Võ Anh Khôi – 20225870

Exercise 5

Bài 1

#Laboratory Exercise 5, Home Assignment 1

.data

test: .asciiz "Hello World"

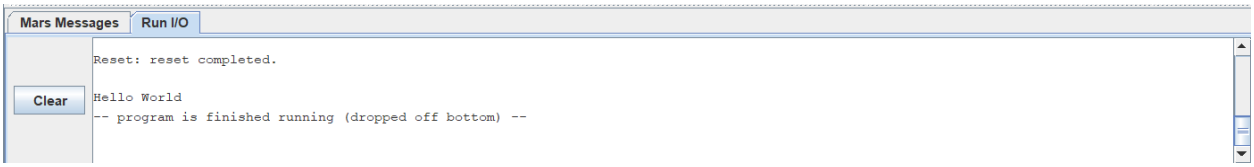
.text

li $v0, 4

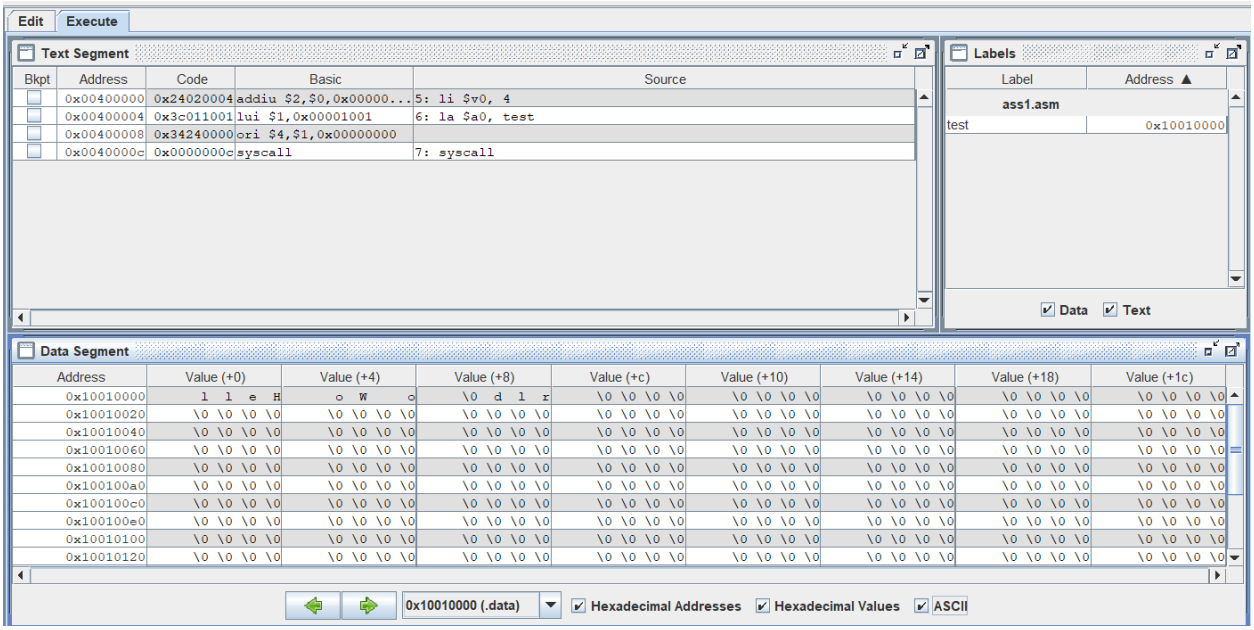
la $a0, test

syscall

kết quả



Chuỗi được hiện lên màn hình output đúng như lý thuyết



Mỗi kí tự trong chuỗi được lưu bằng 1 byte chứa 4 kí tự trong một thanh ghi và được bắt đầu bằng phía bên phải (LSB)

Bài 2

Xét với trường hợp $s0 =13 và $s1 = 5

```
.data
        mess1: .asciiz "The sum of "
        mess2: .asciiz " and "
        mess3: .asciiz " is "
.text
        addi $s0, $0, 13
        addi $s1, $0, 5
        add $s2, $s1, $s0

        addi $v0, $0, 4 #print string “the sum of”
        la $a0, mess1
        syscall

        addi $v0, $0, 1 #print num 1
        add $a0, $0, $s0
        syscall

        addi $v0, $0, 4 #print string “and”
        la $a0, mess2
        syscall

        addi $v0, $0, 1 #print num 2
        add $a0, $0, $s1
        syscall

        addi $v0, $0, 4 #print string “is”
        la $a0, mess3
```

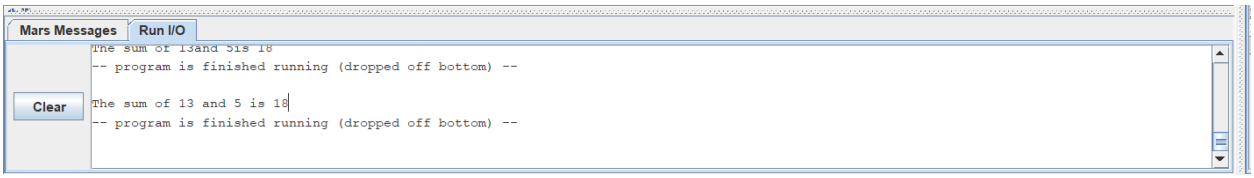
syscall

addi $v0, $0, 1 #print num

add $a0, $0, $s2

syscall

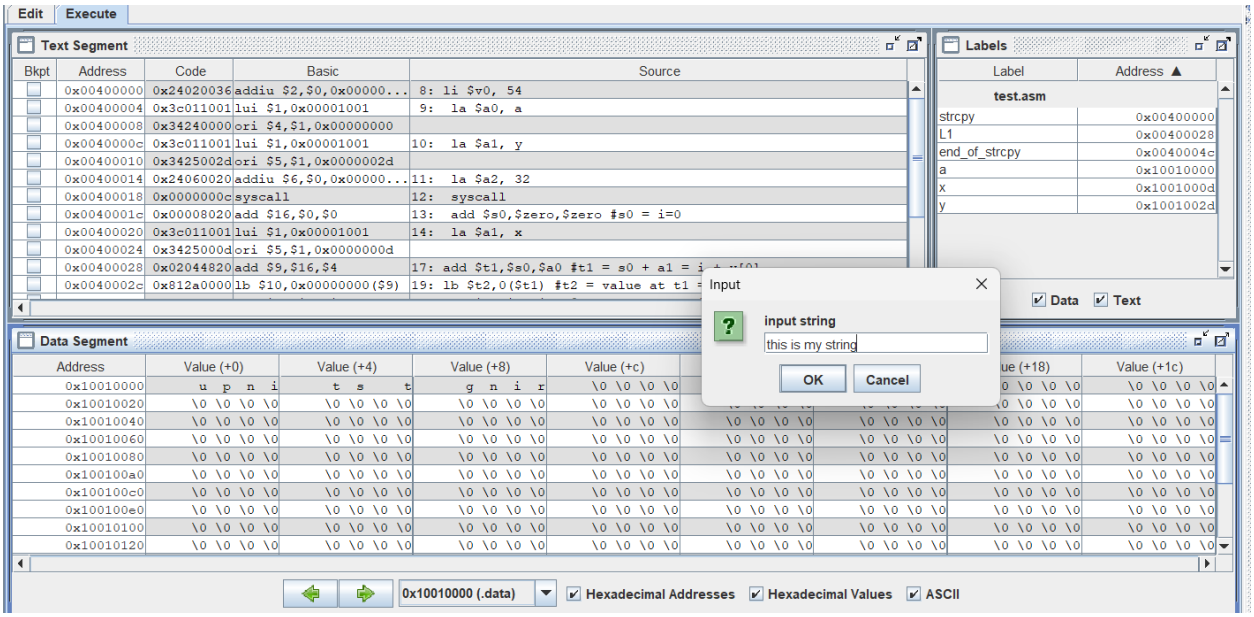
kết quả trả về :



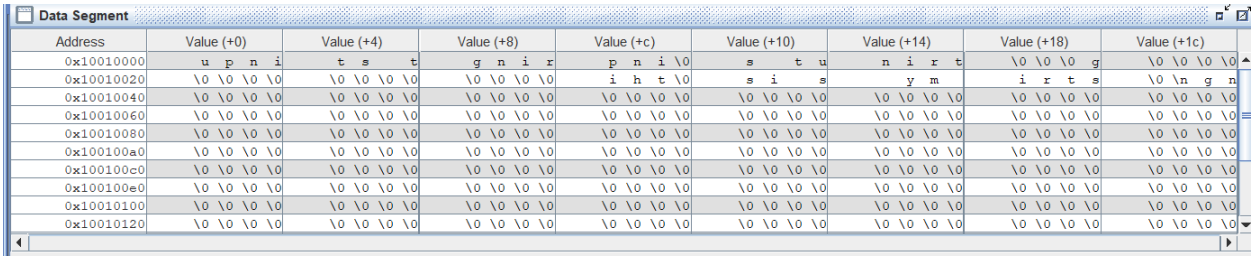
Bài 3

```
a: .asciiz "input string"
x: .space 32 # destination string x, empty
y: .space 32 # source string y
.text
strcpy:
li $v0, 54
 la $a0, a
 la $a1, y
 la $a2, 32
 syscall
 add $s0,$zero,$zero #s0 = i=0
 la $a1, x
L1:
add $t1,$s0,$a0 #t1 = s0 + a1 = i + y[0]
 # = address of y[i]
lb $t2,0($t1) #t2 = value at t1 = y[i]
add $t3,$s0,$a1 #t3 = s0 + a0 = i + x[0]
```

```
 # = address of x[i]
sb $t2,0($t3) #x[i]= t2 = y[i]a
beq $t2,$zero,end_of_strcpy #if y[i]==0, exit
nop
addi $s0,$s0,1 #s0=s0 + 1 <-> i=i+1
j L1 #next character
nop
end_of_strcpy:
```



Kết quả trả về :



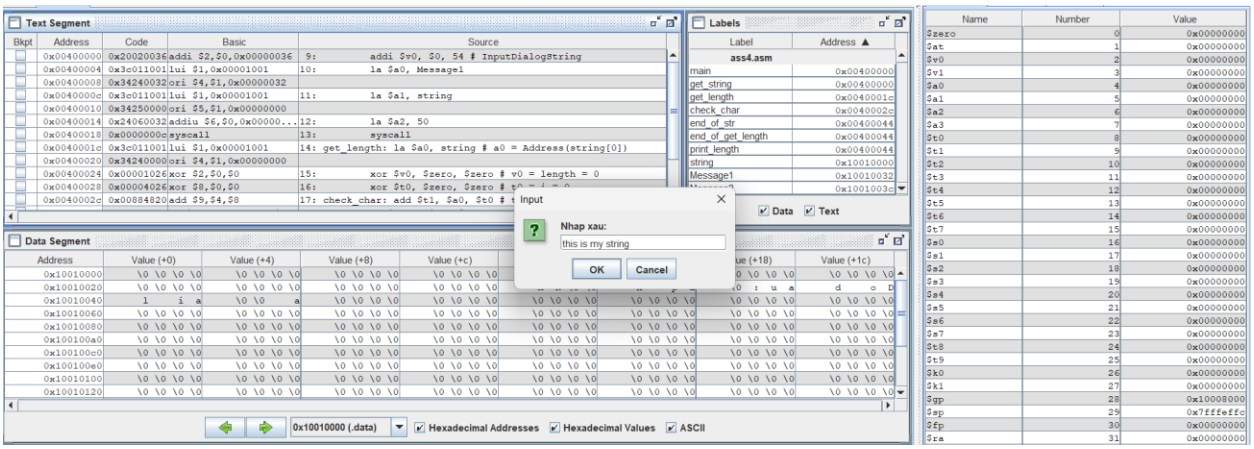
Chuỗi thu được bắt đầu lưu ở vị trí địa chỉ y+32byte

Bài 4

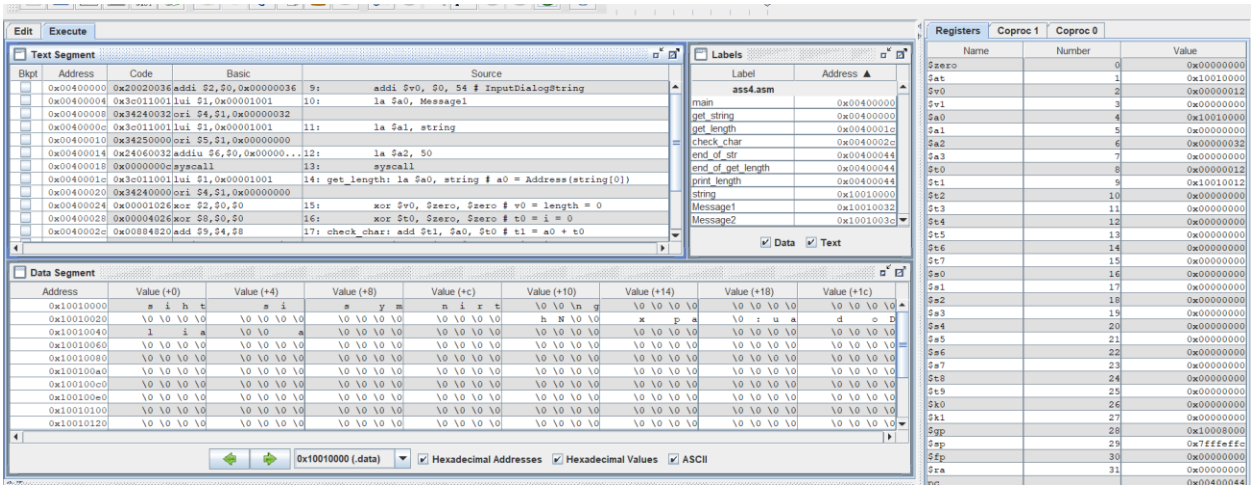
```
#Laboratory Exercise 5, Home Assignment 3
.data
string: .space 50
```

```
Message1: .asciiz "Nhap xau:”
Message2: .asciiz "Do dai la "
.text
main:
get_string: # TODO Đoạn code đã được thêm
        addi $v0, $0, 54 # InputDialogString
        la $a0, Message1
        la $a1, string
        la $a2, 50
        syscall
get_length: la $a0, string # a0 = Address(string[0])
        xor $v0, $zero, $zero # v0 = length = 0
        xor $t0, $zero, $zero # t0 = i = 0
check_char: add $t1, $a0, $t0 # t1 = a0 + t0
 #= Address(string[0]+i)
        lb $t2, 0($t1) # t2 = string[i]
        beq $t2,$zero,end_of_str # Is null char?
        addi $v0, $v0, 1 # v0=v0+1->length=length+1
        addi $t0, $t0, 1 # t0=t0+1->i = i + 1
        j check_char
end_of_str:
end_of_get_length:
print_length: # TODO đoạn code đã được thêm
        add $t9, $0, $v0
        addi $t9, $t9,-1 # -1 do getlength se dem ca ki tu \n
        addi $v0, $0, 56 #
        la $a0, Message2
        add $a1, $t9, $0
        syscall
```

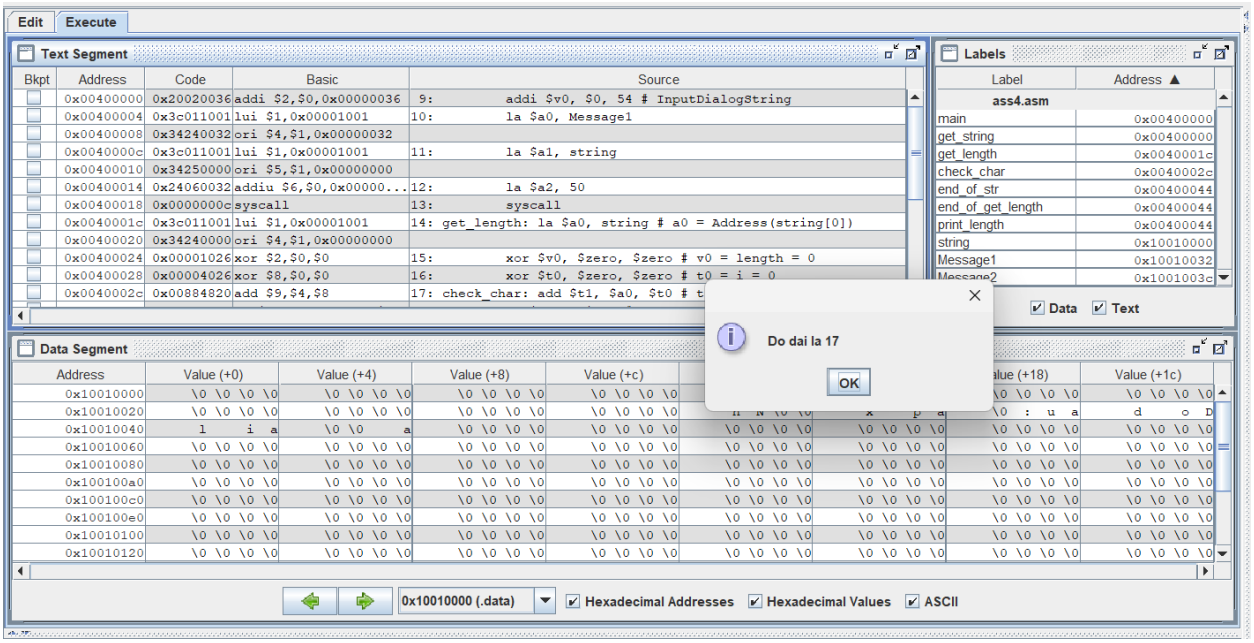
kết quả thu được



Chuỗi được nhập vào được lưu trữ trong bộ nhớ như ảnh dưới



Độ dài của chuỗi thu được



⇨ Đúng với kết quả lý thuyết (nếu đếm cả \n sẽ là 18)

Bài 5

```
.data
        string: .space 20
        message1: .asciiz "nhap ki tu thu "
        message2: .asciiz "\n"
        message3: .asciiz " chuoi ket qua "

.text
        li $t0, 20          # max length
        li $s0, 0           #i
        li $t7, 10          # \n
        la $s1, string

getchar:
        beq $s0, $t0, end_get

        li $v0, 4 #print message 1
        la $a0, message1
        syscall

        addi $t2, $s0, 1 #print “character i+1”
        li $v0, 1
        add $a0, $t2, $0
        syscall

        li $v0, 12 #read character
        syscall
        add $t1, $v0, $0
```

```
        beq $t1, $t7, end_get

        li $v0, 4 #break to new line
        la $a0, message2
        syscall

        add $s7, $s1, $s0 #address of string[i]
        sb $t1, 0($s7)
        addi $s0, $s0, 1 # i++
        j getchar

end_get:
        li $v0, 4          #print message 3
        la $a0, message3
        syscall
printf:
        li $v0, 11 #print character
        lb $a0, 0($s7)
        syscall

        beq $s7,$s1, end #if addrvstring[last] = addstring[first] branch to end
        addi $s7, $s7, -1
        j printf

end:
```

Kết quả thu được : (trang sau)

Với chuỗi có độ dài bé hơn 20:

```
nhap ki tu thu 7h
nhap ki tu thu 8o
nhap ki tu thu 9i
nhap ki tu thu 10
 chuoi ket qua iohkhnaov
-- program is finished running (dropped off bottom) --
```

Với chuỗi có độ dài bằng 20:

```
nhap ki tu thu 17q
nhap ki tu thu 18r
nhap ki tu thu 19s
nhap ki tu thu 20t
 chuoi ket qua tsrqponmlkjihgfedcba
-- program is finished running (dropped off bottom) --
```