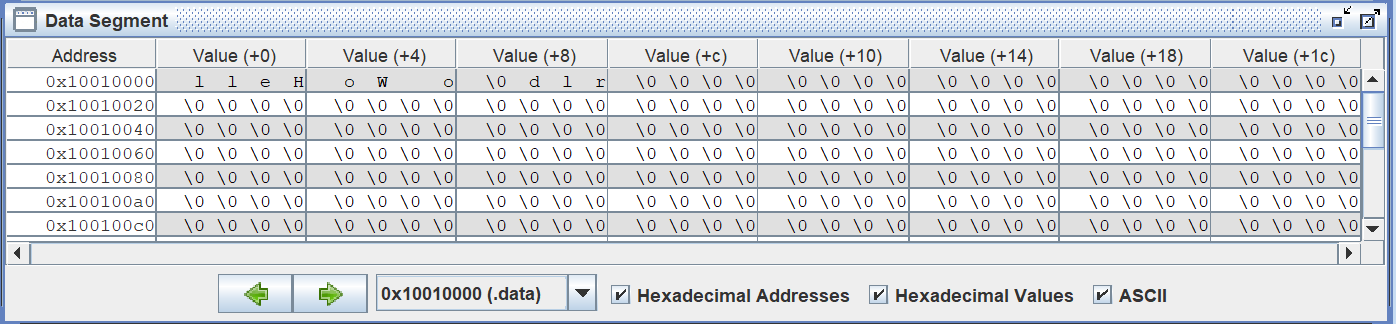
Báo cáo thực hành KTMT tuần 5

Lê Quốc Đảng

20225801

**Assignment 1**

****

Xâu được chia thành các nhóm 4 byte, và thứ tự lưu lần lượt từ phải qua trái

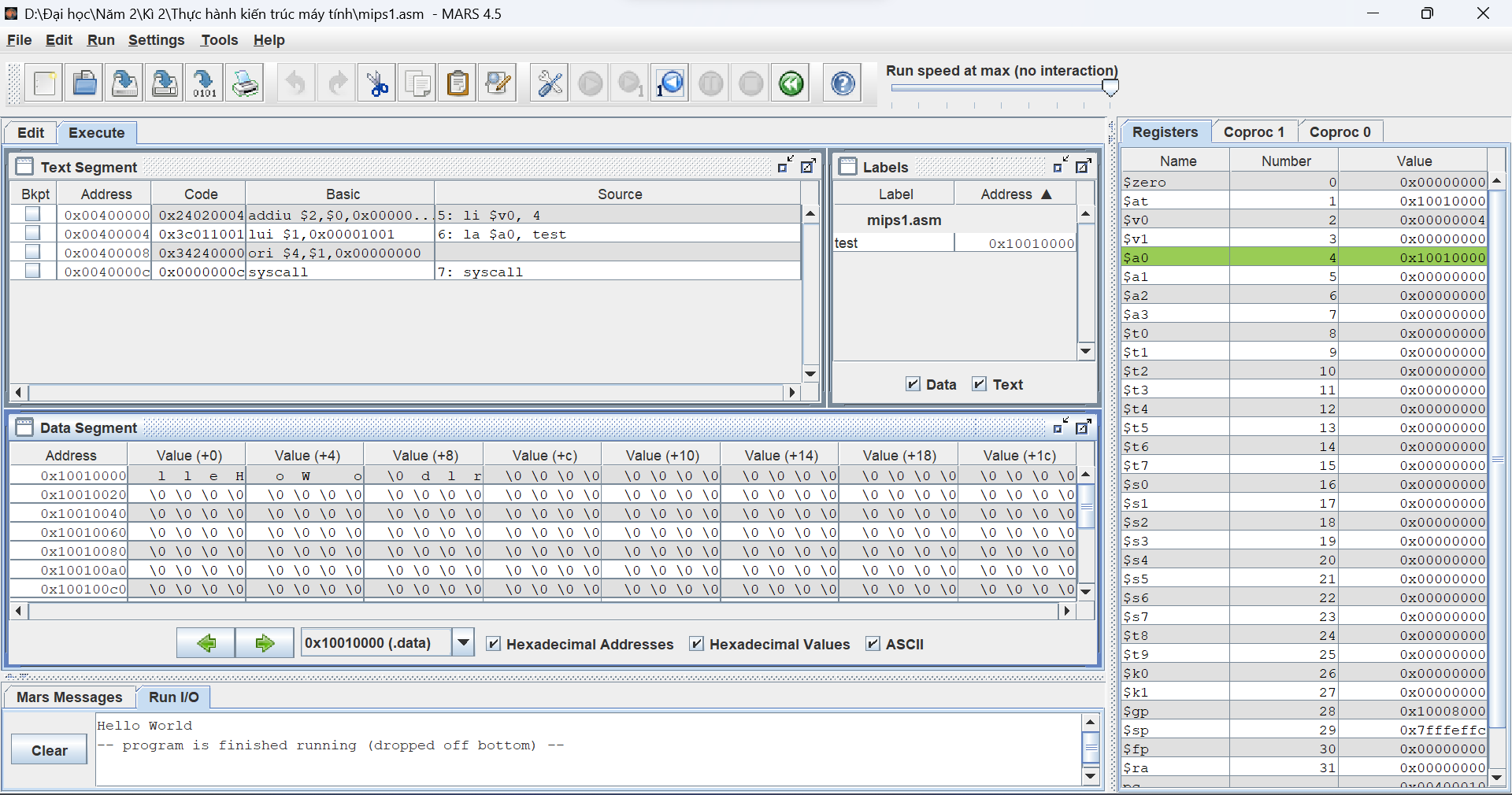
* Chạy chương trình

-Thanh ghi $v0: 0x00000000 -> 0x00000004

-Thanh ghi $at: 0x00000000 -> 0x10010000

-Thanh ghi $a0: 0x00000000 -> 0x10010000

-Giải thích: khi gọi lệnh syscall, HĐH sẽ xem giá trị của $v0. Vì giá trị thanh ghi $v0 = 4 => thanh ghi $a0 sẽ là 1 string và in ra màn hình.

-Sau đó in ra “Hello world”

**Assignment 2**

* Code:

.data

string1: .asciiz "The sum of "

string2: .asciiz " and "

string3: .asciiz " is "

.text

li $s0,4

li $s1,5

add $t1,$s0,$s1

#Print string1

la $a0,string1

li $v0,4

syscall

#Print $s0

move $a0,$s0

li $v0,1

syscall

#Print string2

la $a0,string2

li $v0,4

syscall

#Print $s1

move $a0,$s1

li $v0,1

syscall

#Print string3

la $a0,string3

li $v0,4

syscall

#Print $t1

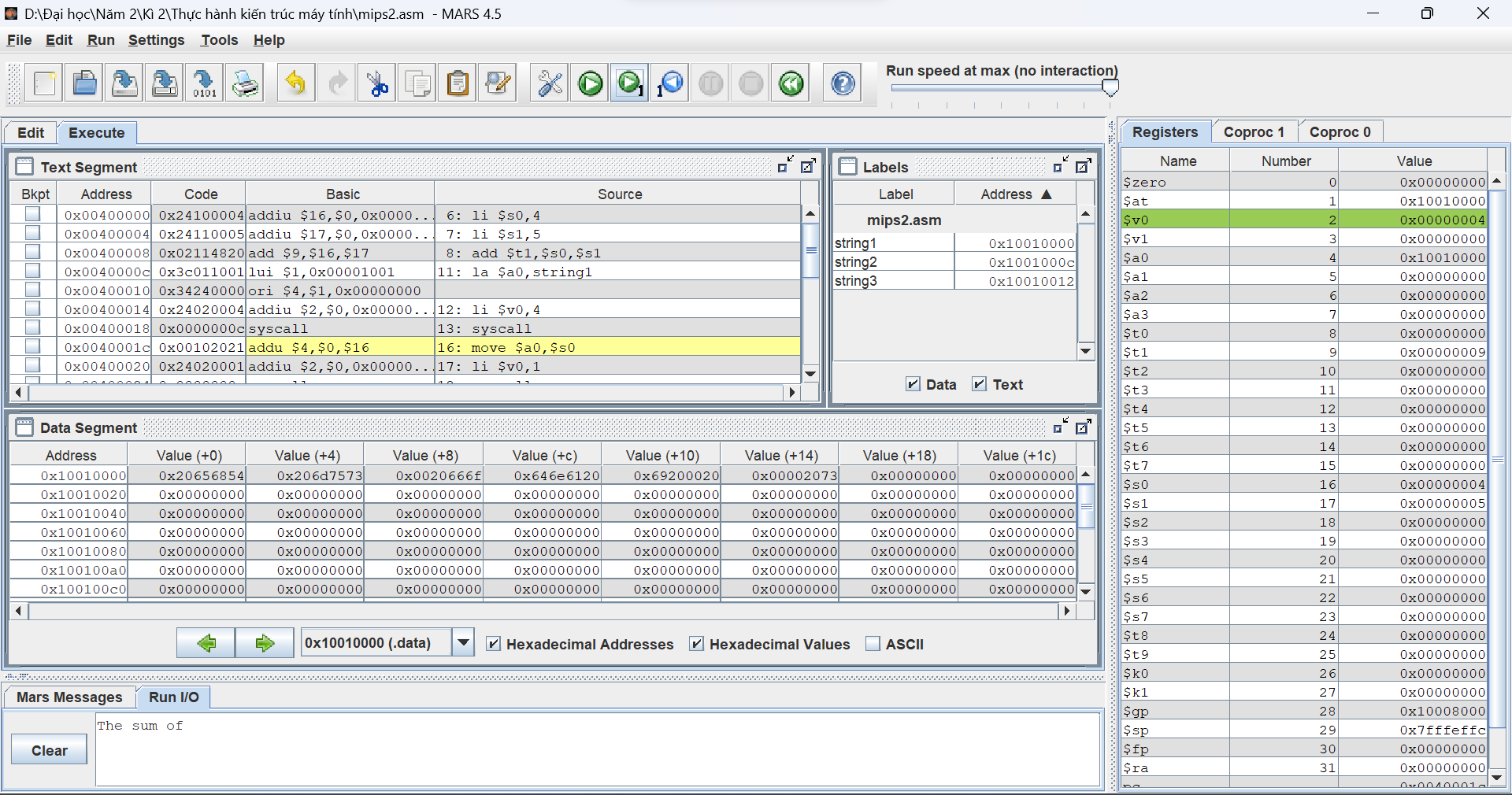
move $a0,$t1

li $v0,1

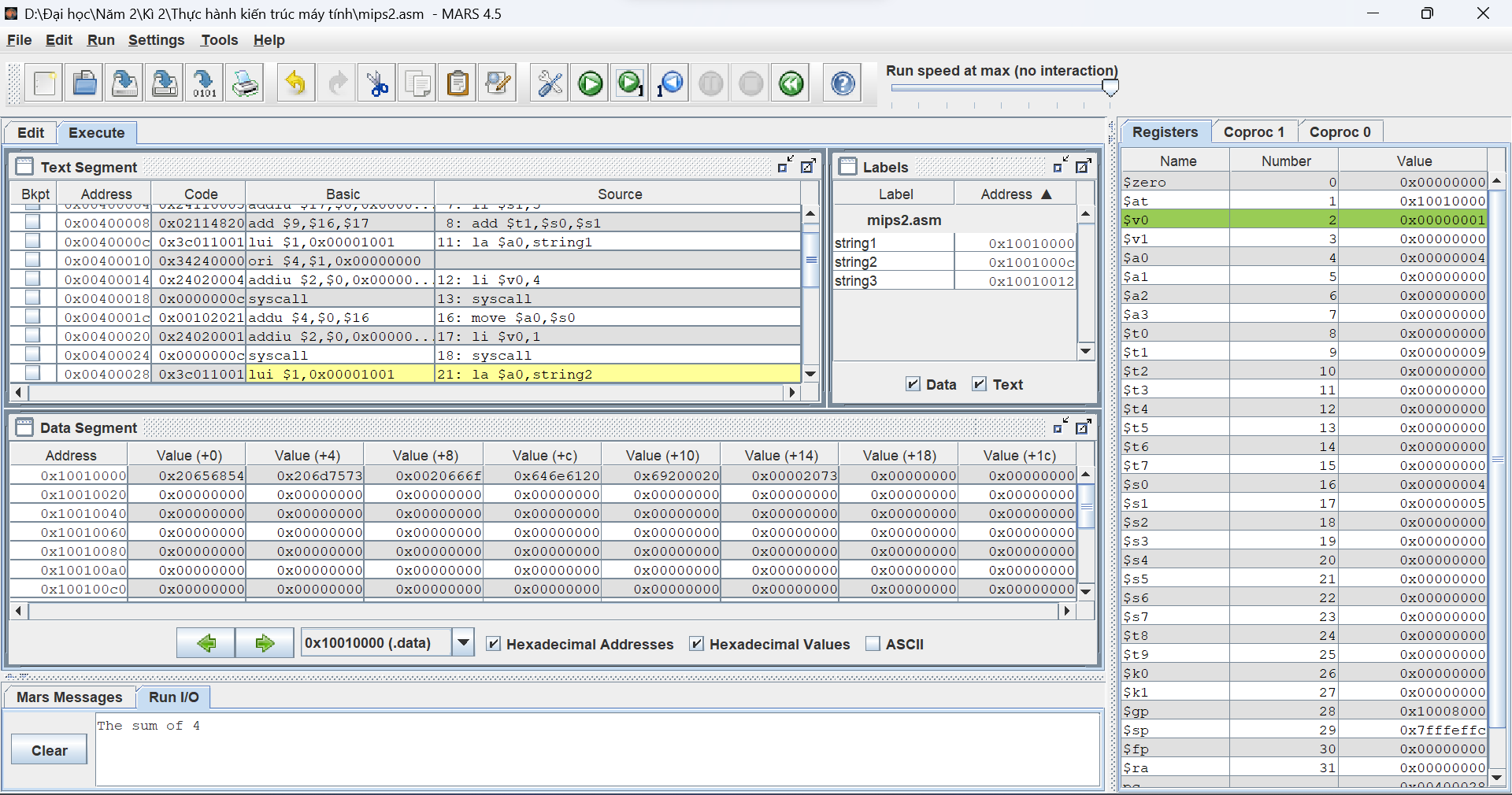
syscall

* Thực thi chương trình

-In ra màn hình “The sum of”. $v0 = 4 => thanh ghi $a0 sẽ là 1 string và in ra màn hình

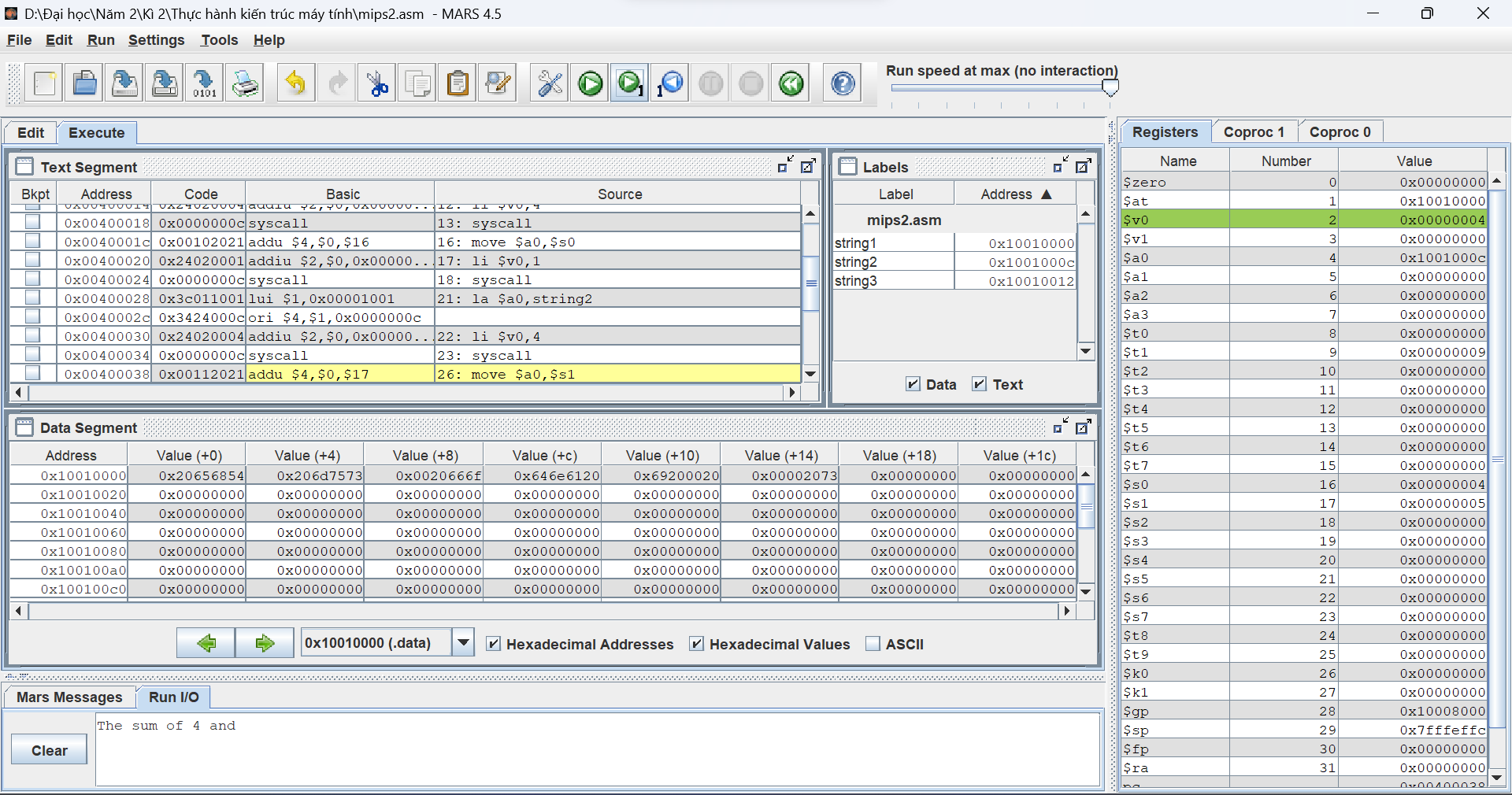


-In ra màn hình $s0. $v0 = 1 => thanh ghi $a0 sẽ là 1 số và in ra màn hình.



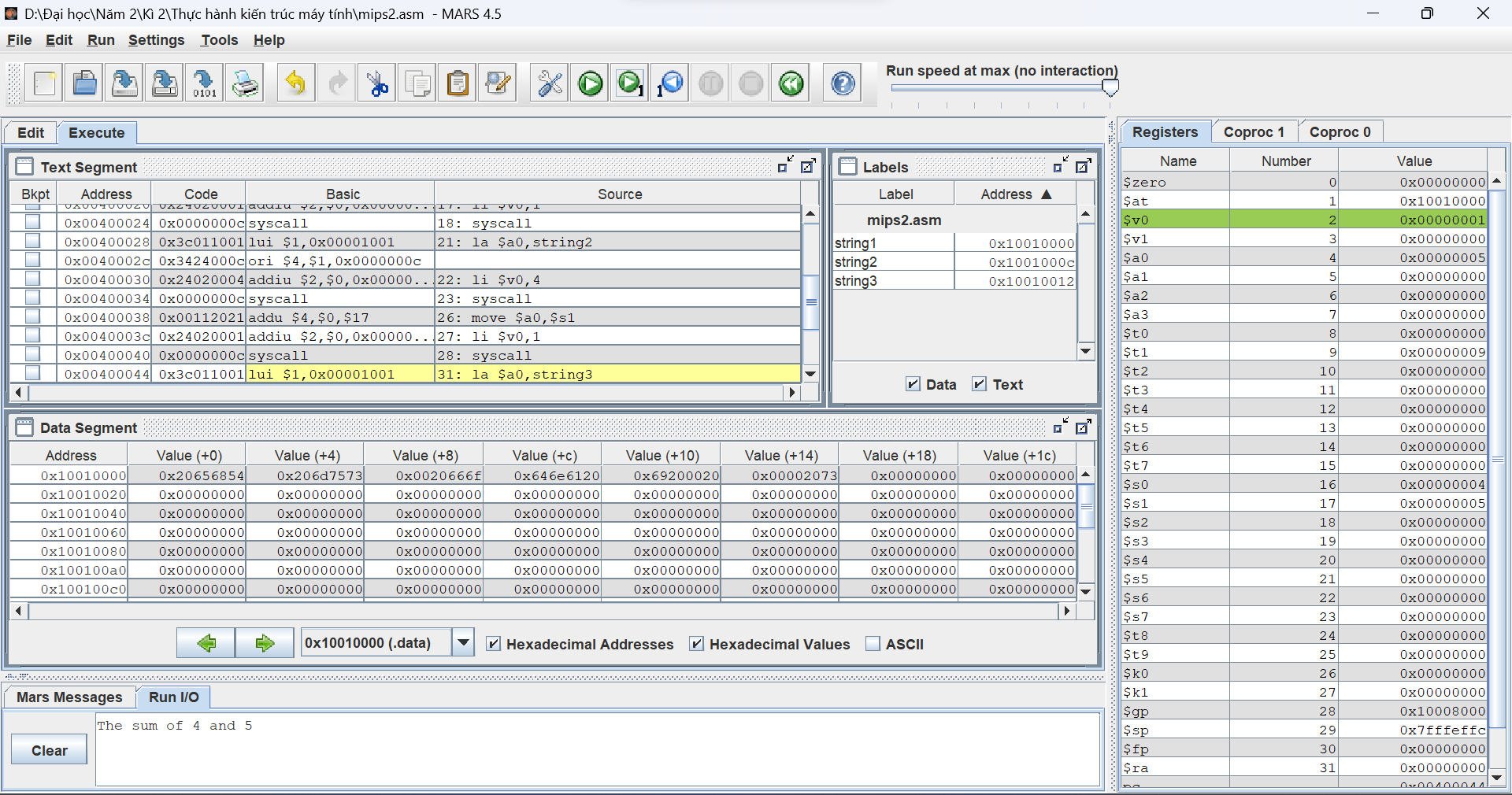
-In ra màn hình “and”

$v0 = 4 => thanh ghi $a0 sẽ là 1 string và in ra màn hình.



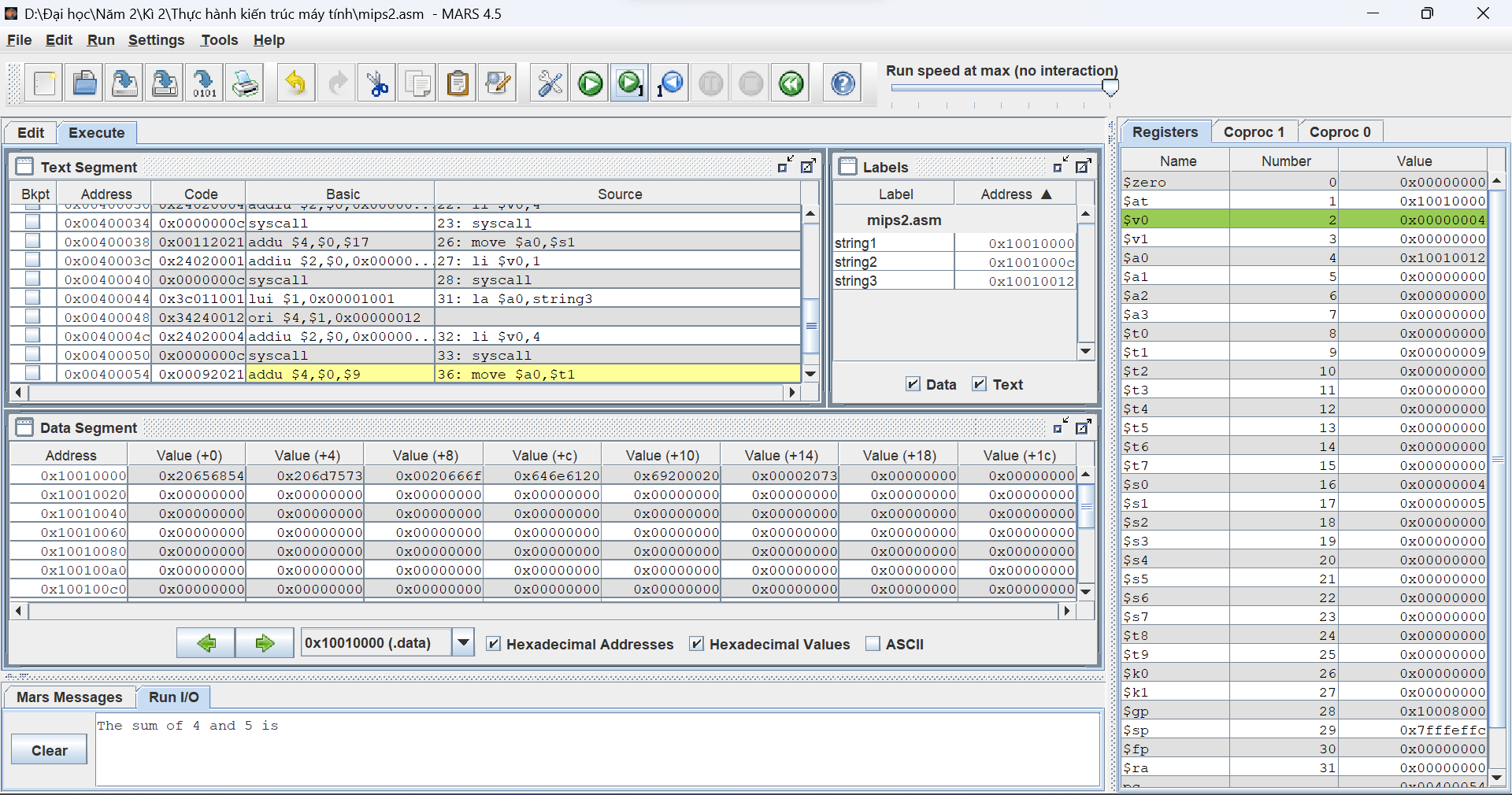
-In ra màn hình $s1

$v0 = 1 => thanh ghi $a0 sẽ là 1 số và in ra màn hình.



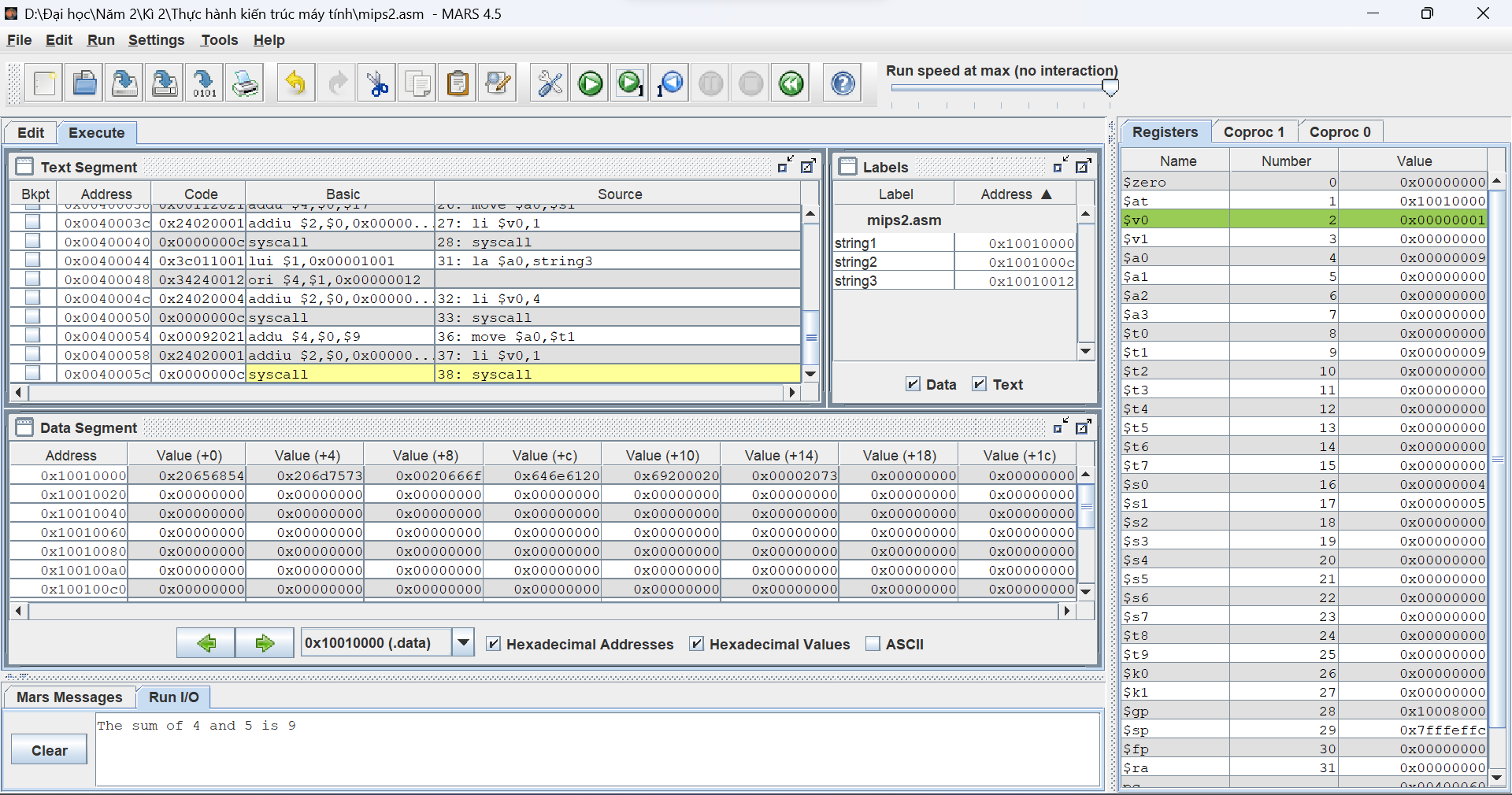
-In ra màn hình “is”

$v0 = 4 => thanh ghi $a0 sẽ là 1 string và in ra màn hình.



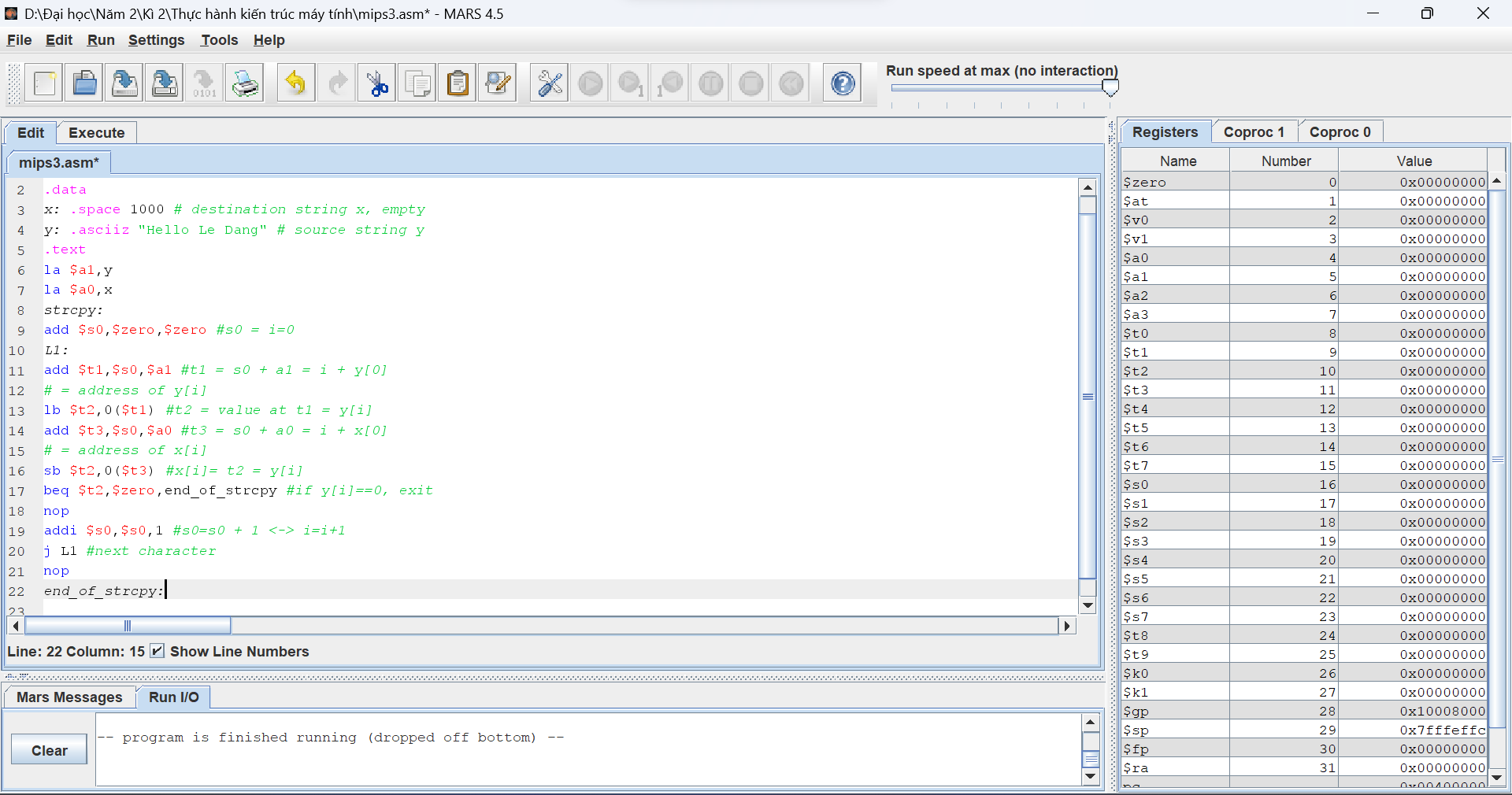
-In ra màn hình kết quả của phép cộng $s0+$s1 = 4+5 = 9

$v0 = 1 => thanh ghi $a0 sẽ là 1 số và in ra màn hình.



* Kết quả đúng

**Assignment 3**

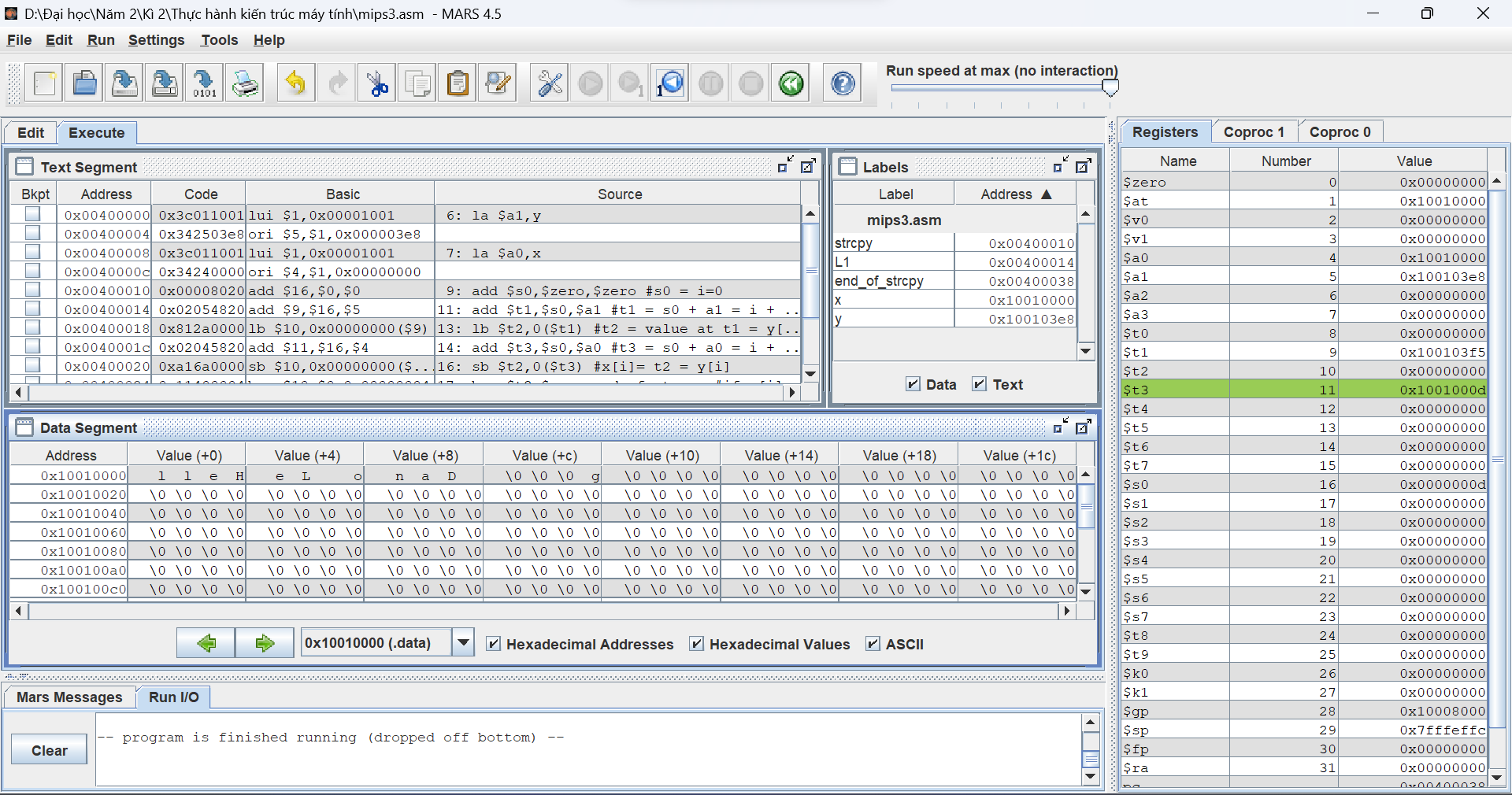
****

Giải thích

-Nạp địa chỉ của X, Y

-Thanh ghi $t1 trỏ vào Y[i], $t3 trỏ vào X[i]

-Vòng lặp thực hiện load từng byte của Y[i] và rồi store từng byte vào X[i]

****

* Kết quả đúng

**Assignment 4**

* Code

.data

string: .space 50

Mess1: .asciiz "Nhap xau: "

Mess2: .asciiz "Do dai xau la: "

.text

main:

get\_string:

li $v0, 54

la $a0, Mess1

la $a1, string

la $a2, 100

syscall

length:

la $a0,string # $a0 = address(string[0])

add $t0,$zero,$zero # $t0 = i = 0

check\_char:

add $t1,$a0,$t0 # $t1 = $a0 + $t0

# = address(string[i])

lb $t2, 0($t1) # $t2 = string[i]

beq $t2, $zero, end\_of\_str # is null char?

addi $t0, $t0, 1 # $t0 = $t0 + 1 -> i = i + 1

j check\_char

end\_of\_str:

end\_of\_length:

print\_length\_of\_string:

addi $t0, $t0, -1

li $v0, 56

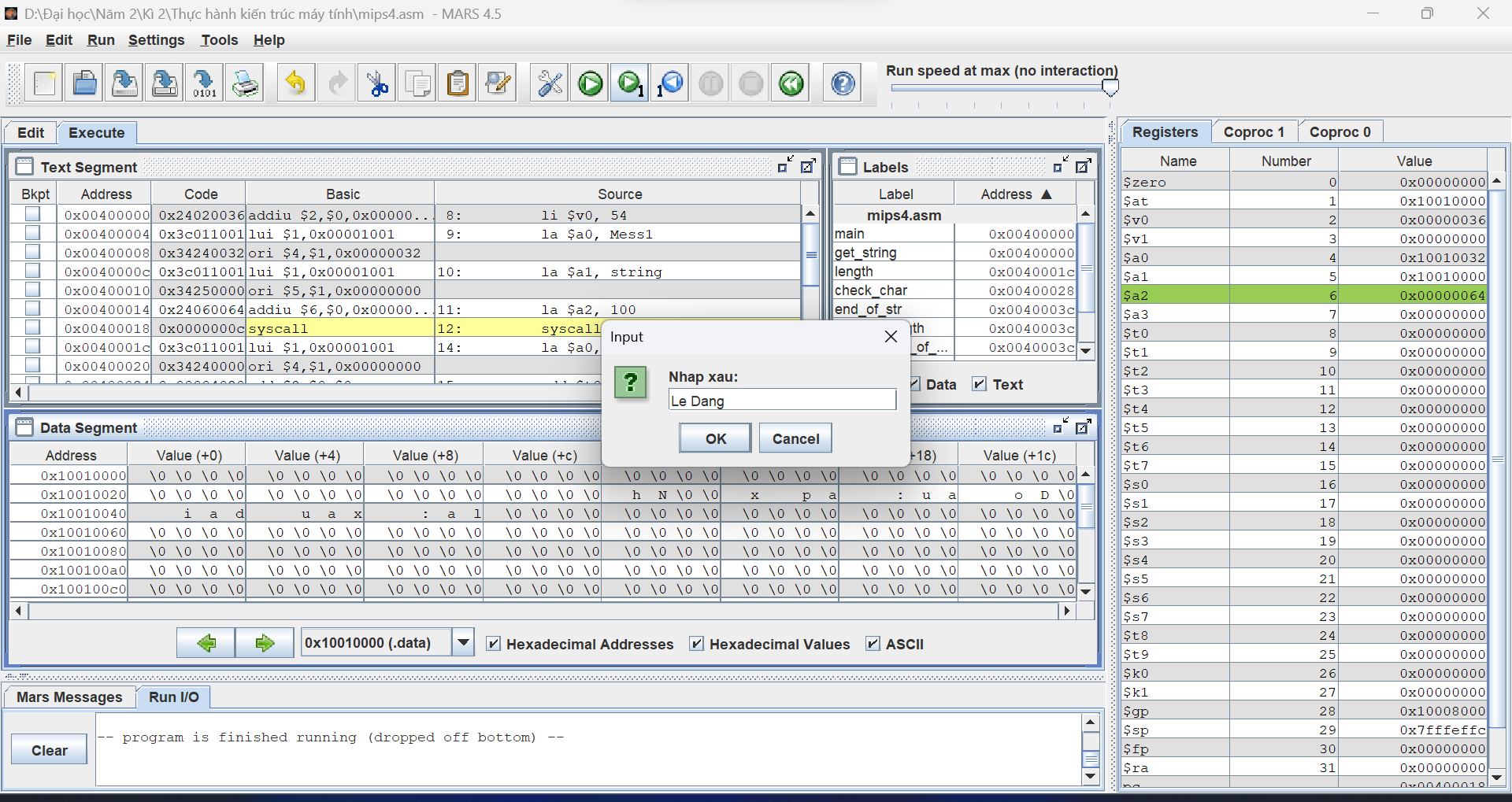
la $a0, Mess2

move $a1, $t0

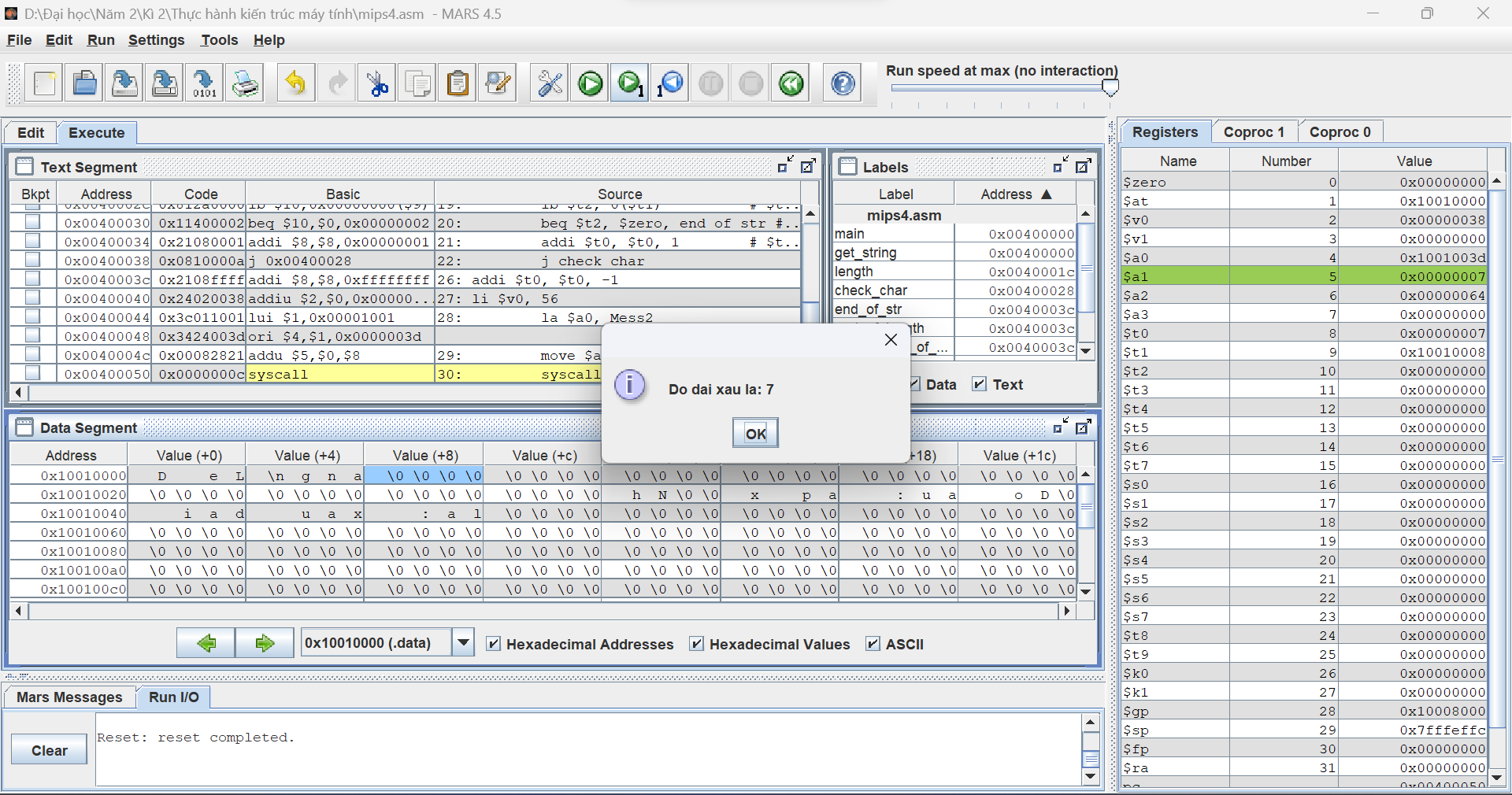
syscall

* Chạy chương trình

-Nhập xâu. $v0 = 54 => nhập xâu vào hộp thoại và $a1 là 1 string



-In ra độ dài xâu. $v0 = 56 => hiển thị hộp thoại xâu



* Kết quả đúng

**Assignment 5**

* Code

# Dao nguoc str va luu vao reverse\_str

.data

str: .space 50

reverse \_str: .space 50

mess: .asciiz "Chuoi dao nguoc la: "

.text

la $a3, reverse\_str #nap dia chi cua reverse\_str

read\_str:

li $v0,8

la $a0,str

li $a1,21 #20 ky tu

syscall

reverse:

#Dem do dai xau str

xor $t3,$0,$0 #t3 = 0

li $t0,0 #t0 = i =0

length: add $t1,$a0,$t0 #t1 = diachi str[i]

lb $t2,0($t1) #load byte t1 vao t2

beq $t2,$0,end\_length #t2 = null -> ket thuc

addi $t0,$t0,1 #i++

addi $t3,$t3,1 #t3++

j length

end\_length:

subi $t3,$t3,1

andi $t0,$t0,0 #reset lai bien i

change: add $t1,$a0,$t3 #t1 = diachi str[i]

lb $t2,0($t1) #load byte t1 vao t2

add $t4,$a3,$t0 # t4 = dia chi reverse\_str[i]

sb $t2,0($t4) #luu t2 vao t4

beq $t2,$0,end\_change #t2=null -> end

addi $t0,$t0,1 #i++

subi $t3,$t3,1 #t3--

j change

end\_change:

printf: # In xau reverse\_str

li $v0,4

la $a0,mess

syscall

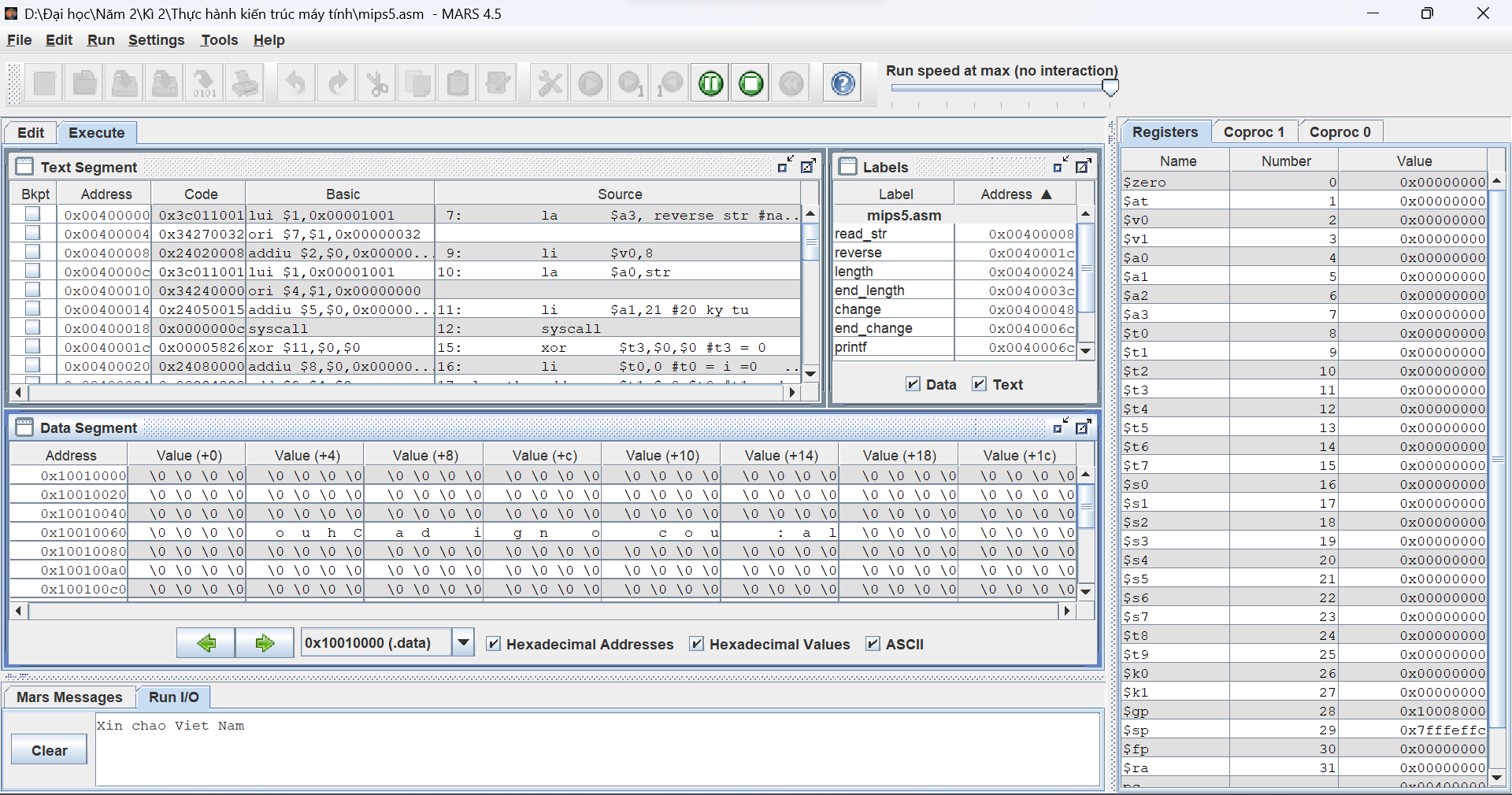
li $v0,4

la $a0, reverse\_str

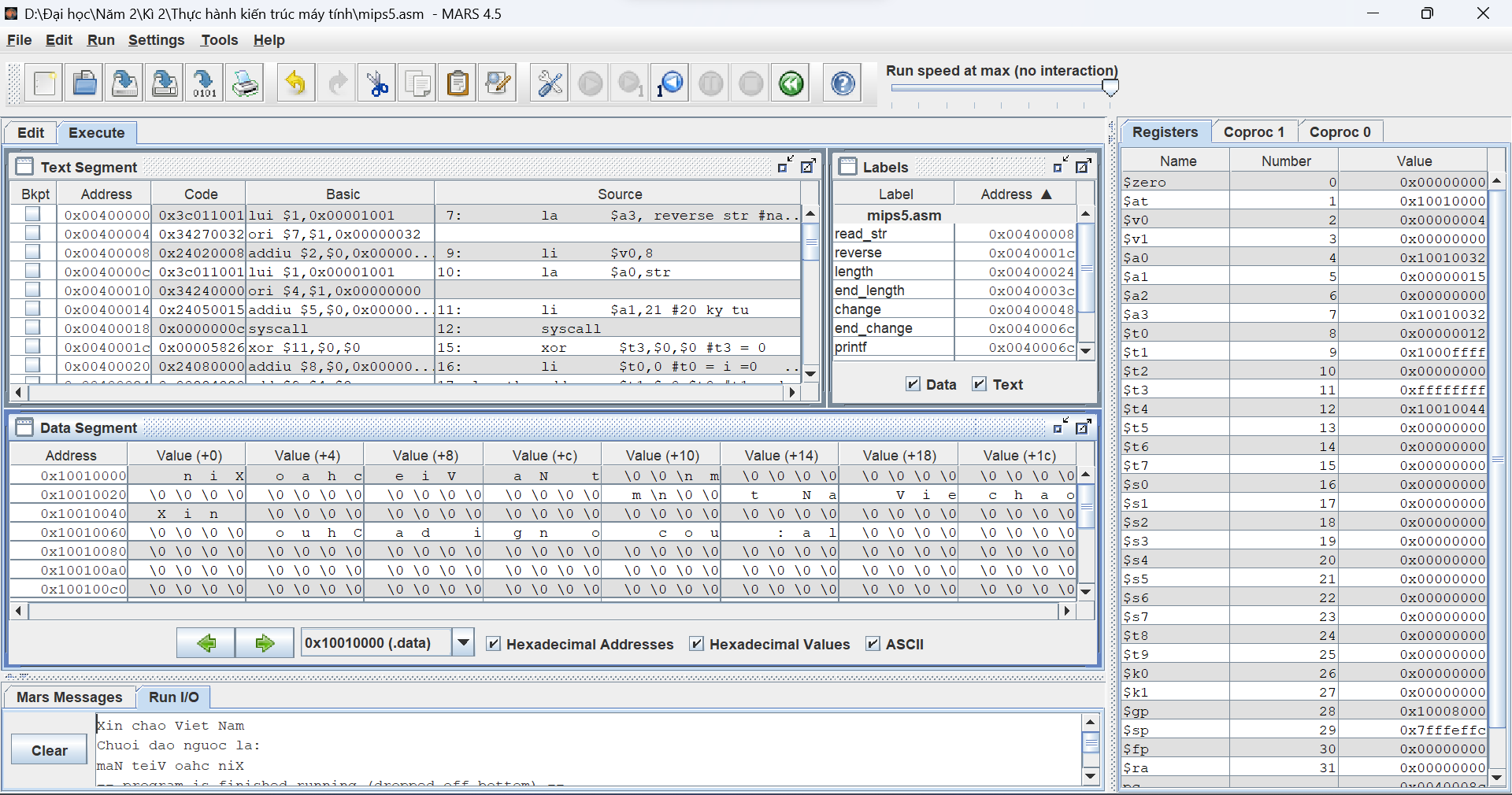
syscall

* Chạy chương trình

-Nhập string



-In ra string bị đảo ngược. $v0 = 4 => in ra màn hình



* Kết quả đúng