**Assignment 5**

.data

input: .space 256

output: .space 256

.text

.globl main

main:

li $v0, 8 # Ask the user for the string they want to reverse

la $a0, input # We'll store it in 'input'

li $a1, 256 # Only 256 chars/bytes allowed

syscall

jal strlen # JAL to strlen function, saves return address to $ra

add $t1, $zero, $v0 # Copy some of our parameters for our reverse function

add $t2, $zero, $a0 # We need to save our input string to $t2, it gets

add $a0, $zero, $v0 # butchered by the syscall.

reverse:

li $t0, 0 # Set t0 to zero

li $t3, 0 # set t3 to zero

reverse\_loop:

add $t3, $t2, $t0 # $t2 is the base address for our 'input' array, add loop index

lb $t4, 0($t3) # load a byte at a time according to counter

beqz $t4, exit # We found the null-byte

sb $t4, output($t1) # Overwrite this byte address in memory

subi $t1, $t1, 1 # Subtract our overall string length by 1 (j--)

addi $t0, $t0, 1 # Advance our counter (i++)

j reverse\_loop # Loop until we reach our condition

exit:

li $v0, 4 # Print

la $a0, output # the string!

syscall

li $v0, 10 # exit()

syscall

strlen: #loops over the character array until it encounters

li $t0, 0

li $t2, 0

strlen\_loop:

add $t2, $a0, $t0

lb $t1, 0($t2) #load byte of t2 and store to t1

beqz $t1, strlen\_exit #if t1 = 0 brand to exit

addiu $t0, $t0, 1 # t0 = t0+1

j strlen\_loop #loop

strlen\_exit:

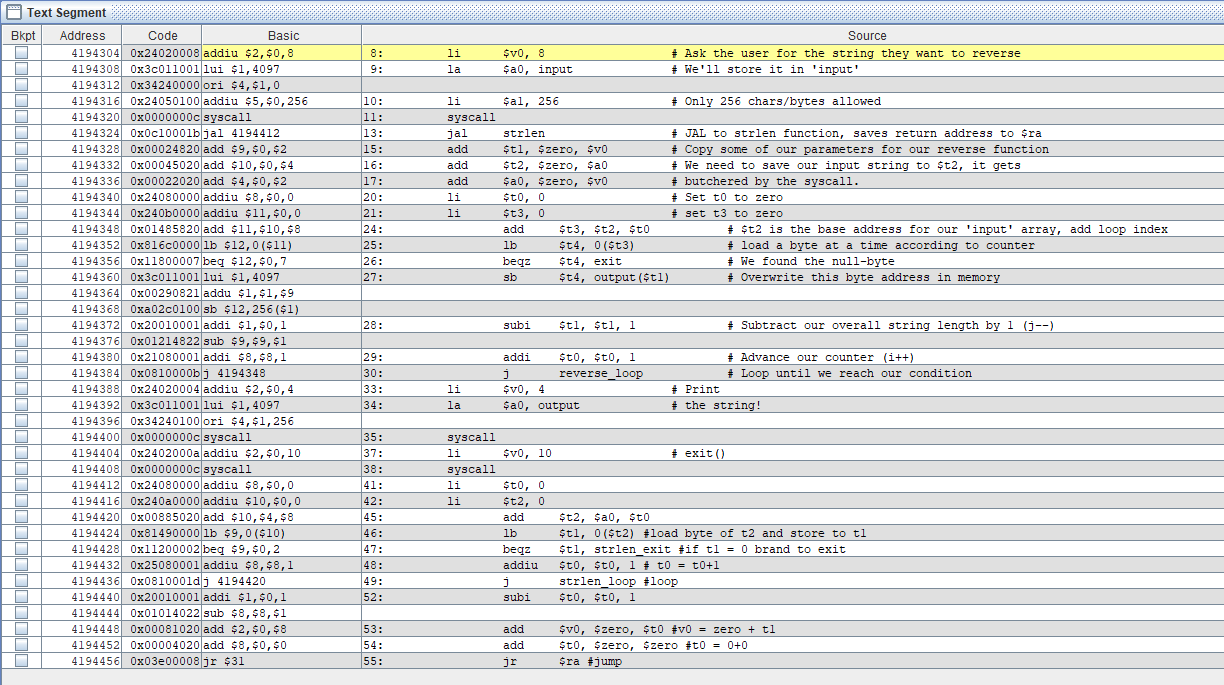
subi $t0, $t0, 1

add $v0, $zero, $t0 #v0 = zero + t1

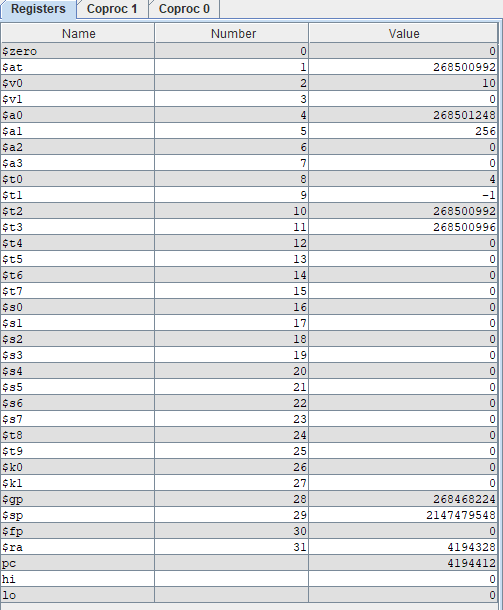
add $t0, $zero, $zero #t0 = 0+0

jr $ra #jump

* **Bảng Text Segment**



* **Bảng giá trị của các thanh ghi**



* **Kết quả**

