Genevieve Leach

CS 264

2 February 2017

Lab 2

**Code:**

.data

array: .space 80

prompt1: .asciiz "Please enter an integer: "

prompt2: .asciiz "\nSmallest Value: "

prompt3: .asciiz "\nLargest Value: "

prompt4: .asciiz "\nAmount of Numbers Divisible by 4: "

.text

.globl main

main: la $t0, array

li $t1, 0 #number of integers entered

li $t2, 20 #loop counter

jal promptloop

li $t2, 20 #loop counter

li $a1, 0 #initialize a1

jal smallestLargest

#begin print for smallestLargest

la $a0, prompt2

li $v0, 4 #syscall 4 - print string

syscall

move $a0, $s0

li $v0, 1 #syscall 1 - print integer

syscall

la $a0, prompt3

li $v0, 4 #syscall 4 - print string

syscall

move $a0, $s1

li $v0, 1 #syscall 1 - print integer

syscall

li $s3, 0 #initialize a1

#end print for smallestLargest

li $a1, 0 #initialize a1

li $t2, 20 #loop counter

jal divisible

#begin print for divisible

la $a0, prompt4

li $v0, 4 #syscall 4 - print string

syscall

move $a0, $s3

li $v0, 1 #syscall 1 - print integer

syscall

#end print for divisible

end: li $v0, 10 #syscall 10 - end program

syscall

promptloop: la $a0, prompt1

li $v0, 4 #syscall 4 - print string

syscall

li $v0, 5 #syscall 5 - read integer

syscall

sw $v0, 0($t0) #store in array

addi $t1, $t1, 1 #increment integer counter

addi $t0, $t0, 4 #next array position

bne $t1, $t2, promptloop#loop back to beginning

jr $ra

smallestLargest:beqz $t2, return #if loop counter = 0, return to main

la $t5, array($a1) #set to array[i]

lw $s2, 0($t5) #load array[i]

blt $s2, $s0, slsmaller #if array[i] is < $s0, set smallest

bgt $s2, $s1, sllarger #if array[i] > $s1, set largest

#else, do nothing

addi $t2, $t2, -1 #decrement loop counter

addi $a1, $a1, 4 #i+1

b smallestLargest

slsmaller: move $s0, $s2

addi $t2, $t2, -1 #decrement loop counter

addi $a1, $a1, 4 #i+1

b smallestLargest

sllarger: move $s1, $s2

addi $t2, $t2, -1 #decrement loop counter

addi $a1, $a1, 4 #i+1

b smallestLargest

divisible: li $s3, 0 #set counter of divisible numbers to 0

divloop: beqz $t2, return #if loop counter = 0, return to main

la $t5, array($a1) #set to array[i]

lw $s2, 0($t5) #load array[i]

beqz $s2, notdiv #if 0, not divisible by 4

rem $a0, $s2, 4

bnez $a0, notdiv #if has a remainder, not divisible by 4

addi $s3, $s3, 1 #if here, number is divisible by 4; #increment count

addi $t2, $t2, -1 #decrease loop counter

addi $a1, $a1, 4 #i+1

b divloop

notdiv: addi $t2, $t2, -1 #decrease loop counter

addi $a1, $a1, 4 #i+1

b divloop

return: jr $ra

**Output:**

Please enter an integer: 04

Please enter an integer: 4

Please enter an integer: 1

Please enter an integer: -5

Please enter an integer: 0

Please enter an integer: 422

Please enter an integer: 100

Please enter an integer: 976

Please enter an integer: 800

Please enter an integer: -70

Please enter an integer: -46

Please enter an integer: 4

Please enter an integer: 5

Please enter an integer: 6

Please enter an integer: 7

Please enter an integer: 8

Please enter an integer: 1

Please enter an integer: 2

Please enter an integer: 3

Please enter an integer: 46

Smallest Value: -70

Largest Value: 976

Amount of Numbers Divisible by 4: 7