TASK #01:

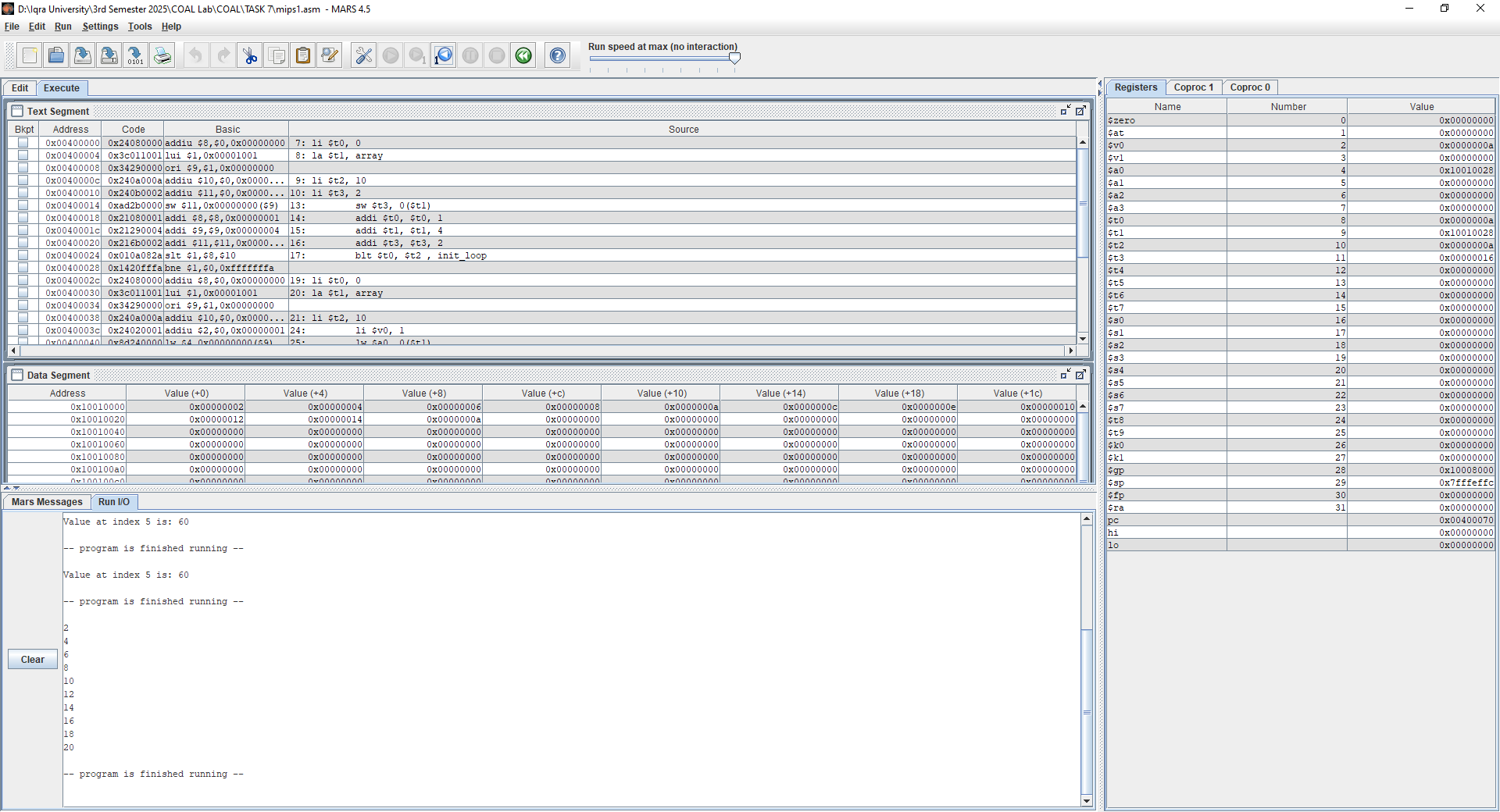
.data

array: .space 40

newline: .asciiz "\n"

.text

main:

li $t0, 0

la $t1, array

li $t2, 10

li $t3, 2

init\_loop:

sw $t3, 0($t1)

addi $t0, $t0, 1

addi $t1, $t1, 4

addi $t3, $t3, 2

blt $t0, $t2 , init\_loop

li $t0, 0

la $t1, array

li $t2, 10

print\_loop:

li $v0, 1

lw $a0, 0($t1)

syscall

li $v0, 4

la $a0, newline

syscall

addi $t1, $t1 , 4

addi $t0, $t0 , 1

blt $t0, $t2 , print\_loop

li $v0, 10

syscall

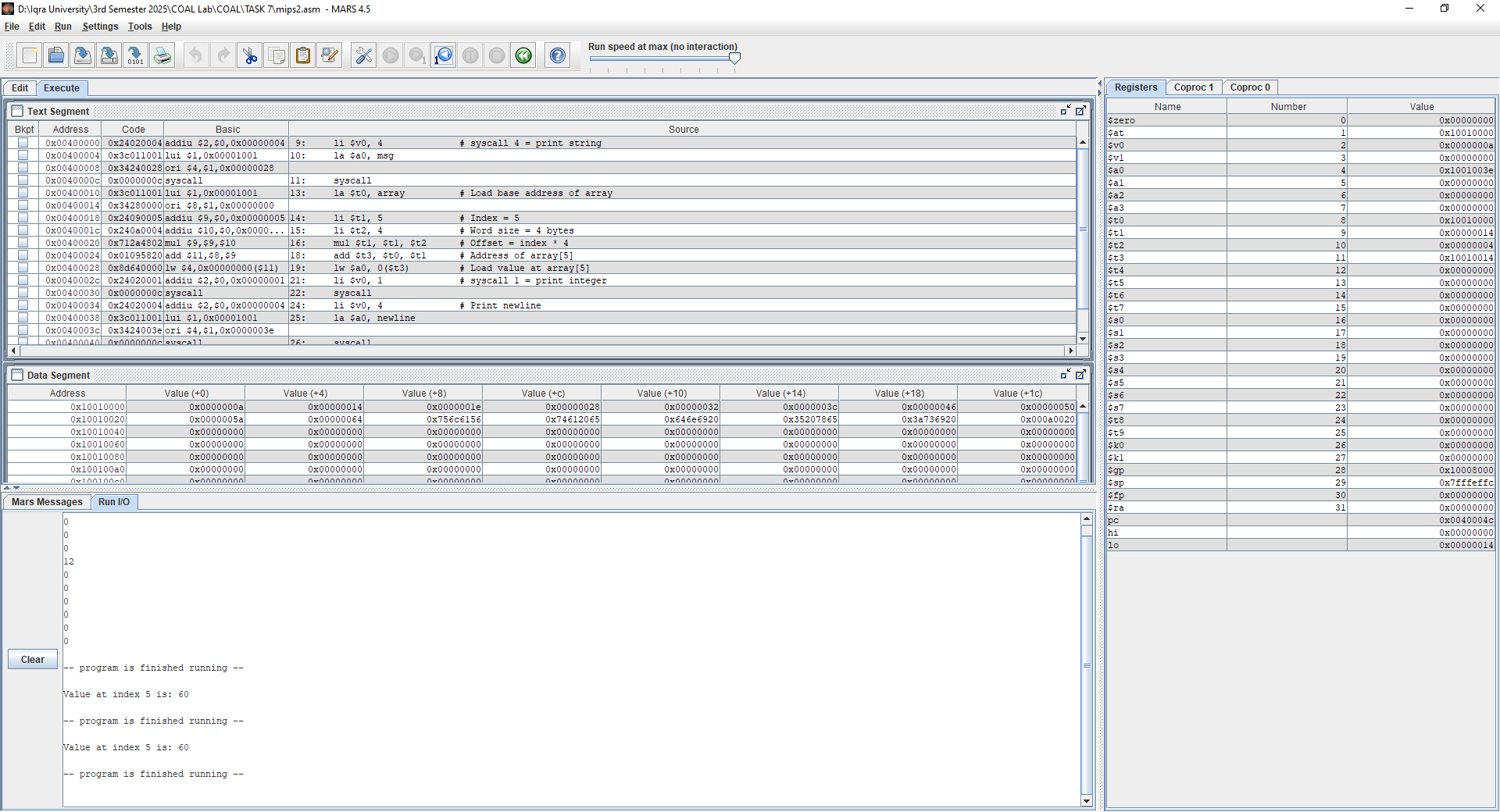
TASK #02:

.data

array: .word 10, 20, 30, 40, 50, 60, 70, 80, 90, 100 # 10-element array

msg: .asciiz "Value at index 5 is: "

newline: .asciiz "\n"

.text

.globl main

main:

li $v0, 4 # syscall 4 = print string

la $a0, msg

syscall

la $t0, array # Load base address of array

li $t1, 5 # Index = 5

li $t2, 4 # Word size = 4 bytes

mul $t1, $t1, $t2 # Offset = index \* 4

add $t3, $t0, $t1 # Address of array[5]

lw $a0, 0($t3) # Load value at array[5]

li $v0, 1 # syscall 1 = print integer

syscall

li $v0, 4 # Print newline

la $a0, newline

syscall

li $v0, 10 # Exit

syscall

TASK #03:

.data

array : .space 40 # Reserve space for 10 integers (10 \* 4 bytes)

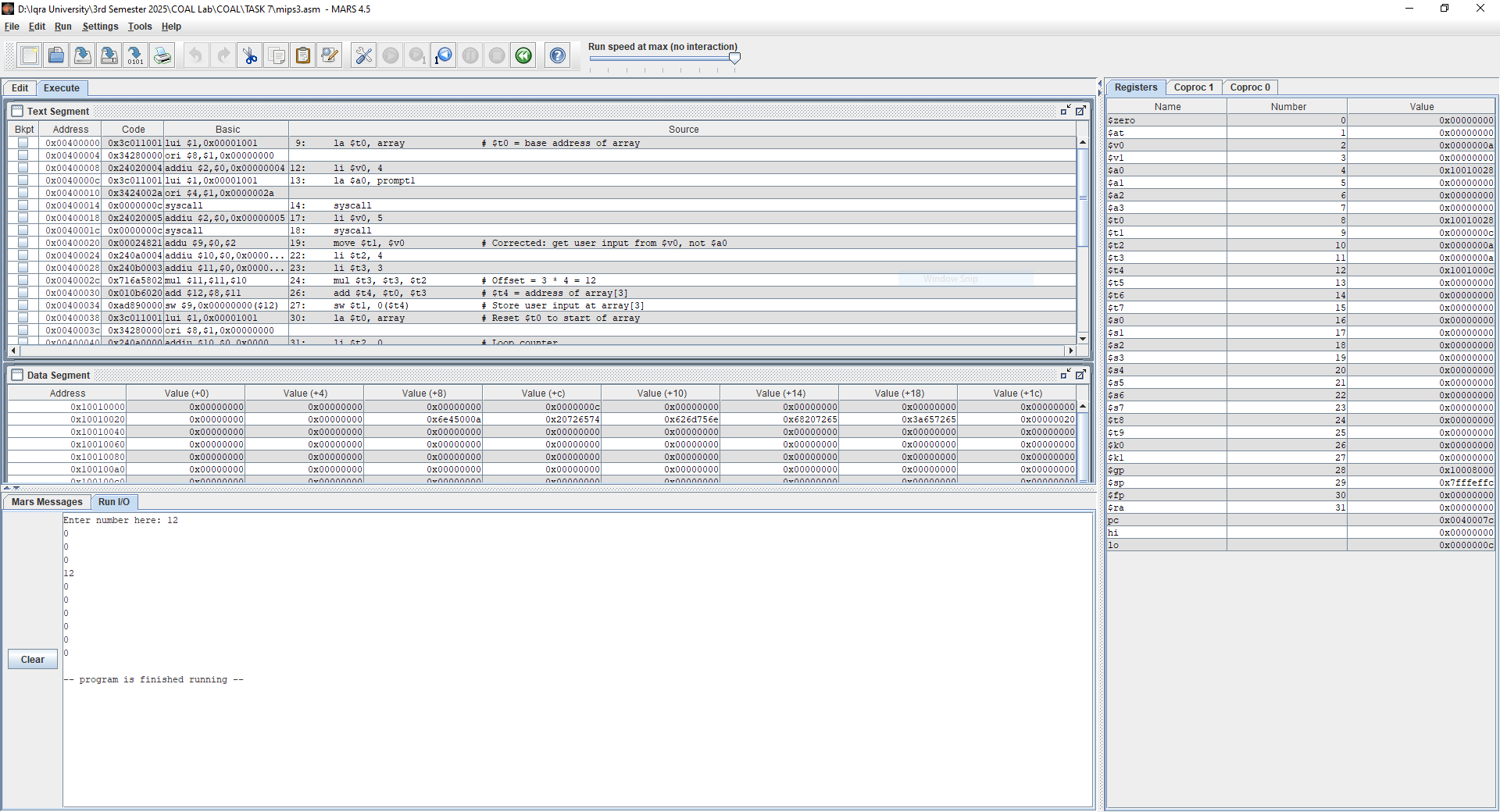
newline : .asciiz "\n" # Corrected spelling of "newlinw"

prompt1 : .asciiz "Enter number here: "

.text

.globl main

main:

 la $t0, array # $t0 = base address of array

# Prompt for input

li $v0, 4

la $a0, prompt1

syscall

# Read integer

li $v0, 5

syscall

move $t1, $v0 # Corrected: get user input from $v0, not $a0

# Store input at index 3

li $t2, 4

li $t3, 3

mul $t3, $t3, $t2 # Offset = 3 \* 4 = 12

add $t4, $t0, $t3 # $t4 = address of array[3]

sw $t1, 0($t4) # Store user input at array[3]

# Print all 10 array elements

la $t0, array # Reset $t0 to start of array

li $t2, 0 # Loop counter

li $t3, 10 # Loop limit

print\_loop:

lw $a0, 0($t0) # Load array[i]

li $v0, 1

syscall # Print integer

li $v0, 4

la $a0, newline

syscall # Print newline

addi $t2, $t2, 1

addi $t0, $t0, 4 # Move to next element

blt $t2, $t3, print\_loop # Loop if i < 10

li $v0, 10

syscall # Exit