TASK #01:

.data

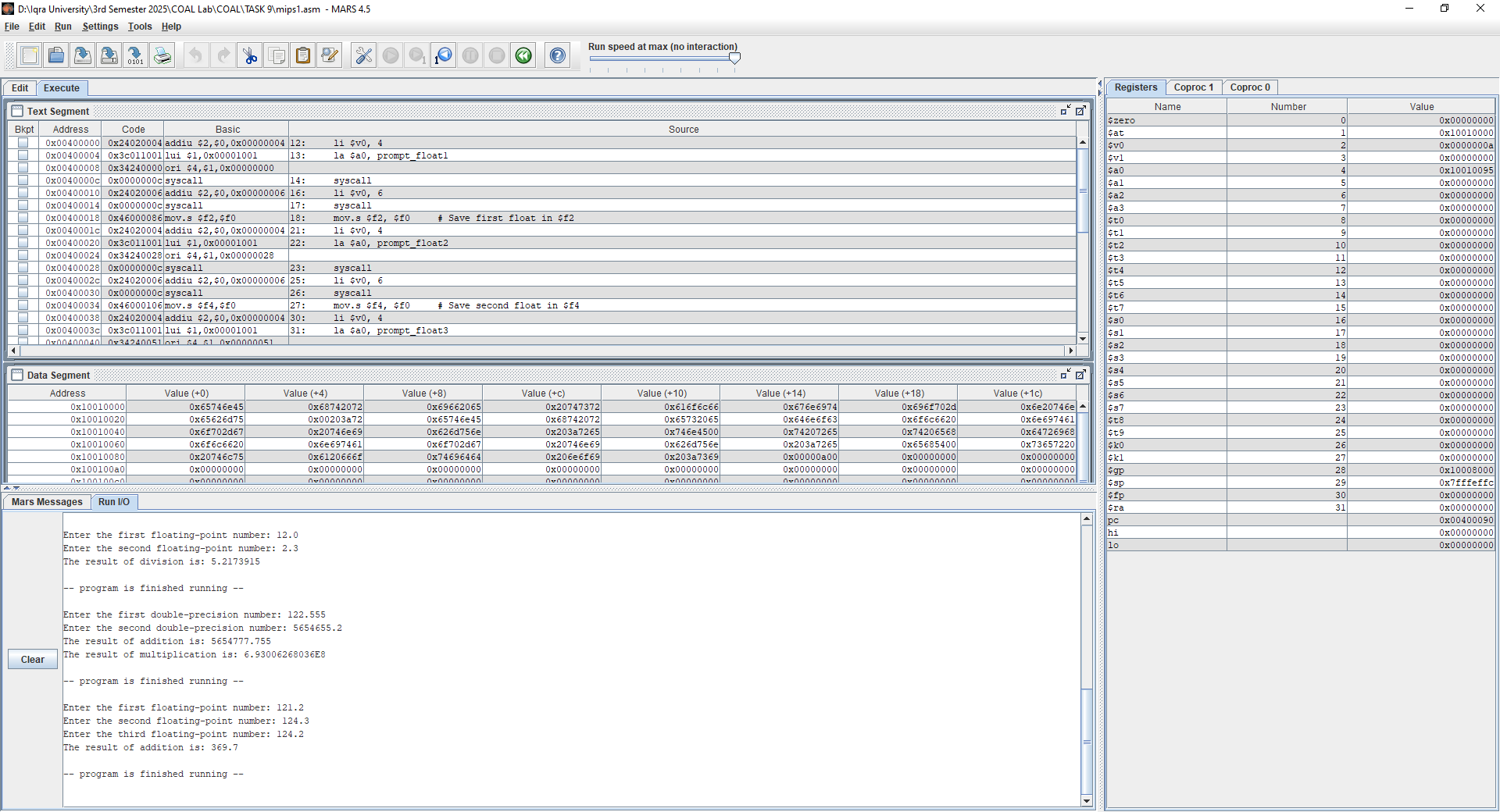
prompt\_float1: .asciiz "Enter the first floating-point number: "

prompt\_float2: .asciiz "Enter the second floating-point number: "

prompt\_float3: .asciiz "Enter the third floating-point number: "

result\_add: .asciiz "The result of addition is: "

newline: .asciiz "\n"



.text

.globl main

main:

# Prompt and read first float

li $v0, 4

la $a0, prompt\_float1

syscall

li $v0, 6

syscall

mov.s $f2, $f0 # Save first float in $f2

# Prompt and read second float

li $v0, 4

la $a0, prompt\_float2

syscall

li $v0, 6

syscall

mov.s $f4, $f0 # Save second float in $f4

# Prompt and read third float

li $v0, 4

la $a0, prompt\_float3

syscall

li $v0, 6

syscall

mov.s $f6, $f0 # Save third float in $f6

# Add three floats: $f2 + $f4 + $f6

add.s $f8, $f2, $f4

add.s $f8, $f8, $f6 # f8 now holds the total

# Print result label

li $v0, 4

la $a0, result\_add

syscall

# Print the result

mov.s $f12, $f8

li $v0, 2

syscall

# Newline

li $v0, 4

la $a0, newline

syscall

# Exit

li $v0, 10

syscall

TASK #02:

.data

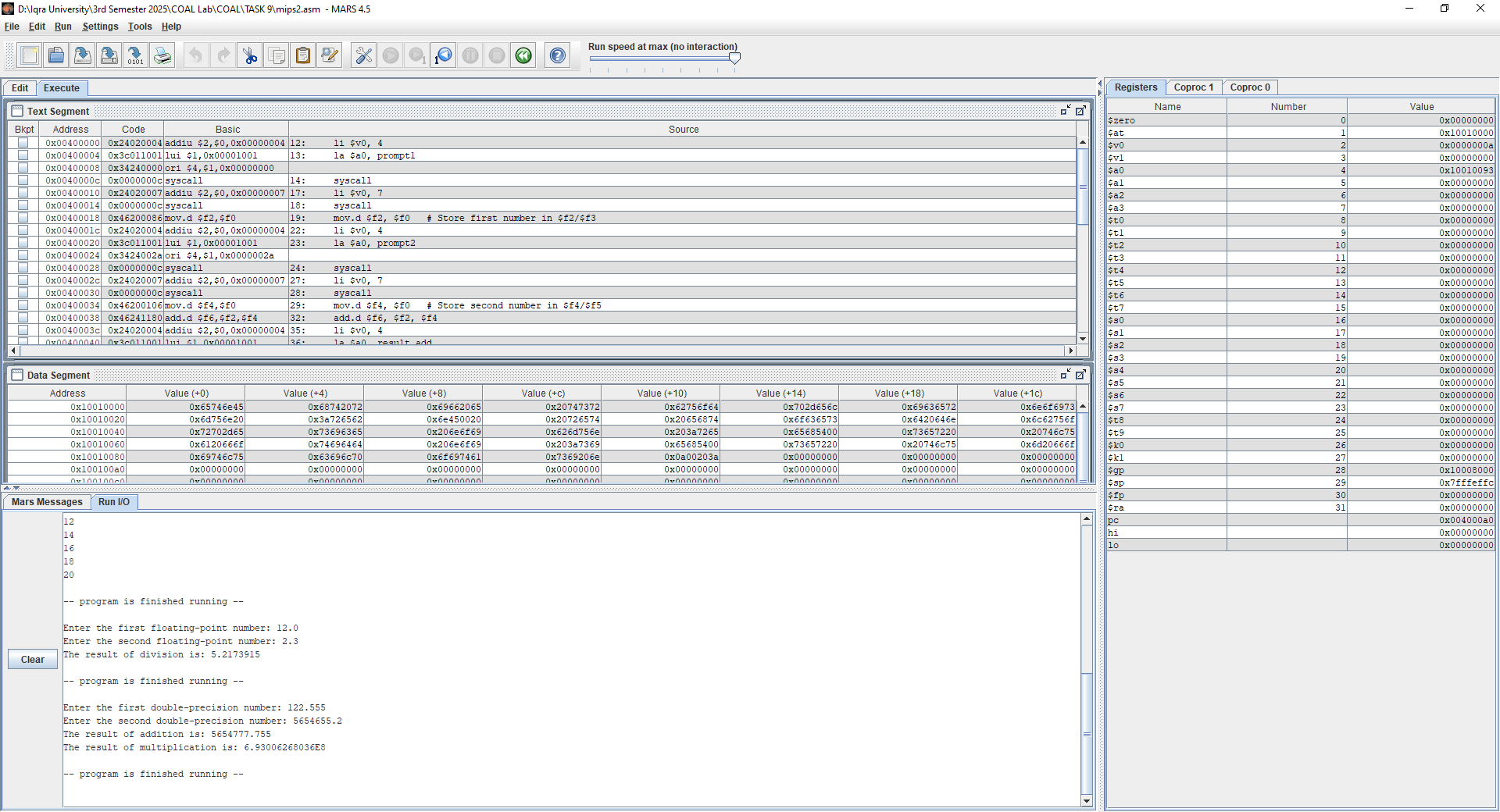
prompt1: .asciiz "Enter the first double-precision number: "

prompt2: .asciiz "Enter the second double-precision number: "

result\_add: .asciiz "The result of addition is: "

result\_mult: .asciiz "The result of multiplication is: "

newline: .asciiz "\n"



.text

.globl main

main:

# Prompt for first number

li $v0, 4

la $a0, prompt1

syscall

# Read first double (64-bit float)

li $v0, 7

syscall

mov.d $f2, $f0 # Store first number in $f2/$f3

# Prompt for second number

li $v0, 4

la $a0, prompt2

syscall

# Read second double

li $v0, 7

syscall

mov.d $f4, $f0 # Store second number in $f4/$f5

# Add the two double numbers

add.d $f6, $f2, $f4

# Print addition result message

li $v0, 4

la $a0, result\_add

syscall

# Print result of addition

mov.d $f12, $f6

li $v0, 3 # 3 = print double

syscall

# Newline

li $v0, 4

la $a0, newline

syscall

# Multiply the two double numbers

mul.d $f8, $f2, $f4

# Print multiplication result message

li $v0, 4

la $a0, result\_mult

syscall

# Print result of multiplication

mov.d $f12, $f8

li $v0, 3 # 3 = print double

syscall

# Newline

li $v0, 4

la $a0, newline

syscall

# Exit program

li $v0, 10

syscall

TASK #03:

.data

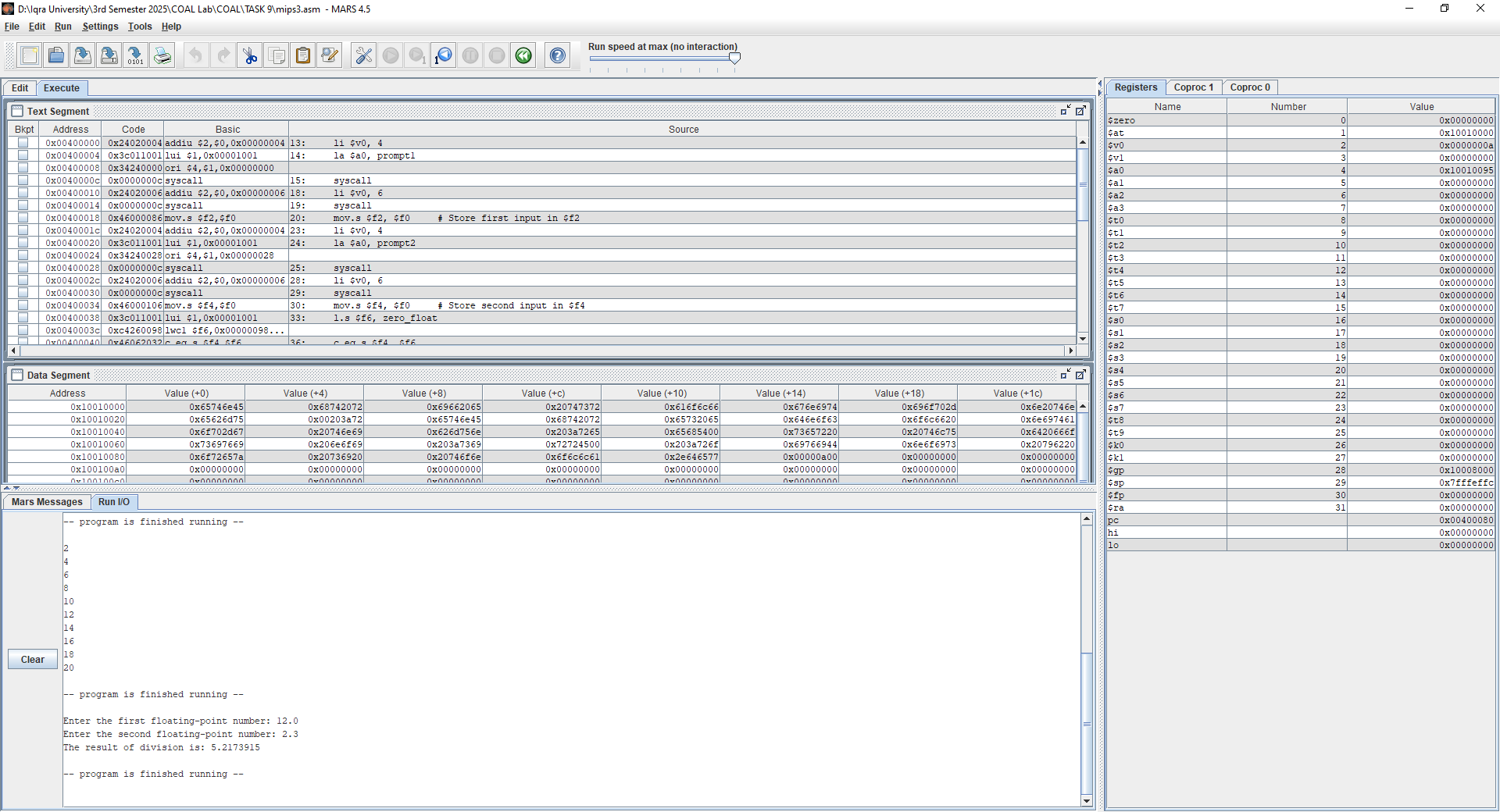
prompt1: .asciiz "Enter the first floating-point number: "

prompt2: .asciiz "Enter the second floating-point number: "

result\_text: .asciiz "The result of division is: "

error\_text: .asciiz "Error: Division by zero is not allowed."

newline: .asciiz "\n"

zero\_float: .float 0.0

.text

.globl main

main:

# Prompt for first number

li $v0, 4

la $a0, prompt1

syscall

# Read first float

li $v0, 6

syscall

mov.s $f2, $f0 # Store first input in $f2

# Prompt for second number

li $v0, 4

la $a0, prompt2

syscall

# Read second float

li $v0, 6

syscall

mov.s $f4, $f0 # Store second input in $f4

# Load 0.0 into $f6 for comparison

l.s $f6, zero\_float

# Compare: if $f4 == 0.0, branch

c.eq.s $f4, $f6

bc1t division\_by\_zero

# Perform division: $f2 / $f4

div.s $f8, $f2, $f4

# Print result message

li $v0, 4

la $a0, result\_text

syscall

# Print result value

mov.s $f12, $f8

li $v0, 2

syscall

# Newline

li $v0, 4

la $a0, newline

syscall

# Exit program

li $v0, 10

syscall

division\_by\_zero:

# Print error message

li $v0, 4

la $a0, error\_text

syscall

# Newline

li $v0, 4

la $a0, newline

syscall

# Exit program

li $v0, 10

syscall