Họ và tên : Lê Ngọc Anh Quân

MSSV:20176852

Bài 1

Soure code :

#Laboratory Exercise 7 Home Assignment 1

.text

main:

li $a0,-30 #load input parameter

jal abs #jump and link to abs procedure

nop

add $s0, $zero, $v0

li $v0,10 #terminate

syscall

endmain:

#--------------------------------------------------------------------

# function abs

# param[in] $a0 the interger need to be gained the absolute value

# return $v0 absolute value

#--------------------------------------------------------------------

abs:

sub $v0,$zero,$a0 #put -(a0) in v0; in case (a0)<0

bltz $a0,done #if (a0)<0 then done

nop

add $v0,$a0,$zero #else put (a0) in v0

done:

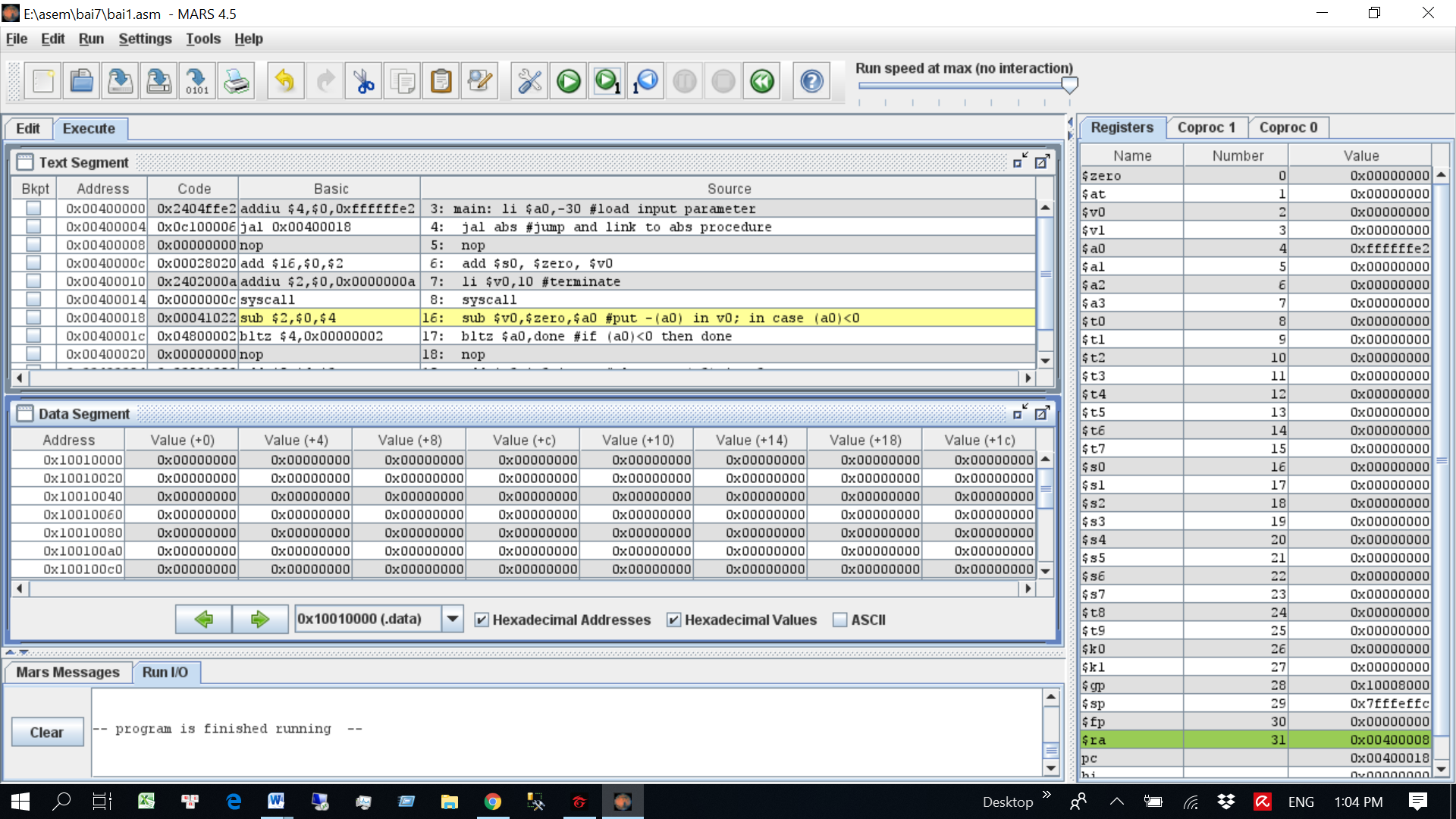
jr $ra

Nhận xét

Câu lệnh : jal abs #jump and link to abs procedure

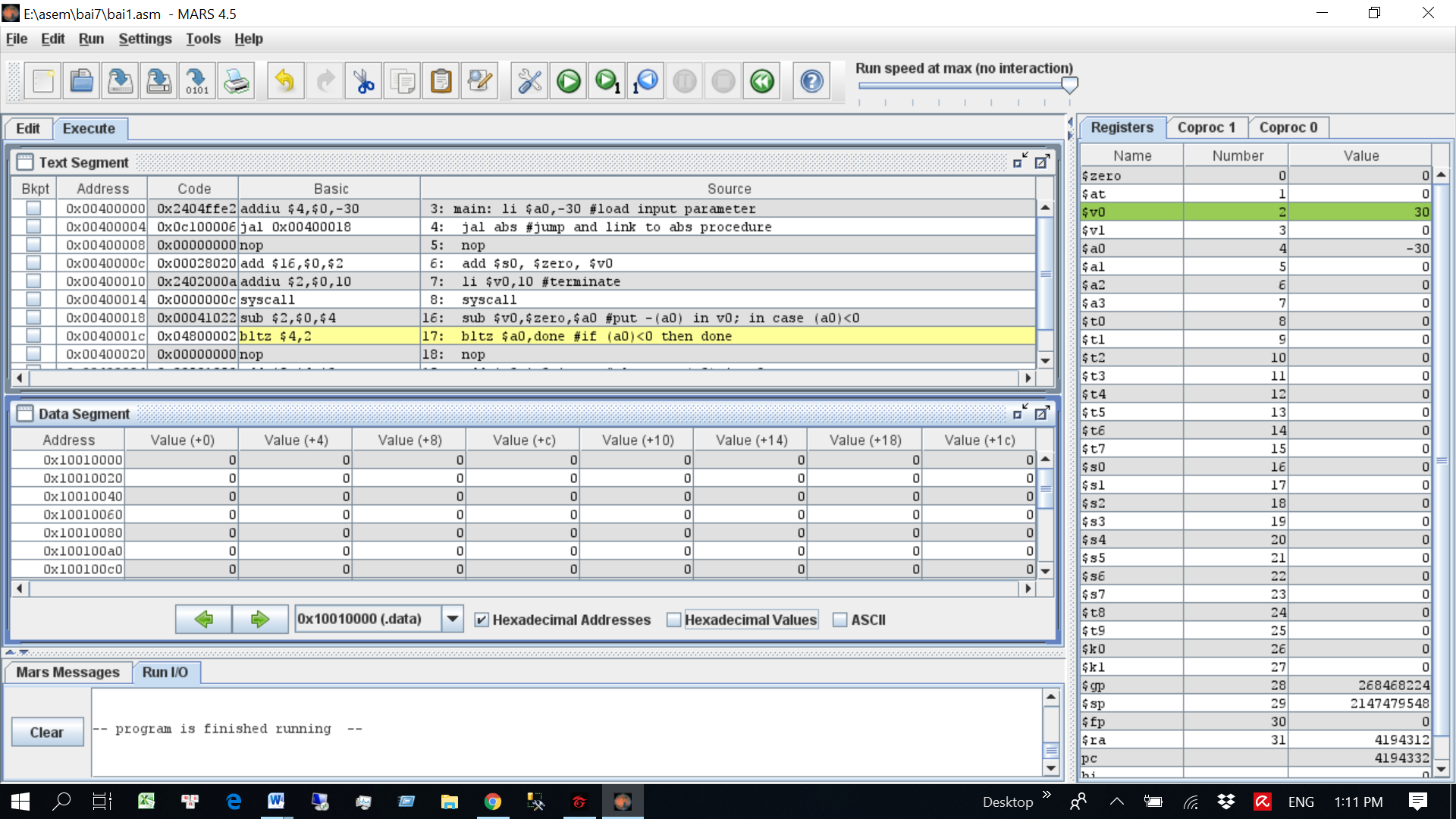
* Câu lệnh này sẽ nhảy đến hàm abs , trước đó sẽ lưu địa chỉ của câu lệnh dòng tiếp theo vào thanh ghi $ra .

Kết quả :thanh ghi $ra=0x00400008 = địa chỉ cảu câu lệnh nop là câu lệnh dòng tiếp theo câu lệnh jal



Câu lệnh tiếp theo

sub $v0,$zero,$a0

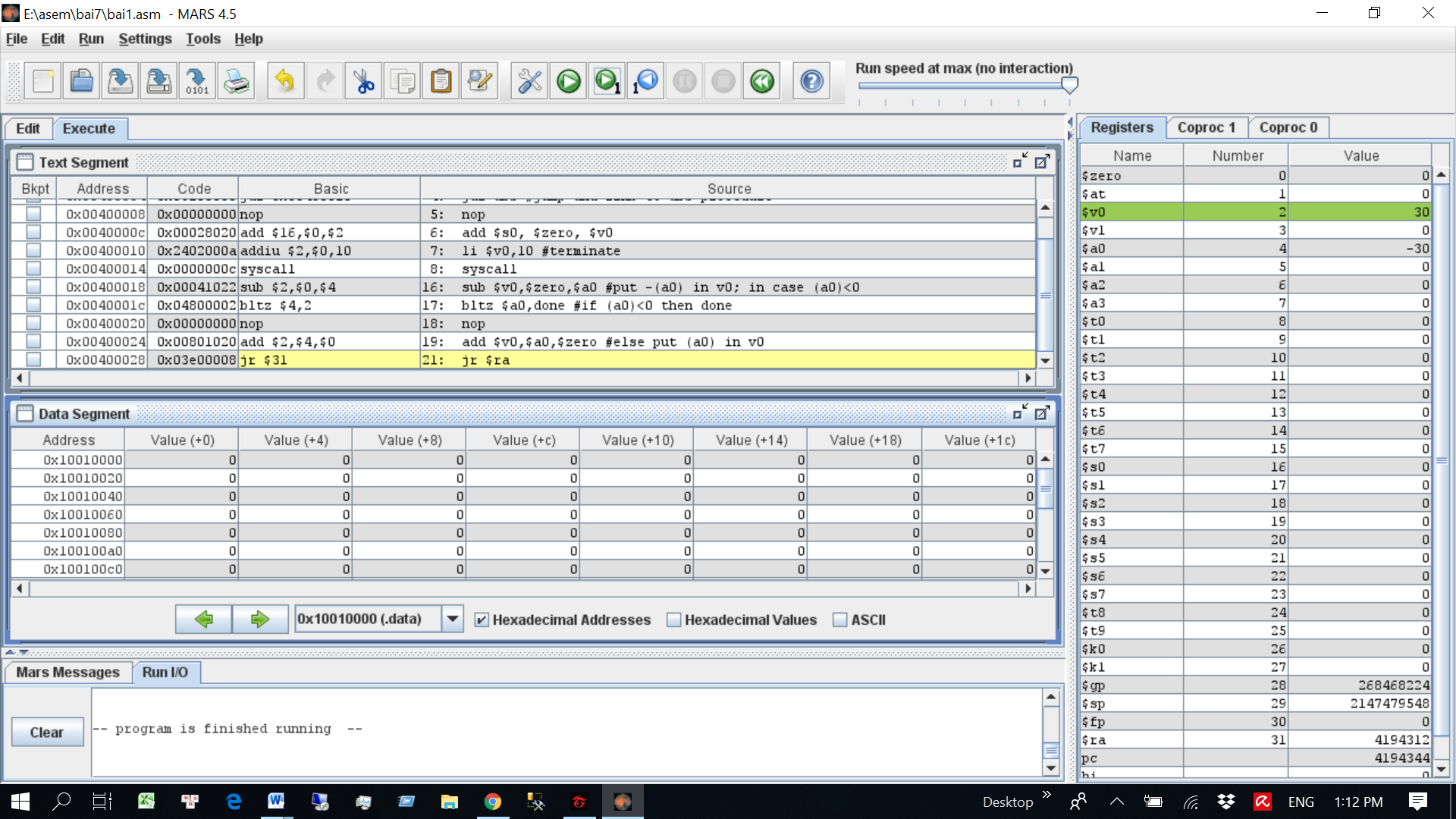
* Lưu $v0 = -$a0 = 30
* Kết quả : 

Câu lệnh tiếp theo :

bltz $a0,done

* Nếu $a0 < 0 t hì chạy câu lệnh done . Vì $a0 gán bằng -30 nên chạy câu lệnh done

Kết quả :

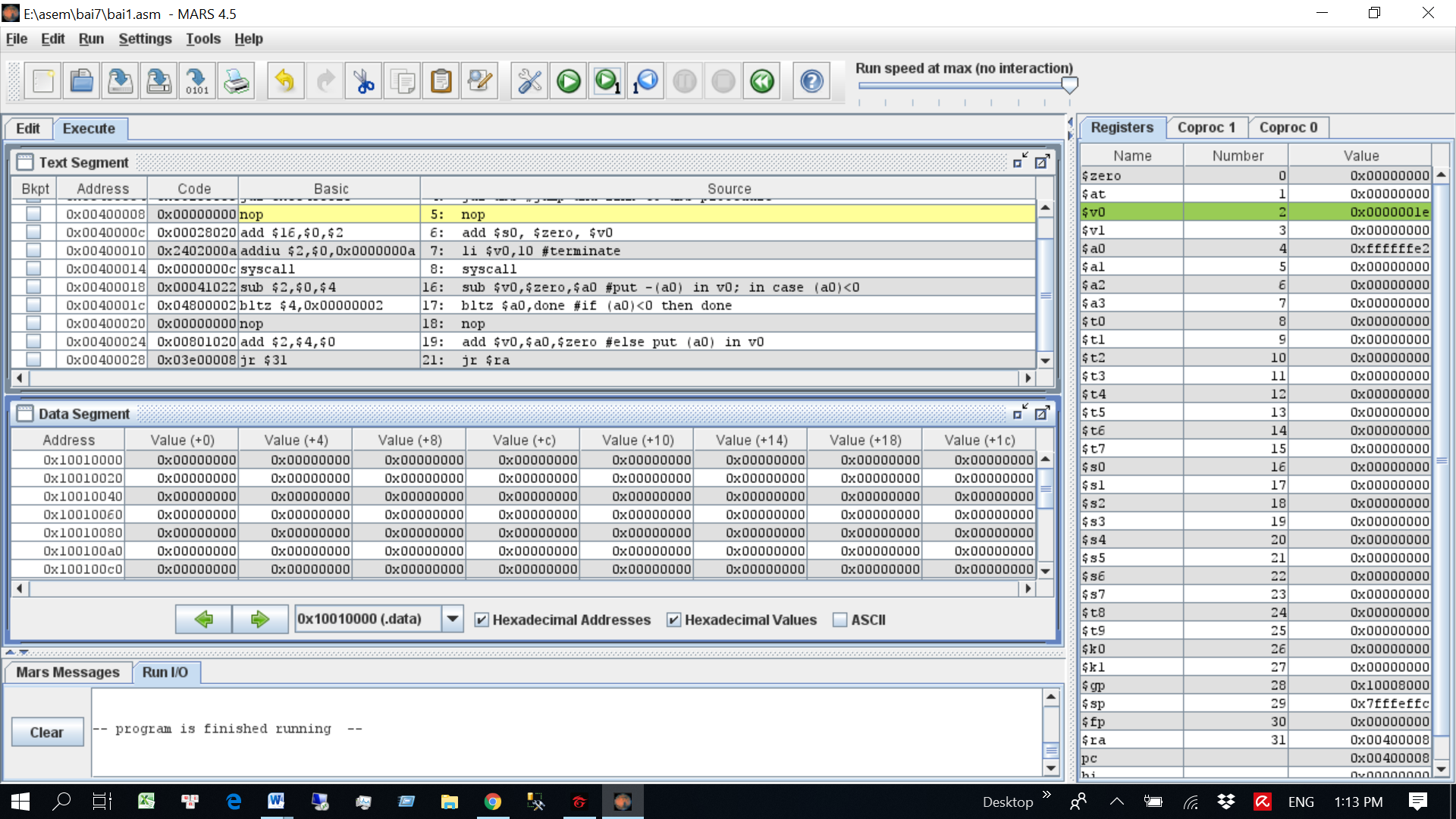


Câu lệnh tiếp theo

Jr $ra

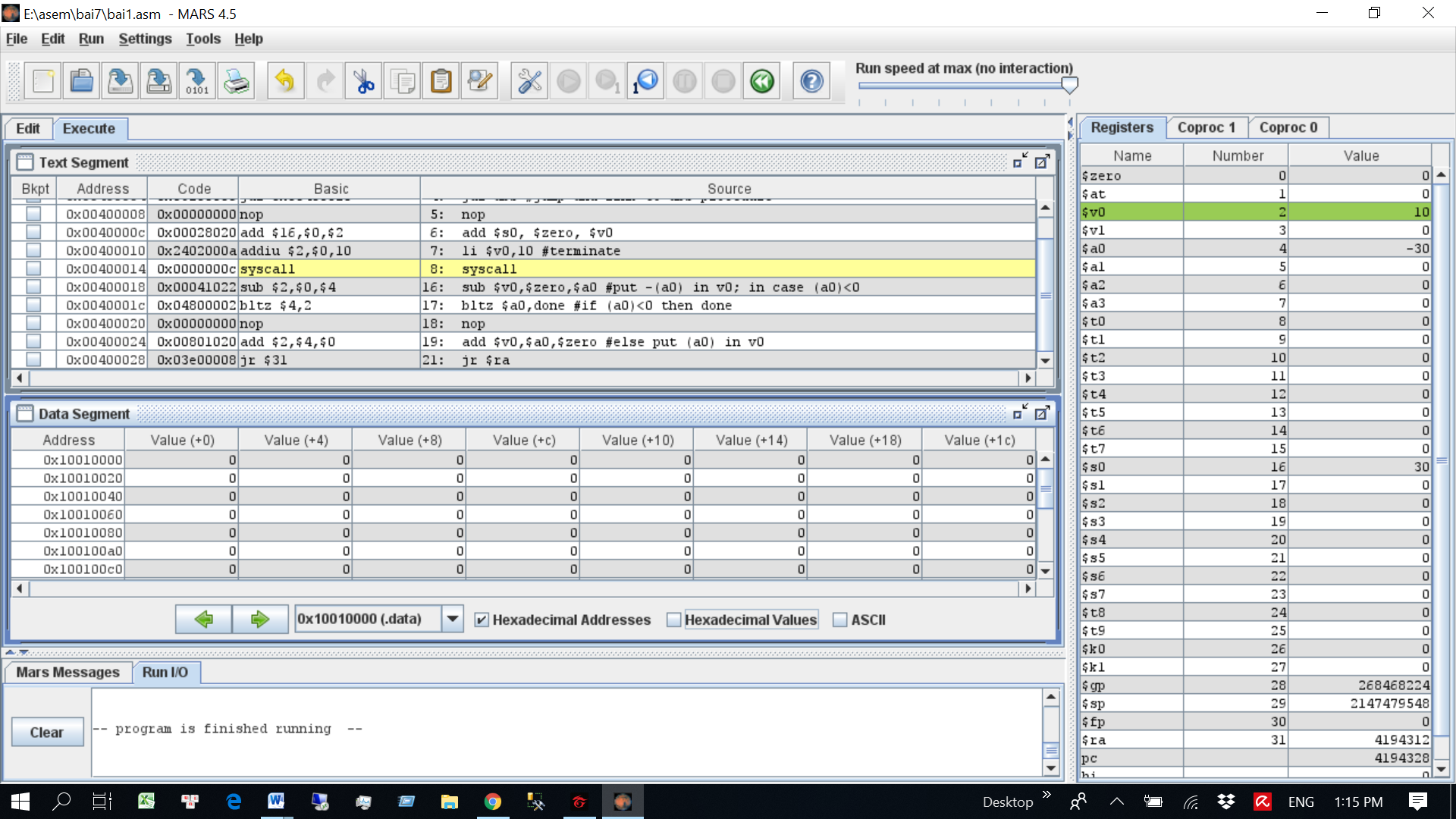
* Câu lệnh để nhảy đến dòng lệnh có địa chỉ được lưu ở thanh ghi $ra

Kết quả :



* Các câu lệnh tiếp theo không có gì mới , add $s0,$rezo,$v0 để gán giá trị $v0 vào $s0

Kết quả toàn bộ chương trình :



Nhận xét tổng kết :

Function trong mips , có thể đưa vào 4 giá trị từ $a0 đến $a3 . trả về tối đa 2 giá trị $v0 và $v1 .

Để có thể tăng lượng dữ liệu truyền vào và lấy ra , ta có thể sử dụng stack ( bài 3 ).