

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH



UIT
Trường Đại học
Công nghệ Thông tin

**Khoa Khoa học
và Kỹ thuật Thông tin**

Nguyễn Lê Quỳnh Hương – 21520255

IT012.N11.CNCL

BÁO CÁO:

Lab 4

GVHD: Nguyễn Văn Tín

Hồ Chí Minh, 2022

1. Lý thuyết

2. Thực hành :

Chuyển đoạn code trong bảng theo sau sang MIPS và sử dụng MARS để kiểm tra lại kết quả:

a)

```
if (i == j)
    f = g + h;
else
    f = g - h;
```

Code	Giải thích
IF: bne \$s0,\$s1,ELSE	#if(i==j)
add \$s2,\$t0,\$t1	#f=g+h
j EXIT	#jump EXIT
ELSE: sub \$s2,\$t0,\$t1	#f=g-h
EXIT:	

b)

```
int Sum = 0
for (int i = 1; i <= N; ++i){
    Sum = Sum + i;
}
```

Code	Giải thích
li \$s0, 1	# index = 1
li \$s2, 0	# Sum = 0
Loop:	
bgt \$s0, \$s1, End	#i>n, jump End
add \$s2, \$s2, \$s0	#sum = sum + i
addi \$s0, \$s0, 1	#++i
j Loop	# jump Loop
End:	

3. Bài tập

a. Nhập vào một ký tự, xuất ra cửa sổ I/O của MARS theo từng yêu cầu sau:

- ❖ Ký tự liền trước và liền sau của ký tự nhập vào.
- ❖ Ký tự nhập vào chỉ được phép là ba loại: số, chữ thường và chữ hoa. Nếu ký tự nhập vào rơi vào một trong ba loại, xuất ra cửa sổ đó là loại nào; nếu ký tự nhập không rơi vào một trong ba loại trên, xuất ra thông báo “invalid type”.

Code	Giải thích
<pre> .data string0: .asciiz "Nhap ky tu: " string1: .asciiz"\nKy tu nhap vao la chu hoa" string2: .asciiz"\nKy tu nhap vao la chu thuong" string3: .asciiz"\nKy tu nhap vao la chu so" string4: .asciiz"\nKy tu truoc: " string5: .asciiz"\nKy tu sau: " string6: .asciiz "\ninvalid type" .text li \$v0, 4 la \$a0, string0 syscall li \$v0, 12 syscall move \$a1, \$v0 KTChuHoa: li \$s1, 'A' li \$s2, 'Z' bge \$a1, \$s1, KTChuHoa1 j KTSo KTChuHoa1: ble \$a1,\$s2,LaChuHoa j KTChuThuong LaChuHoa: li \$v0, 4 la \$a0, string1 </pre>	<pre> # Khai báo vùng nhớ data # Khai báo chuỗi string0 # Khai báo chuỗi string1 # Khai báo chuỗi string2 # Khai báo chuỗi string3 # Khai báo chuỗi string4 # Khai báo chuỗi string5 # Khai báo chuỗi string6 # Khai báo vùng nhớ chứa mã lệnh #Syscall in chuỗi string0 #Syscall nhập ký tự #Gán A cho thanh ghi s1 #Gán Z cho thanh ghi s2 #Ký tự >=A, jump KTChuHoa1 #jump KTSo #Ký tự <=Z, jump LaChuHoa #jump KTChuThuong #Syscall in chuỗi string1 </pre>

syscall j Xuat KTChuThuong: li \$s1, 'a' li \$s2, 'z' bge \$a1,\$s1,KTChuThuong1 j KhongThuoc KTChuThuong1: ble \$a1,\$s2,LaChuThuong j KhongThuoc LaChuThuong: li \$v0, 4 la \$a0, string2 syscall j Xuat KTSol: li \$s1, '0' li \$s2, '9' bge \$a1, \$s1, KTSol1 j KhongThuoc KTSol1: ble \$a1,\$s2,LaSo j KhongThuoc LaSo: li \$v0, 4 la \$a0, string3 syscall j Xuat KhongThuoc: li \$v0, 4 la \$a0, string6	#jump Xuat #Gán a cho thanh ghi s1 #Gán z cho thanh ghi s2 #Ký tự >=a, jump KTChuThuong1 #jump KhongThuoc #Ký tự <=z, jump LaChuThuong #jump KhongThuoc #Syscall in chuỗi string2 #jump Xuat #Gán 0 cho thanh ghi s1 #Gán 9 cho thanh ghi s2 #Ký tự >=0, jump KTSol1 #jump KhongThuoc #Ký tự <=9, jump LaSo #jump KhongThuoc #Syscall in chuỗi string3 #jump Xuat #Syscall in chuỗi string6
--	--

syscall j EndProgram Xuất: li \$v0, 4 la \$a0, string4 syscall li \$v0, 11 subi \$a0, \$a1, 1 syscall li \$v0, 4 la \$a0, string5 syscall li \$v0, 11 addi \$a0, \$a1, 1 syscall EndProgram:	#jump EndProgram #Syscall in chuỗi string4 #Syscall in ra 1 ký tự #In ra \$a0 = ký tự – 1 #Syscall in chuỗi string5 #Syscall in ra 1 ký tự #In ra \$a0 = ký tự + 1
---	--

b. Nhập từ bàn phím 2 số nguyên, in ra cửa sổ I/O của MARS theo từng yêu cầu sau:

- ❖ Số lớn hơn
- ❖ Tổng, hiệu, tích và thương của hai số

Code	Giải thích
<pre> .data inCompare: .asciiz "So lon hon : " inSum: .asciiz "\nTong : " inSub: .asciiz "\nHieu : " inMul: .asciiz "\nTich : " inDiv: .asciiz "\nThuong : " inLoi: .asciiz "\nKhong the chia cho so 0" .text li \$v0, 5 syscall move \$t0, \$v0 li \$v0, 5 syscall move \$t1, \$v0 bgt \$t0, \$t1, AMoreThanB li \$v0, 4 la \$a0,inCompare syscall li \$v0, 1 move \$a0, \$t1 syscall j Calculation AMoreThanB: li \$v0, 4 la \$a0,inCompare syscall li \$v0, 1 move \$a0, \$t0 syscall Calculation : add \$t2, \$t0, \$t1 sub \$t3, \$t0, \$t1 </pre>	<pre> # Khai báo vùng nhớ data # Khai báo chuỗi inCompare # Khai báo chuỗi inSum # Khai báo chuỗi inSub # Khai báo chuỗi inMul # Khai báo chuỗi inDiv # Khai báo chuỗi chia lỗi # Khai báo vùng nhớ chứa mã lệnh # Syscall nhập vào a # \$t0 = \$v0 (a = \$t0) # Syscall nhập vào b # \$t1 = \$v0 (b = \$t1) # If (a > b) => Jump AMoreThanB # Else => Syscall in chuỗi inCompare #Syscall in giá trị b # jump Calculation #Syscall in chuỗi inCompare #Syscall in giá trị a # sum = a + b # sub = a - b </pre>

mul \$t4, \$t0, \$t1	# mul = a * b
li \$v0, 4	# Syscall in chuỗi inSum
la \$a0, inSum	
syscall	
li \$v0, 1	# Syscall in giá trị sum
move \$a0, \$t2	
syscall	
li \$v0, 4	# Syscall in chuỗi inSub
la \$a0, inSub	
syscall	
li \$v0, 1	# Syscall in giá trị sub
move \$a0, \$t3	
syscall	
li \$v0, 4	# Syscall in chuỗi inMul
la \$a0, inMul	
syscall	
li \$v0, 1	# Syscall in giá trị mul
move \$a0, \$t4	
syscall	
beqz \$t1, Loi	# If (b == 0) => jump label Loi
li \$v0, 4	# Syscall in chuỗi inDiv
la \$a0, inDiv	
syscall	
li \$v0, 1	# Syscall in giá trị của div
div \$a0, \$t0, \$t1	
syscall	
j EndProgram	# jump label EndProgram
Loi:	
li \$v0, 4	# Syscall xuất string inLoi
la \$a0, inLoi	
syscall	
EndProgram:	