Thực hành Kiến trúc máy tính

Week 5

Vũ Ngọc Đức – 20225816

Assignment 1

.data

test: .asciiz "Hello World"

.text

li $v0, 4 # v0 = 4

la $a0, test # load địa chỉ test vào a0

syscall # gọi hệ thống

.data: định nghĩa phần dữ liệu của chương trình.

test: .asciiz "Hello World": định nghĩa một chuỗi ký tự được lưu trữ trong bộ nhớ, với tên biến là "test" và giá trị là "Hello World". Trong đó, ".asciiz" là một từ khóa để định nghĩa chuỗi ký tự, và "z" cuối cùng đảm bảo rằng chuỗi kết thúc bằng ký tự null.

.text: định nghĩa phần mã của chương trình.

li $v0, 4: gán giá trị 4 cho thanh ghi $v0, xác định loại hệ thống gọi, xác định rằng chương trình sẽ gọi hệ thống gọi in chuỗi.

la $a0, test: lệnh này lấy địa chỉ của biến "test" và lưu nó vào thanh ghi $a0. Thanh ghi $a0 là thanh ghi đặc biệt trong MIPS, được sử dụng để truyền đối số cho các hệ thống gọi.

syscall: lệnh này gọi hệ thống gọi để hiển thị chuỗi "Hello World" trên màn hình console. Nội dung chuỗi cần được in được truyền vào hệ thống gọi thông qua thanh ghi $a0.

Assignment 2

.data

s1: .asciiz "The sum of "

s2: .asciiz " and "

s3: .asciiz " is "

.text

main:

# Nhập giá trị cho $s0

li $v0, 5 # v0 = 5 read integer

syscall

move $s0, $v0 # Lưu giá trị nhập vào vào $s0

# Nhập giá trị cho $s1

li $v0, 5 # v0 = 5 read integer

syscall

move $s1, $v0 # Lưu giá trị nhập vào vào $s1

# Tính tổng của $s0 và $s1

add $t0, $s0, $s1

# In chuỗi "The sum of "

li $v0, 4 # v0 = 4 print string

la $a0, s1

syscall

# In giá trị của $s0

li $v0, 1 # v0 = 1 print integer

move $a0, $s0

syscall

# In chuỗi " and "

li $v0, 4 # v0 = 4 print string

la $a0, s2

syscall

# In giá trị của $s1

li $v0, 1 # v0 = 1 print integer

move $a0, $s1

syscall

# In chuỗi " is "

li $v0, 4 # v0 = 4 print string

la $a0, s3

syscall

# In kết quả của phép cộng

li $v0, 1 # v0 = 1 print integer

move $a0, $t0

syscall

# Kết thúc chương trình

li $v0, 10 #exit

syscall

Assignment 3

.data

x: .space 32 # destination string x, empty

y: .asciiz "Hello" # source string y

.text

li $v0, 4 # v0 = 4 print string

la $a0, x

la $a1, y

strcpy:

add $s0,$zero,$zero # $s0 = i = 0

L1:

add $t1,$s0,$a1 # $t1 = $s0 + $a1 = i + y[0]

# = address of y[i]

lb $t2,0($t1) # $t2 = value at $t1 = y[i]

add $t3,$s0,$a0 # $t3 = $s0 + $a0 = i + x[0]

# = address of x[i]

sb $t2,0($t3) # x[i]= $t2 = y[i]

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:

syscall

li $v0, 4 : gán giá trị 4 vào thanh ghi $v0, chứa mã syscall cho lệnh in chuỗi kí tự

la $a0, x : đặt thanh ghi $a0 trỏ đến vị trí đầu tiên của xâu đích x.

la $a1, y : đặt thanh ghi $a1 trỏ đến vị trí đầu tiên của xâu nguồn y.

Tiếp theo, chương trình bắt đầu quá trình sao chép bằng cách thiết lập giá trị ban đầu cho biến $s0, tương đương với chỉ số của ký tự đầu tiên trong xâu đích.

Trong vòng lặp L1, chương trình sử dụng địa chỉ của ký tự thứ i trong xâu nguồn (y) để sao chép ký tự tương ứng vào vị trí tương ứng trong xâu đích (x).

Sau đó, biến $s0 được tăng lên một đơn vị để chỉ đến ký tự tiếp theo trong xâu đích, và chương trình tiếp tục vòng lặp cho đến khi gặp ký tự NULL kết thúc chuỗi (được biểu diễn bằng giá trị 0).

Cuối cùng, chương trình gọi lệnh syscall để in ra chuỗi kí tự xâu đích đã được sao chép.

Assignment 4

.data

string: .space 50

Message1: .asciiz "Nhap xau: "

Message2: .asciiz "Do dai xau la: "

.text

main:

get\_string:

li $v0, 4 # print string

la $a0, Message1

syscall

li $v0, 8 #read string

la $a0, string

li $a1, 50 #max length

syscall

jal get\_length

get\_length:

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

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

add $v0,$zero,$zero # $v0 = length = 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 $s0, $s0, 1 # v0 = v0 + 1 -> length = length + 1

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

j check\_char

end\_of\_str:

addi $s0, $s0, -1

la $a1, ($s0)

li $v0, 56

syscall

Assignment 5

.data

string: .space 50

stringdao: .space 50

Message1: .asciiz "Nhap xau: "

Message2: .asciiz "Reverse: "

.text

main:

get\_string:

li $v0, 4 # print string

la $a0, Message1

syscall

li $v0, 8 #read string

la $a0, string

li $a1, 50 #max length

syscall

get\_length:

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

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

add $v0,$zero,$zero # $v0 = length = 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 $s0, $s0, 1 # v0 = v0 + 1 -> length = length + 1

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

j check\_char

end\_of\_str:

reverse:

la $a1, stringdao

add $s0, $s0, -1

add $s1, $zero, $zero

L1:

add $t3, $s0, $a0

lb $t4, 0($t3)

add $t5, $s1, $a1

sb $t4, 0($t5)

beq $t4, $zero, end\_of\_reverse

nop

addi $s0, $s0, -1

addi $s1, $s1, 1

j L1

nop

end\_of\_reverse:

li $v0, 59

la $a0, Message2

syscall