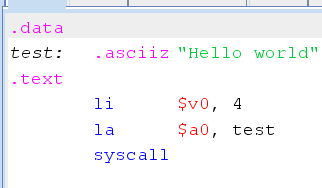
**Báo cáo thực hành KTMT Tuần 5**

**Phạm Thành Lập – 20215076**

**Assignment 1**



Ảnh có chứa văn bản

Mô tả được tạo tự động

* Data Segment

Ảnh có chứa bàn

Mô tả được tạo tự động

* Sự thay đổi giá trị thanh ghi

|  |  |  |  |
| --- | --- | --- | --- |
| Trạng thái | $v0 | $a0 | pc |
| Ban đầu | 0x00000000 | 0x00000000 | 0x00400000 |
| Sau lệnh li | 0x00000004 | - | 0x00400004 |
| Sau lệnh lui | - | - | 0x00400008 |
| Sau lệnh ori | - | 0x10010000 | 0x0040000c |
| Sau lệnh syscall | - | - | 0x00400010 |

* Mỗi kí tự khi được lưu trong thanh ghi sẽ tốn 1 byte trong thanh ghi lên cứ 1 ký tự sẽ được lưu vào 1 địa chỉ của thanh ghi các kí tự tiếp theo sẽ là địa chỉ ban đầu cộng thêm 1, các kí tự trong chuỗi sẽ được nạp vào địa chỉ thanh ghi theo thứ tự từ phải qua trái 2 bit bé nhất để lưu giá trị chữ cái đầu tiên sau đó tăng dần 2 bit cho chữ cái tiếp theo, các kí tự sẽ được đẩy dần cho đến khi lưu trữ đầy đủ chuỗi “hello world”

**Assignment 2**

.text code

.data

Message\_1: .asciiz "Nhap so thu nhat: "

Message\_2: .asciiz "Nhap so thu hai: "

Message\_3: .asciiz "Tong cua "

Message\_4: .asciiz " va "

Message\_5: .asciiz " la: "

Newline: .asciiz "\n"

.text

li $v0, 4

la $a0, Message\_1

syscall #in thong tin Message\_1

li $v0, 5

syscall #doc vao so nguyen

move $s0, $v0 #luu so nguyen vua nhap vao thanh ghi $s0

li $v0, 4

la $a0, Message\_2

syscall #in thong tin Message\_2

li $v0, 5

syscall

move $s1, $v0 #luu so vua nhap vao thanh ghi $s1

add $t1, $s0, $s1 #$t1 = $s0 + $s1

li $v0, 4

la $a0, Message\_3

syscall

li $v0, 1

la $a0, 0($s0)

syscall

li $v0, 4

la $a0, Message\_4

syscall

li $v0, 1

la $a0, 0($s1)

syscall

li $v0, 4

la $a0, Message\_5

syscall

li $v0, 1

la $a0, 0($t1)

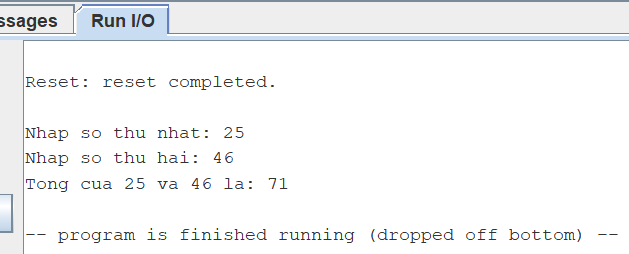
syscall

li $v0, 4

la $a0, Newline

syscall #in ra thong tin tong

.result



**Assignment 3**

Ảnh có chứa văn bản

Mô tả được tạo tự động

Ảnh có chứa bàn

Mô tả được tạo tự độngẢnh có chứa bàn

Mô tả được tạo tự động

* Khởi tạo x là chuỗi rỗng với độ dài là 32 kí tự và y là chuỗi “Hello” sau đó gắn địa chỉ y cho $a1 và x cho $a0
* Khởi tạo $s0 = 0
* Tăng giá trị $t1 chỉ đến địa chỉ chữ cái i trong chuỗi y sau mỗi lần loop
* Lấy giá trị y[i] đưa vào thanh ghi $t2 sau đó sẽ tăng giá trị $t3 chỉ đến địa chỉ chữ cái i trong chuỗi x sau mỗi lần loop
* Lấy dữ từ thanh ghi $t2 đưa vào thanh ghi $t3 sau đó so sánh $t2 với $zero nếu bằng 0 tức là y[i] đang ở kí tự ‘\n’ kết thúc chuỗi thì sẽ nhảy đến thẻ end\_of\_strcpy để kết thúc chương trình và thu được chuỗi x được copy từ x
* Nếu $t2 khác $zero ta sẽ tăng biến con chạy i tương ứng với thanh ghi $t1 lên 1 và lặp lại chương trình
* Sự thay đổi giá trị thanh ghi

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Trạng thái | $a0 | $a1 | $s0 | $t1 | $t2 | $t3 |
| Ban đầu | 0x00000000 | 0x00000000 | 0x00000000 | 0x00000000 | 0x00000000 | 0x00000000 |
| Sau lệnh la x,y | 0x10010000  (địa chỉ của x) | 0x10010020  (địa chỉ của y) | - | - | - | - |
| Sau lệnh beq (loop1) | - | - | - | 0x10010020 | 0x00000048  (tương ứng chữ h) | 0x10010000 |
| Sau lệnh beq (loop 2) | - | - | 0x00000001 | 0x10010021 | 0x00000065  (tương ứng chữ e) | 0x10010001 |
| Sau lệnh beq (loop 3) | - | - | 0x00000002 | 0x10010022 | 0x0000006c  (tương ứng chữ l) | 0x10010002 |
| Sau lệnh beq  (loop 4) | - | - | 0x00000003 | 0x10010023 | - | 0x10010003 |
| Sau lệnh beq  (loop 5) | - | - | 0x00000004 | 0x10010024 | 0x0000006f  (tương ứng chữ o) | 0x10010004 |
| Sau lệnh beq  (loop 6) | - | - | 0x00000005 | 0x10010025 | 0x00000000  (tương ứng với \0 kết thúc chuỗi) | 0x10010005 |

**Assignment 4**

Ảnh có chứa văn bản

Mô tả được tạo tự động

.text code

.data

string: .space 50

Message1: .asciiz "Nhap xau "

Message2: .asciiz "Do dai xau la "

.text

main:

get\_string: li $v0, 54

la $a0, Message1

la $a1, string

la $a2, 50

syscall #in ra thong tin va nhap vao chuoi

get\_length: la $a0, string #gan dia chi string cho $a0

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

lb $t3, 0($a0) #$t3 = s[0]

beq $t3, $zero, end\_of\_get\_length #if s[0] = '\0' (length = 0) jump

check\_char: add $t1, $t0, $a0 #$t1 = s[i]

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

beq $t2, $zero, end\_of\_str #if s[i] = '\0' jump

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

j check\_char

end\_of\_str: subi $t0, $t0, 1 #tru di ki tu '\n'

end\_of\_get\_length:

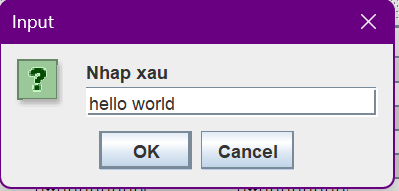
print\_length: li $v0, 56

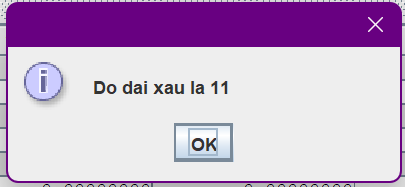
la $a0, Message2

la $a1, 0($t0)

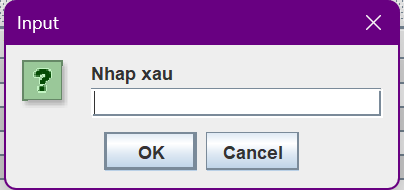
syscall #in ra length

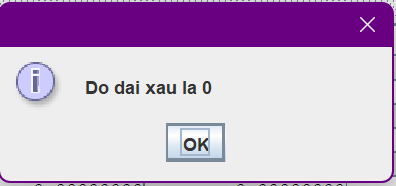
.result



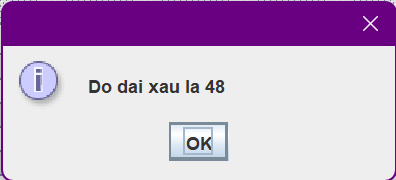


* Trường hợp không nhập gì vào chuỗi:





* Trường hợp nhập quá 50 kí tự:



* Khi chuỗi có độ dài lớn hơn hoặc bằng 50 thì kết quả sẽ trả về 48 do độ dài tối đa của chuỗi là 50 nhưng chuỗi sẽ mất 2 giá trị để lưu dấu Enter nhập từ bàn phím và dấu ‘\0’ kết thúc chuỗi nên độ dài thực tế chỉ là 48

**Assignment 5**

.text code

.data

string: .space 21

reverse: .space 21

Message1: .asciiz "Nhap vao ki tu: "

Message2: .asciiz "Chuoi dao nguoc la:"

Message3: .asciiz "Khong ton tai chuoi dao nguoc"

Newline: .asciiz "\n"

.text

main:

add $s0, $zero, $zero #khoi tao step

la $a1, string #$a1 = string

get\_char:

beq $s0, 20, reverse\_string #if strlen = 20 thuc hien reverse

li $v0, 4

la $a0, Message1

syscall #in ra chuoi nhap ki tu

li $v0, 12

syscall #doc ki tu vao

move $t0, $v0 #luu ky tu vao $t0 ($t0 = c)

beq $t0, 10, check\_eror #neu gap ki tu Enter thi thuc hien len tiep

li $v0, 4

la $a0, Newline

syscall #in ra dau xuong dong

add $t1, $s0, $a1 #$t1 = s[i]

sb $t0, 0($t1) #lay ki tu $t0 nap vao $t1 (s[i] = c)

add $s0, $s0, 1 #i += 1

j get\_char

check\_eror:

beq $s0, $zero, print\_eror #if strlen = 0 in loi

reverse\_string:

subi $s0, $s0, 1 #$s0 -= 1 do i chay tu 0

la $a2, reverse #$a2 = reverse

add $s1, $zero, $zero #i = 0

R1:

sub $t1, $s0, $s1 #j = $s0 - i

blt $t1, $zero, print\_reverse #if j<0 thuc hien print

add $t2, $t1, $a1 #$t2 = s[j]

lb $t3, 0($t2) #$t3 = $t2 = s[j]

add $t4, $s1, $a2 #$t4 = rs[i]

sb $t3, 0($t4) #r[j] = rs[i]

addi $s1, $s1, 1 #i += 1

j R1

print\_reverse:

li $v0, 4

la $a0, Message2

syscall

la $a0, Newline

syscall

la $a0, reverse

syscall #in ra chuoi dao

j end\_main

print\_eror:

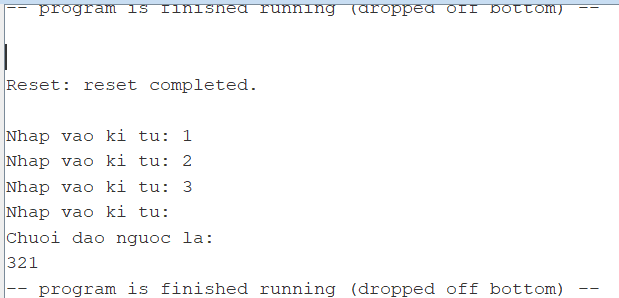
li $v0, 55

la $a0, Message3

syscall #in ra thong bao loi

end\_main:

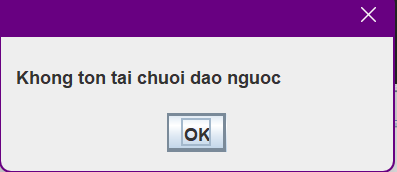
.result



* Trường hợp không nhập gì vào chuỗi

Ảnh có chứa văn bản

Mô tả được tạo tự động



* Trường hợp nhập 20 kí tự

Ảnh có chứa văn bản

Mô tả được tạo tự động