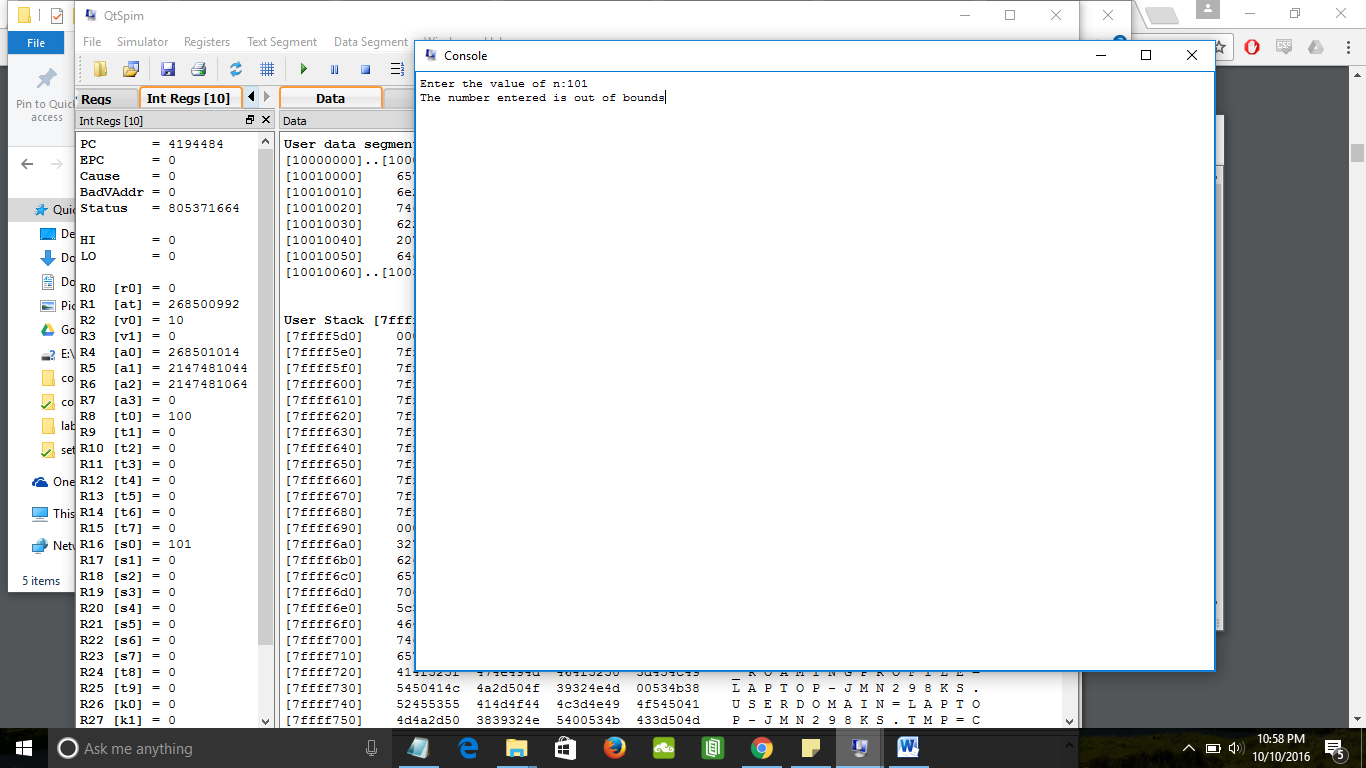
Also catches when an invalid input is entered

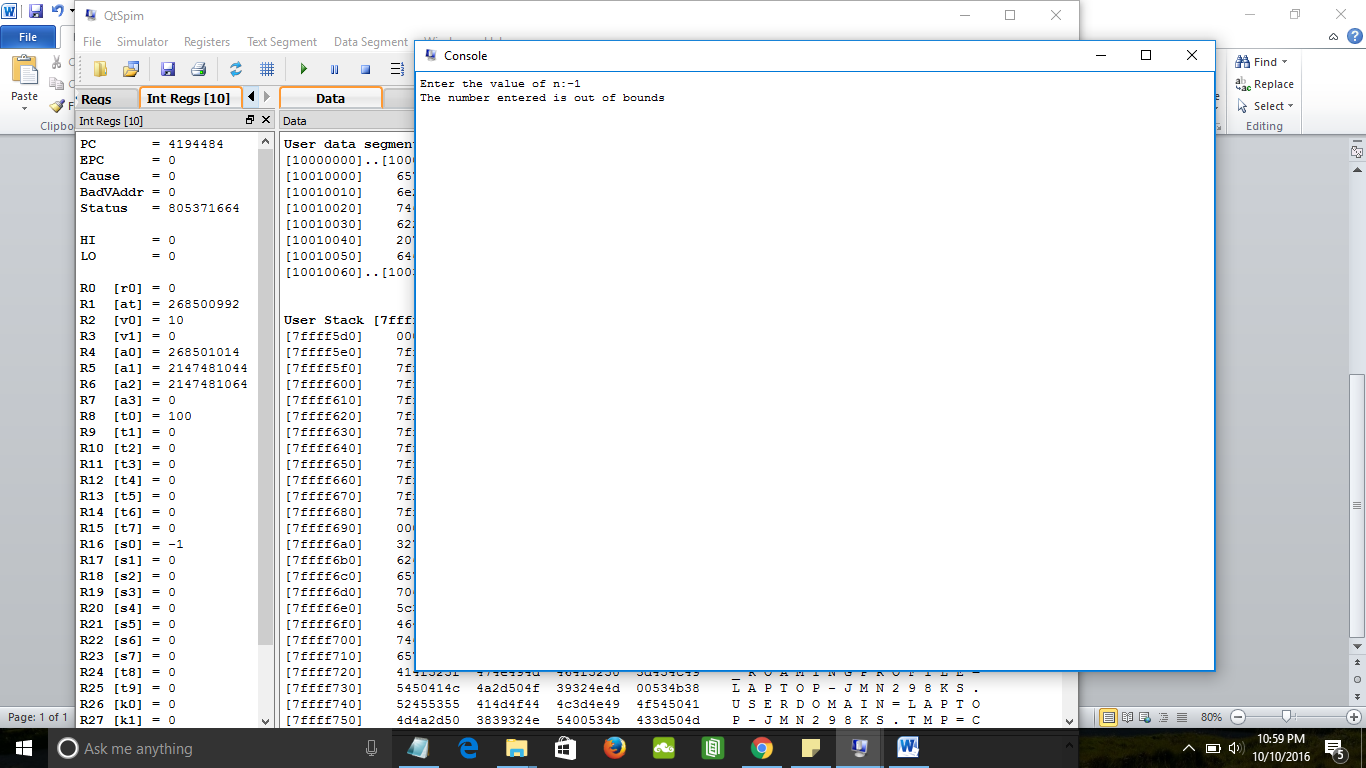


Assignment 2:

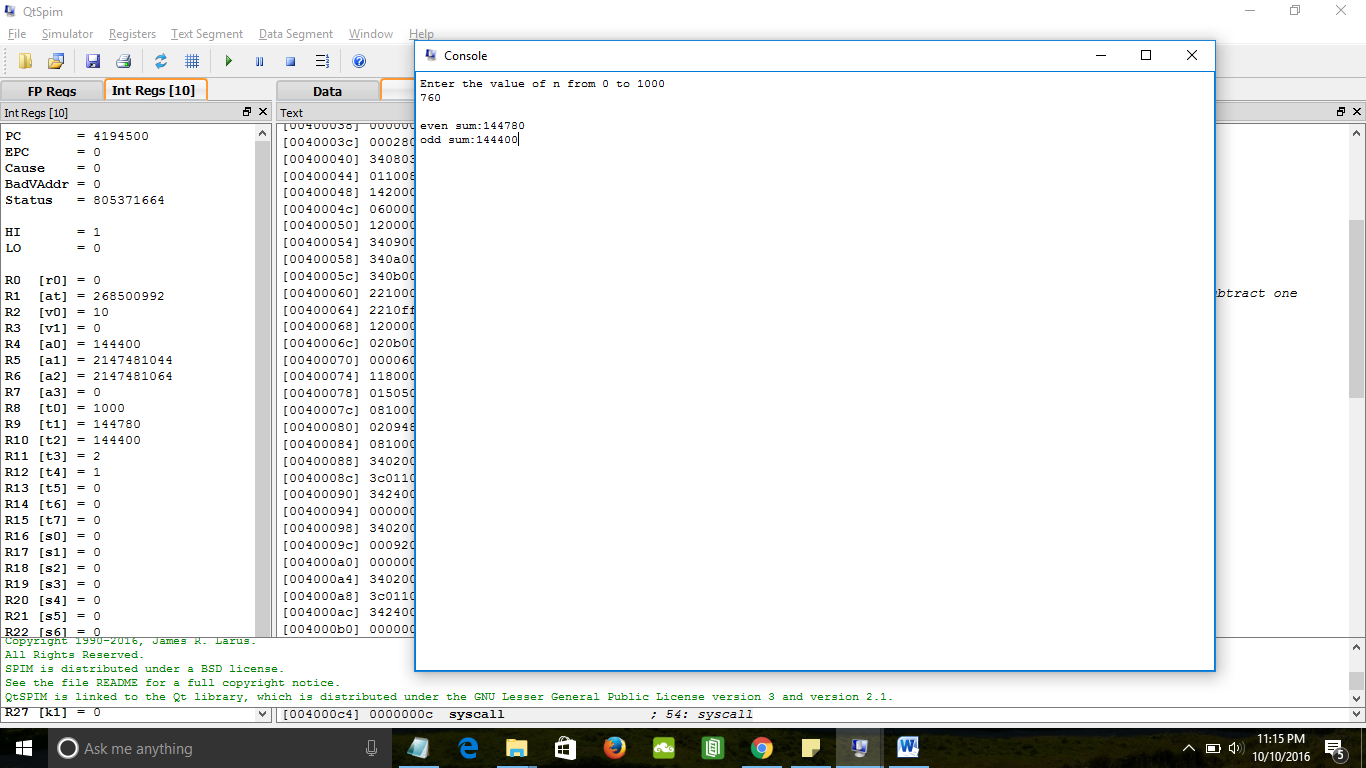


Catches cases out of bound of 1 to 100

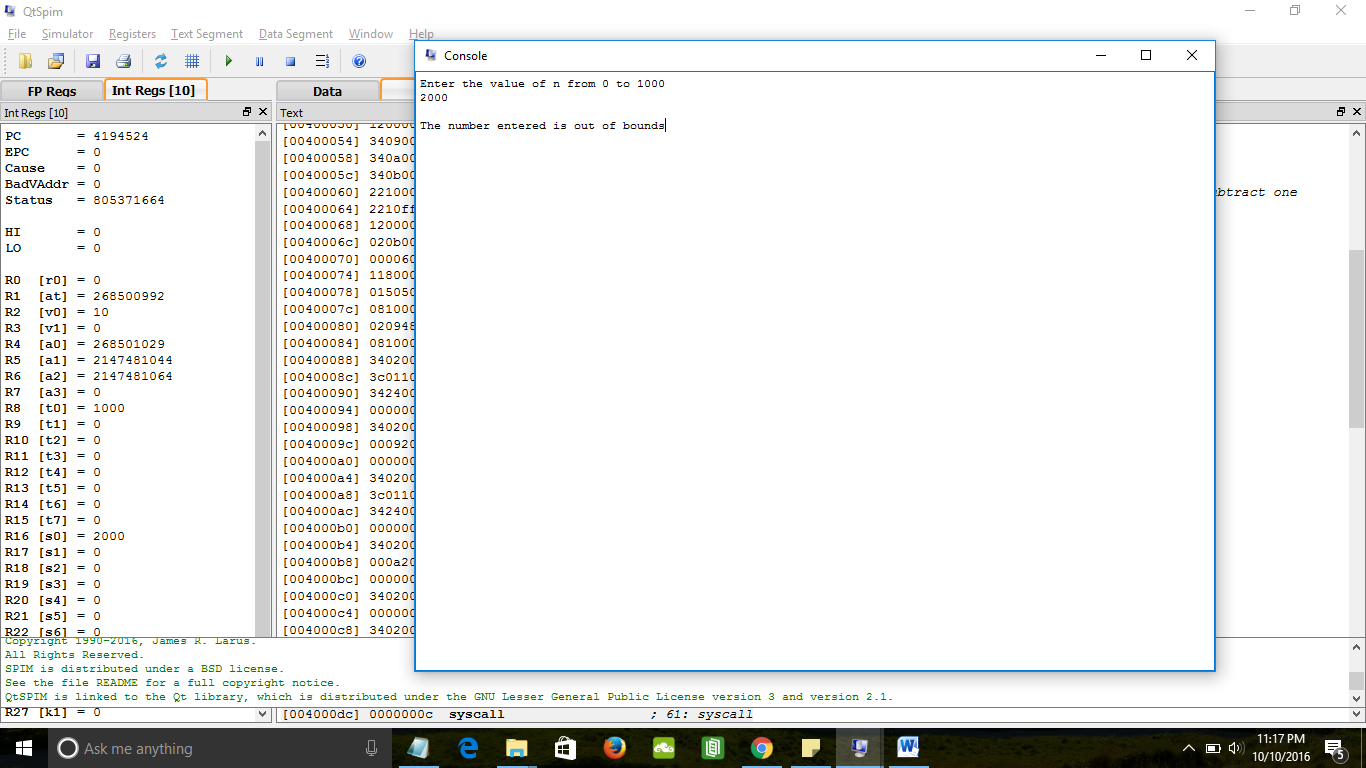




Assignment 3:

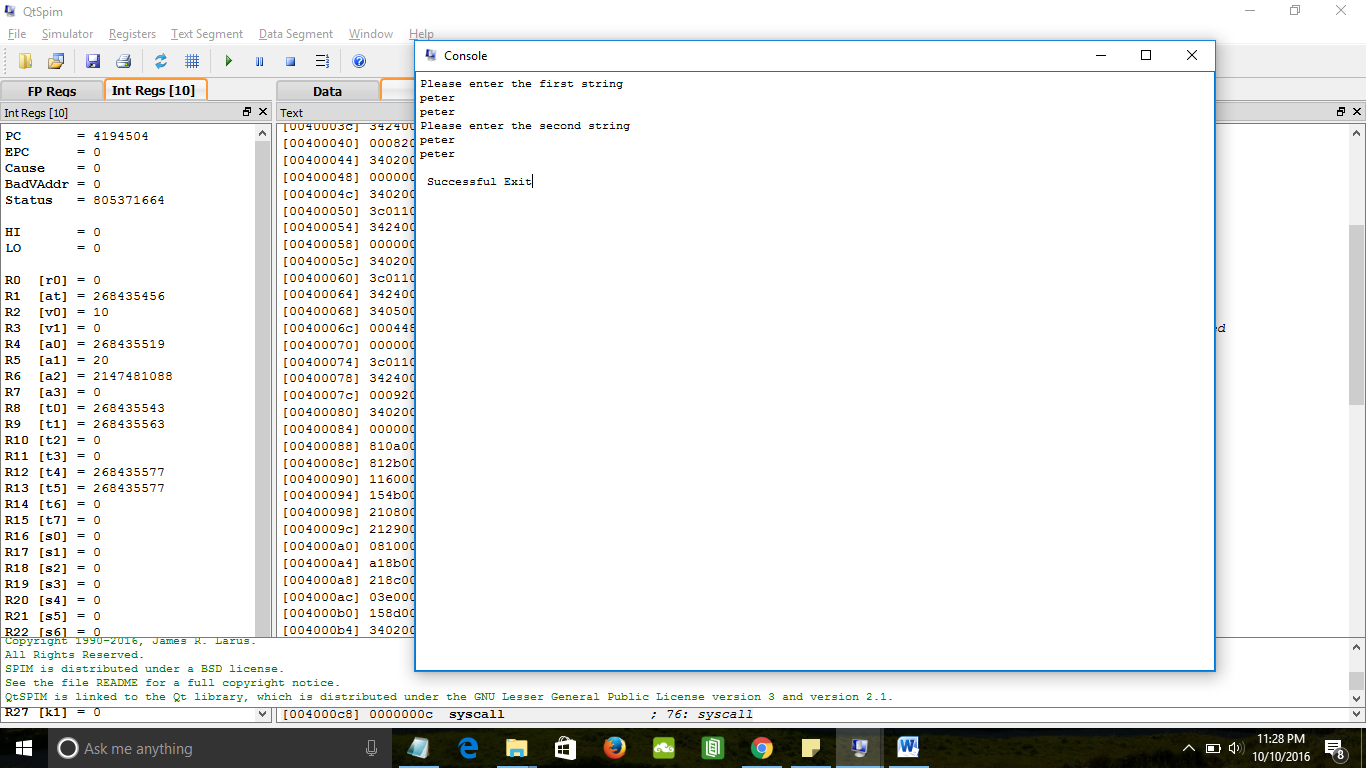


Catches out of bounds



Assignment 4: I programed this to also output the input value for each string that is stored.

Successful exit:



I spent a long time stuck on a problem. I had tried to use jr $ra to return to where the branch instruction called. However, branch instructions are not jal instructions and do not link to itself in $ra. Therefore, I created a label to jump to instead.

Incorrect on the first try:

String 1: abd

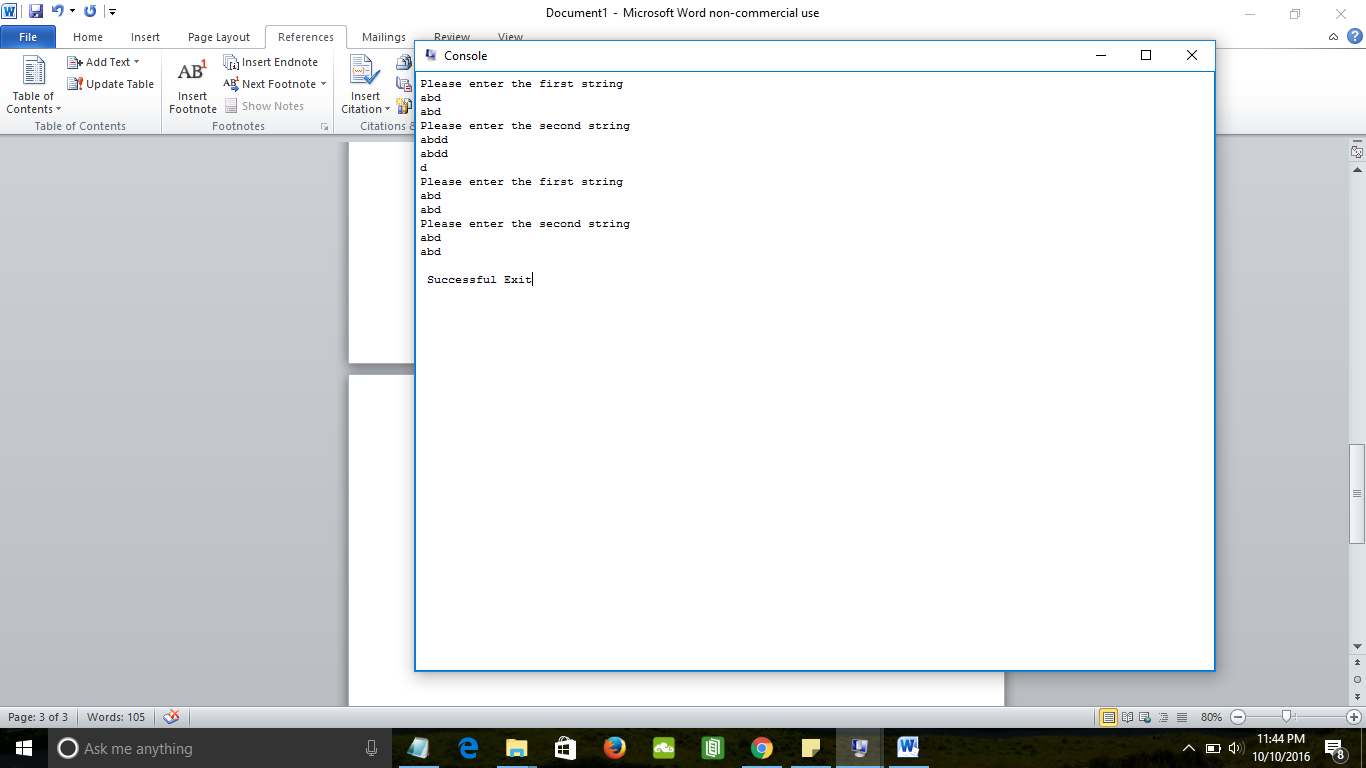
String 2: abdd

Mismatched: d

Then restarts to allow a correct version to be called.

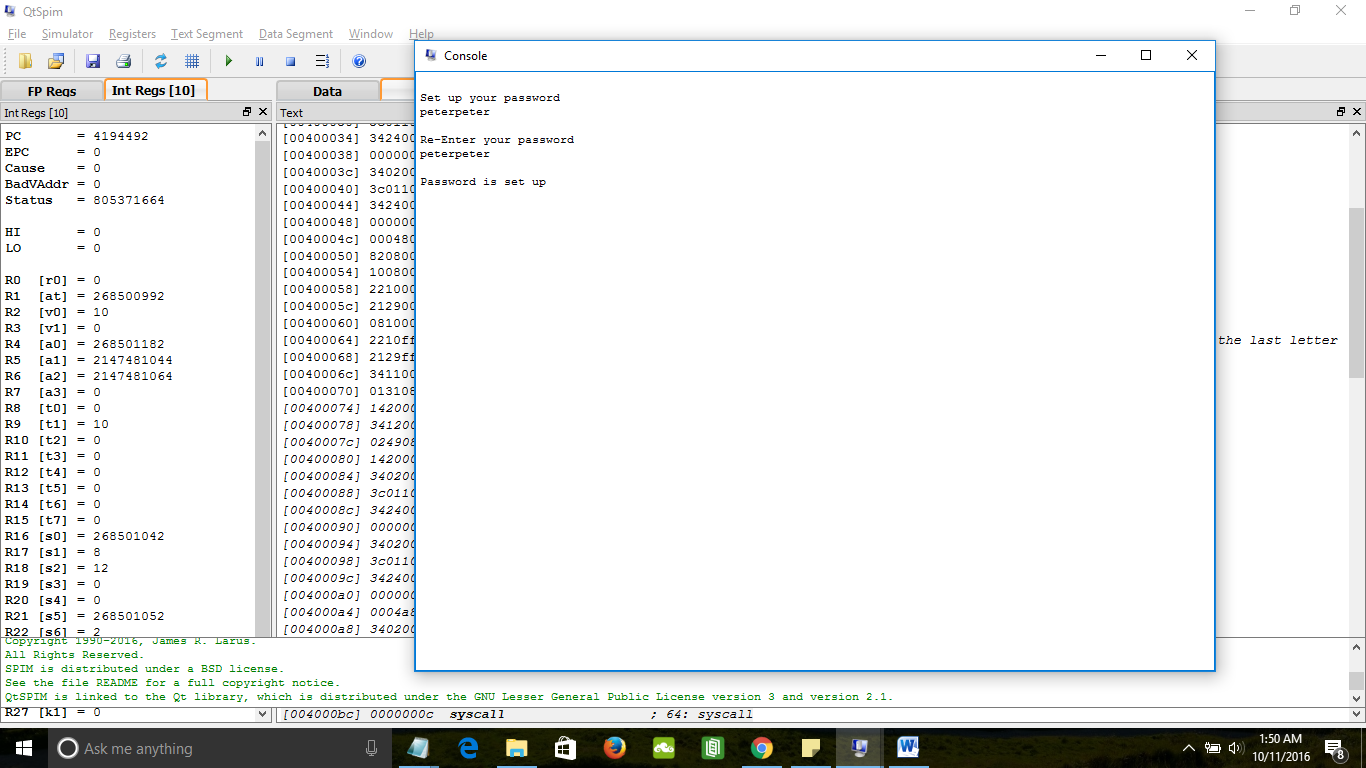
Correct version is called.

Successful exit

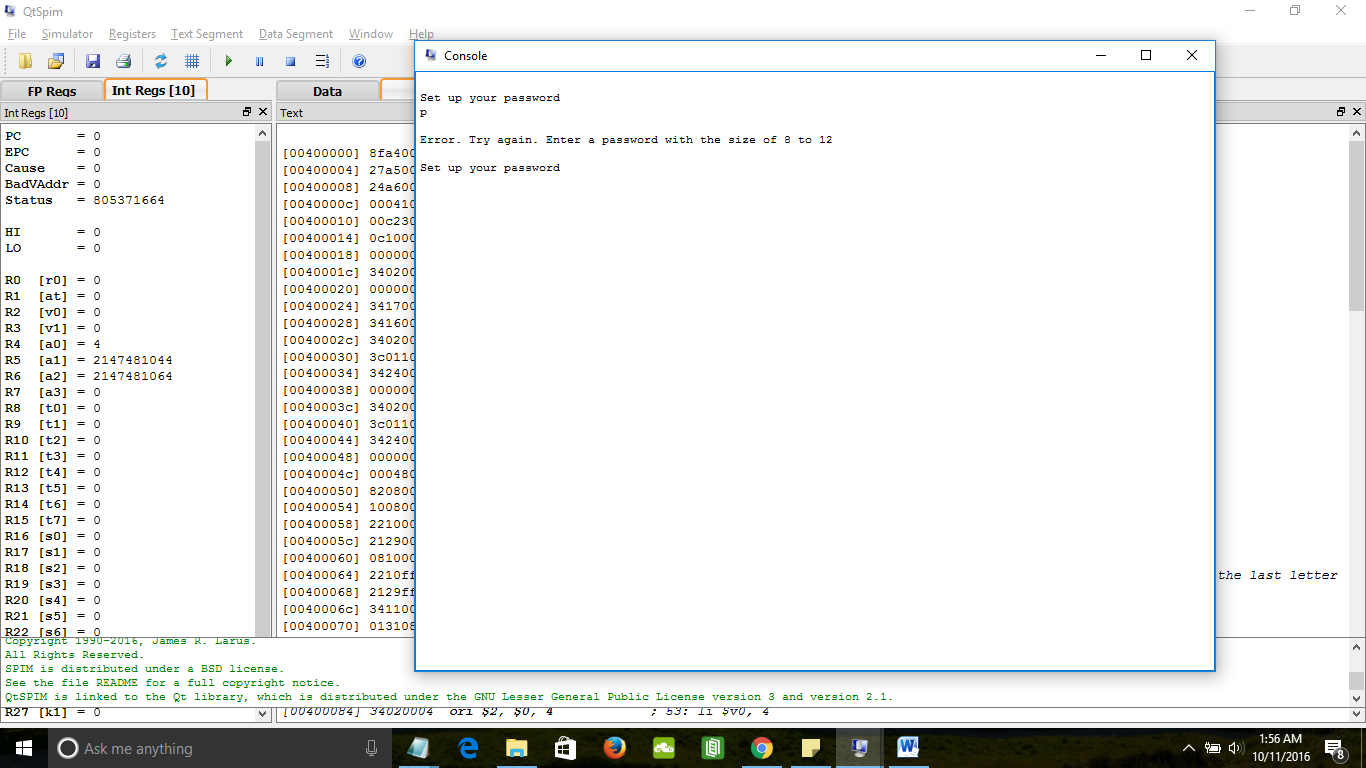


Assignment 5:

Successful Case:



Asks for password again



Assignment 6:

I couldn’t get my code to work. It kept giving me errors. The pseudocode that I would use:

Since size of arrays are not fixed, they are like vectors.

Int a = input integer;

If (a<0) exit;

Boolean found = false; int index;

New Vector2;

For int I; array[I] does not equal 0xF ;i++{

if array[i] == a; // number is in array

print “number at index ” + i;

else

Vector2 has the integer value of array[i] inserted by looping through the values until it reaches a value that is bigger than it. Then it goes back 4 bytes and is stored there while all the words after this location in the Vector2 is pushed back by 4 bytes. Insertion sort.

}

//at this point, the number isn’t in the array.

Same method as before, add int a into vector2 with insertion.

Print Vector2 // the sorted array with the entered number included too