

Báo cáo thực hành LAB4

GVHD: Phan Đình Duy

Họ và tên: Đỗ Phương Duy

MSSV: 23520362

Lớp: IT012.021.KHTN

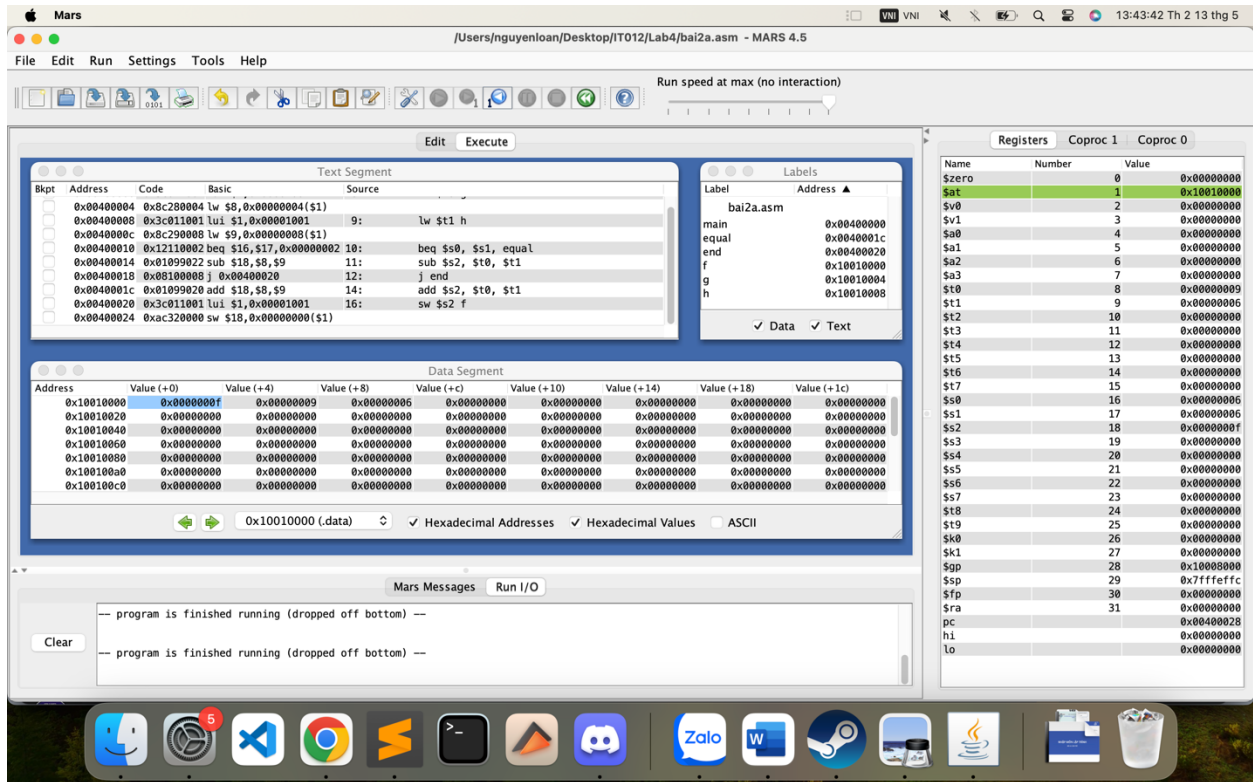
2. Thực hành:

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

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

*Code và kết quả:

```
.data
    f: .word 0
    g: .word 9
    h: .word 6
.text
main:
    lw $t0 g
    lw $t1 h
    beq $s0, $s1, equal
    sub $s2, $t0, $t1
    j end
equal:
    add $s2, $t0, $t1
end:
    sw $s2 f
```



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

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

(Với giá trị của i , N , Sum lần lượt chứa trong các thanh ghi $\$s0$, $\$s1$, $\$s2$)

*Code và kết quả chạy:

```

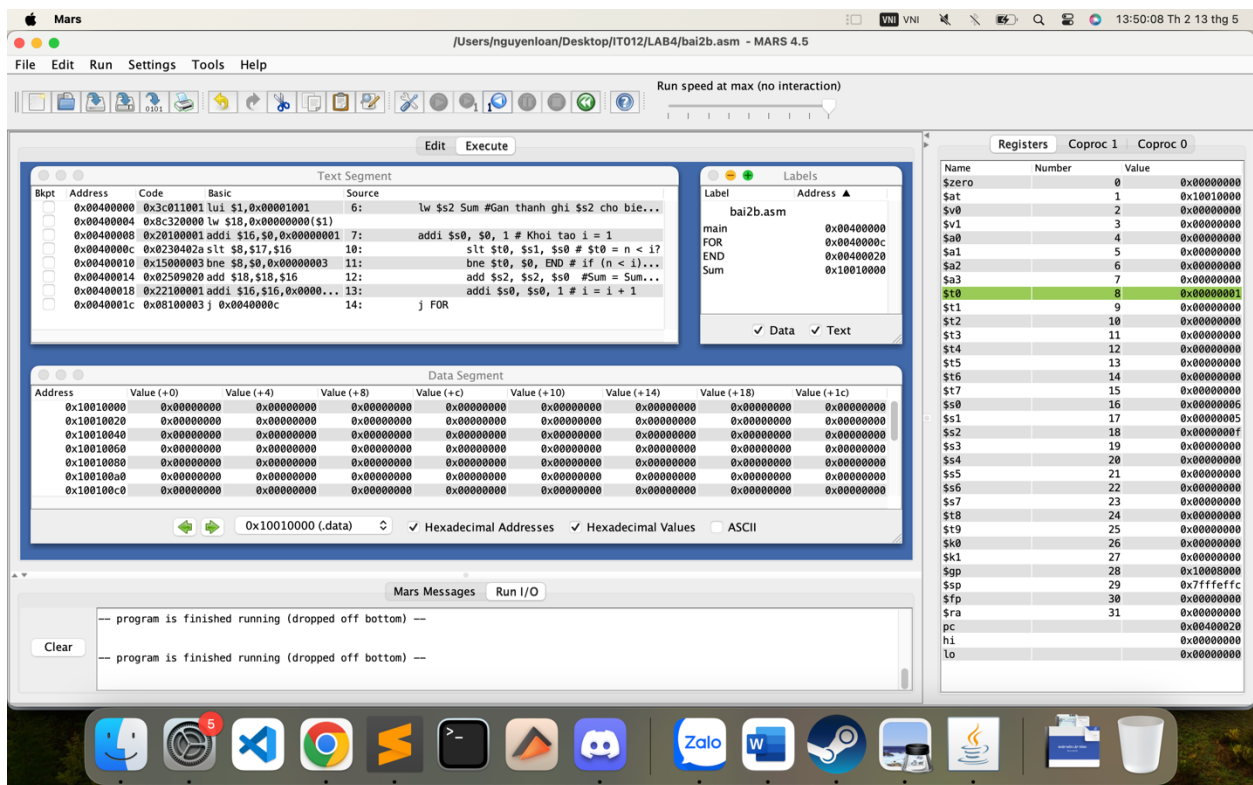
.data
Sum: .word 0 # Khai bao Sum = 0

.text

main:
    lw $s2 Sum #Gan thanh ghi $s2 cho bien Sum
    addi $s0, $0, 1 # Khoi tao i = 1

    FOR:
        slt $t0, $s1, $s0 # $t0 = n < i?
        bne $t0, $0, END # if (n < i) go to end
        add $s2, $s2, $s0 #Sum = Sum + i
        addi $s0, $s0, 1 # i = i + 1
    j FOR
    END:

```



3. Bài tập:

- 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

Ví dụ:

Nhap ky tu (chỉ một ký tự): b

Ky tu truoc: a

Ky tu sau: c

✓ 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:**

.data

kotruoc: .asciiz "Khong ton tai ky tu truoc\n"

kosau: .asciiz "Khong ton tai ky tu sau\n"

nhap: .asciiz "Nhap ky tu (chỉ một ký tự): "

truoc: .asciiz "Ky tu truoc: "

sau: .asciiz "Ky tu sau: "

khac: .asciiz "Invalid type"

endl: .asciiz "\n"

in: .asciiz "Ky tu la in hoa"

thuong: .asciiz "Ky tu la chu thuong"

so: .asciiz "Ky tu la so"

.text

li \$v0, 4

la \$a0, nhap

syscall

li, \$v0, 12

syscall

move \$s0, \$v0

li \$v0, 4

la \$a0, endl

syscall

jal check_upper

jal check_lower

jal check_number

jal special

j end_proc

check_upper:

blt \$s0, 'A', end_func

bgt \$s0, 'Z', end_func

li \$t2, 'A'

```
jal front
li $t2, 'Z'
jal back
```

```
li $v0, 4
la $a0, in
syscall
j end_proc
```

```
check_lower:
    blt $s0, 'a', end_func
    bgt $s0, 'z', end_func
    li $t2, 'a'
    jal front
    li $t2, 'z'
    jal back
    li $v0, 4
    la $a0, thuong
    syscall
    j end_proc
```

```
check_number:
    blt $s0, '0', end_func
    bgt $s0, '9', end_func
    li $t2, '0'
    jal front
    li $t2, '9'
    jal back
    li $v0, 4
    la $a0, so
    syscall
    j end_proc
```

```
special:
    li $v0, 4
    la $a0, khac
    syscall
    jr $ra
```

```
front:
    addi $a0, $s0, -1
    blt $a0, $t2, do
    li $v0, 4
    la $a0, truoc
    syscall
    li $v0, 11
```

```
addi $a0, $s0, -1  
syscall
```

```
li $v0, 4  
la $a0, endl  
syscall  
jr $ra
```

do:

```
li $v0, 4  
la $a0, kotruoc  
syscall  
jr $ra
```

back:

```
addi $a0, $s0, 1  
bgt $a0, $t2, do2  
li $v0, 4  
la $a0, sau  
syscall  
li $v0, 11  
addi $a0, $s0, 1  
syscall  
li $v0, 4  
la $a0, endl  
syscall  
jr $ra
```

do2:

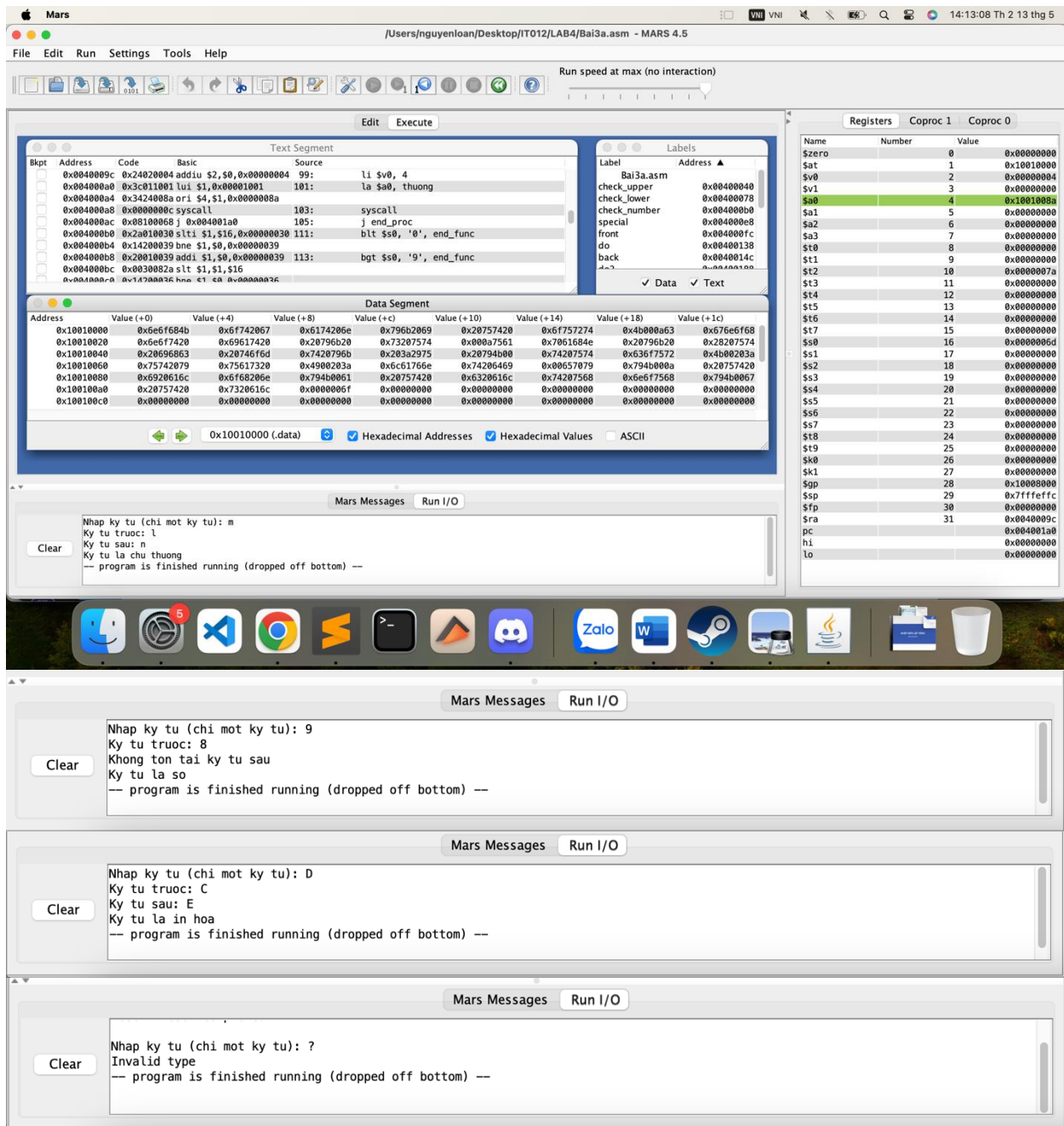
```
li $v0, 4  
la $a0, kosau  
syscall  
jr $ra
```

end_func:

```
jr $ra
```

end_proc:

***Kết quả chạy:**



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:

.data

Nhapa: .ascii "Nhap a: "
Nhapb: .ascii "Nhap b: "
LonHon: .ascii "So lon hon la: "
BangNhau: .ascii "Hai so bang nhau va cung bang: "
Tong: .ascii "\nTong: "
Hieu: .ascii "\nHieu: "
Tich: .ascii "\nTich: "
Thuong: .ascii "\nThuong: "
newline: .ascii "\n"

.text

addi \$v0, \$0, 4
la \$a0, Nhapa
syscall

addi \$v0, \$0, 5
syscall

move \$s1, \$v0
addi \$v0, \$0, 4
la \$a0, Nhapb
syscall

addi \$v0, \$0, 5
syscall

move \$s2, \$v0
slt \$t1, \$s1, \$s2
beq \$t1, 1, bLon
bne \$s1, \$s2, aLon
addi \$v0, \$0, 4
la \$a0, BangNhau
syscall
addi \$v0, \$0, 1
add \$a0, \$s1, \$0
syscall
j ketqua

aLon:

add \$t0, \$s1, \$0
j InSoLonHon

bLon:

add \$t0, \$s2, \$0

InSoLonHon:

addi \$v0, \$0, 4
la \$a0, LonHon
syscall
addi \$v0, \$0, 1
add \$a0, \$t0, \$0
syscall

ketqua:

addi \$v0, \$0, 4
la \$a0, Tong
syscall

add \$t0, \$s1, \$s2
add \$a0, \$t0, \$0
addi \$v0, \$0, 1
syscall

addi \$v0, \$0, 4
la \$a0, Hieu
syscall

sub \$t0, \$s1, \$s2
add \$a0, \$t0, \$0
addi \$v0, \$0, 1
syscall

addi \$v0, \$0, 4
la \$a0, Tich
syscall

mul \$t0, \$s1, \$s2

add \$a0, \$t0, \$0

addi \$v0, \$0, 1

syscall

addi \$v0, \$0, 4

la \$a0, Thuong

syscall

mtc1 \$s1, \$f1

cvt.s.w \$f1, \$f1

mtc1 \$s2, \$f2

cvt.s.w \$f2, \$f2

div.s \$f12, \$f1, \$f2

addi \$v0, \$0, 2

syscall

*Kết quả:

