CptS260: Introduction to Computer Architecture

Coding Problem #1 Solution

School of Electrical and Computer Engineering
Spring 2022

Calculator

```
# Registers:
# $t0 - first integer
# $t1 - second integer
# $t2 - opcode; then re-used for result
# $v0 - syscall parameter and return value
# $a0 - syscall parameter

###### syscall that has been used in this program #######
########### 1) syscall 4 to print a string
########### 2) syscall 5 to read an integer
########### 3) syscall 1 to print an integer
########### 4) syscall 10 to exit the simulator

.data
############### Data specified in memory ##################
Integ1: .asciiz "Enter the first integer: "
Integ2: .asciiz "Enter the second integer: "
Op: .asciiz "Enter the operation type (add=0, sub=1, multiply=2):"
Result: .asciiz "The result is "
############################################################

.text
.global main
main:
# syscall to print the string and ask for the first number
##############################################################
li $v0, 4
la $a0, Integ1
syscall
##############################################################

# syscall to read the first integer number from the console
##############################################################
li $v0, 5
syscall
move $t0, $v0
##############################################################
```

```
# syscall to print the string and ask for the second number
################################################################
li $v0, 4
la $a0, Integ2
syscall
################################################################

# syscall to read the second integer number from the console
################################################################
li $v0, 5
syscall
move $t1, $v0
################################################################

# syscall to print the string and ask for the operation type
#################################################################
li $v0, 4
la $a0, Opcode
syscall
#################################################################

# syscall to read the operation from the console
li $v0, 5
syscall
move $t2, $v0
#################################################################

# syscall to print the string before printing the result
#################################################################
li $v0, 4
la $a0, Result
syscall
#################################################################

#################################################################
beq $t2, $zero, ADD_L
beq $t2, 1, SUB_L
#################################################################

######### Multiply operation when opcode is not 0 and 1 #########
mul $t2, $t0, $t1
j RESULT_L
#################################################################

############# Addition operation when opcode is 0 ###############
ADD_L:
     add $t2, $t0, $t1
     j RESULT_L
#################################################################

############# Addition operation when opcode is 1 ###############
SUB_L:
     sub $t2, $t0, $t1
```

```
#################################################################

####################### Print final result ######################
RESULT_L:
     move $a0, $t2
     li $v0, 1
     syscall
#################################################################

######################Exit simulator#############################
li $v0, 10
syscall
#################################################################
```

CptS260: Introduction to Computer Architecture

Coding Problem 2 Solution

School of Electrical and Computer Engineering
Spring 2022

```
#### array is saved to address space from 10010000 to 10010024
###### syscall that has been used in this program #######
########### 1) syscall 4 to print a string
########### 2) syscall 1 to print an integer
########### 3) syscall 10 to exit the simulator


.data
############### Data specified in memory #######################
array_a: .word 11, 12, -10, 13, 9, 12, 14, 15, -20
funct1: .asciiz "The maximum is: "
funct2: .asciiz "\nThe summation is: "
################################################################

.text
.globl main

main:
# syscall to print the the string to show the output of maximum
##############################################################
li $v0, 4
la $a0, funct1
syscall
###############################################################

jal maxf
move $a0, $v0

# syscall to print the first integer number from the console
##############################################################
li $v0, 1
syscall
###############################################################

# syscall to print the the string to show the output of maximum
##############################################################
li $v0, 4
la $a0, funct2
syscall
```

```
###############################################################

jal sumf
move $a0, $v0

# syscall to print the first integer number from the console
##############################################################
li $v0, 1
syscall
###############################################################

##################### Exit simulator ##########################
li $v0, 10
syscall
#################################################################


maxf:

li $t5, 1
li $t3, -10000
li $t1, 10
la $t0, array_a
loop1:
lw $s0, 0($t0)
ble $s0, $t3, cont
addi $t3, $s0, 0
cont:
addi $t0, $t0, 4
sub $t1, $t1, $t5
bne $t1, $t5, loop1
move $v0, $t3

jr $ra


sumf:

li $t5, 1
li $t3, 0
li $t1, 10
la $t0, array_a
loop2:
lw $s0, 0($t0)
add $t3, $t3, $s0
addi $t0, $t0, 4
sub $t1, $t1, $t5
bne $t1, $t5, loop2
move $v0, $t3

jr $ra
```