

Họ và tên: Vũ Minh Hiếu
MSSV: 20194563

Báo cáo thực hành giữa kỳ

Môn Kiến trúc máy tính

Tuần 8

Bài 1:

Nhập 2 số nguyên dương M và N từ bàn phím.
In ra màn hình ước số chung lớn nhất của M và N.

Code:

```
# Nhập 2 số nguyên dương M và N từ bàn phím.
# In ra màn hình ước số chung lớn nhất của M và N.

.data
msg_m:    .ascii "Nhap nguyen duong M: "
msg_n:    .ascii "Nhap nguyen duong N: "
msg_out:  .ascii "Uoc chung lon nhat cua "
msg_and:  .ascii " va "
msg_is:   .ascii " la: "
.text

mInput:
    # Nhập M
    li $v0, 51
    la $a0, msg_m
    syscall

    sne $t0, $a1, $zero    # Ktra nhap thanh cong
    beq $t0, 1, mInput
    slti $t0, $a0, 0       # Kiem tra M > 0?
    bne $t0, $zero, mInput
```

```
addi $t0, $a0, 0    # $t0 = M
add $s0, $t0, $zero
```

nInput:

```
# Nhap N
li $v0, 51
la $a0, msg_n
syscall
```

```
sne $t1, $a1, $zero    # Ktra nhap thanh cong
beq $t1, 1, nInput
slti $t1, $a0, 0    # Kiem tra N > 0?
bne $t1, $zero, nInput
```

```
addi $t1, $a0, 0    # $t1 = N
add $s1, $t1, $zero
```

loop:

```
seq $t2, $t0, $t1
beq $t2, 1, print
sgt $t2, $t0, $t1    # $t2=1 neu M>N
beq $t2, 1, MgtN
slt $t2, $t0, $t1    # $t2=1 neu N<M
beq $t2, 1, NgzM
```

MgtN:

```
sub $t2, $t0, $t1    # M->M-N
add $t0, $t2, $zero
j loop
```

NgzM:

```
sub $t2, $t1, $t0    # N->N-M
add $t1, $t2, $zero
j loop
```

print:

```
# In "Uoc chung lon nhat cua "
la $a0, msg_out
li $v0, 4
syscall
```

```
# In M
```

```
la $a0, ($s0)
li $v0, 1
syscall

# In " va "
la $a0, msg_and
li $v0, 4
syscall

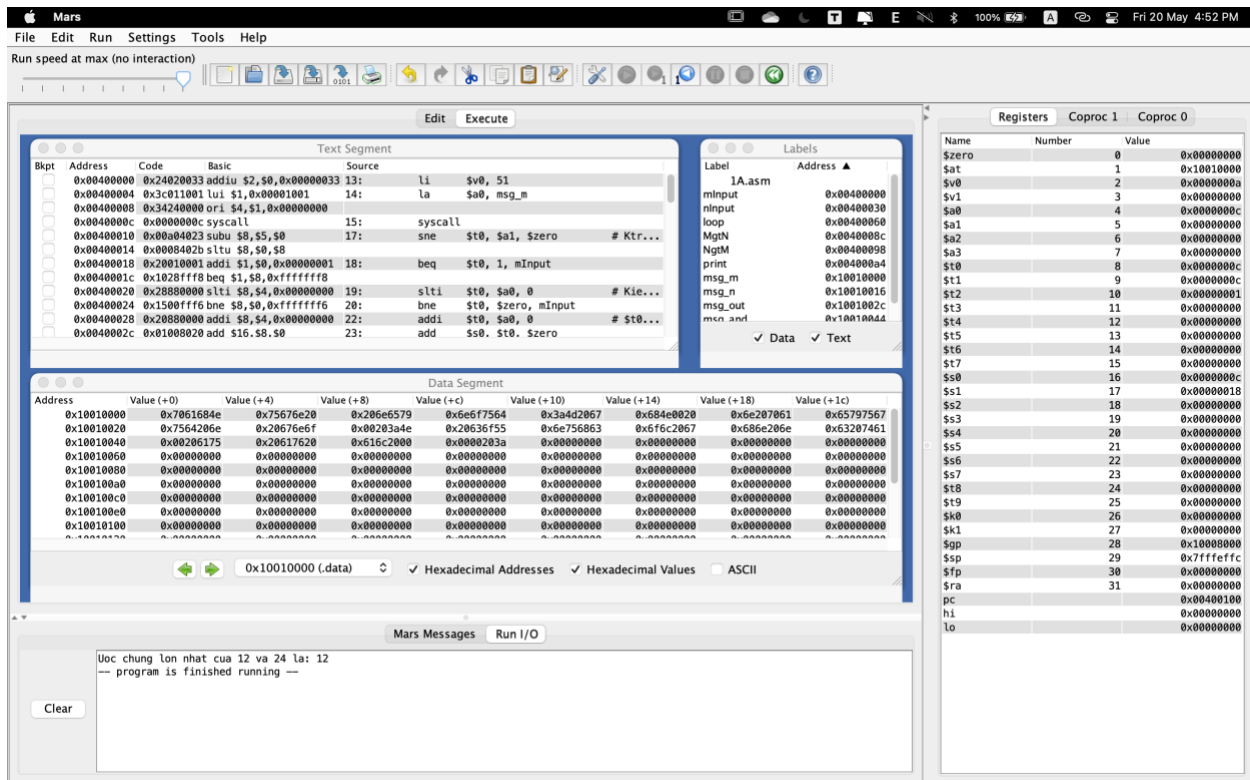
# In N
la $a0, ($s1)
li $v0, 1
syscall

# In "la "
la $a0, msg_is
li $v0, 4
syscall

# In ket qua
la $a0, ($t0)
li $v0, 1
syscall

#Thoat chuong trinh
li $v0, 10
syscall
```

Chạy thử chương trình:



Lần thử 1:

Mở dialog "Nhập M:" -> Nhập -12

Mở lại dialog "Nhập M:" -> Nhập 12

Mở dialog "Nhập N:" -> Nhập 24

Ước chung lớn nhất của 12 và 24 là: 12

Bài 2:

Nhập mảng số nguyên từ bàn phím.

Sắp xếp mảng sao cho các số âm đứng đầu dãy, các số dương đứng cuối dãy

Code:

```
# Nhập mảng số nguyên từ bàn phím.
# Sắp xếp mảng sao cho các số âm đứng đầu dãy, các số dương đứng cuối dãy.

.data
msg_length: .asciiz "Nhap vao kich thuoc mang: "
msg_element: .asciiz "Nhap danh sach cac phan tu (bam enter giua cac phan tu): \n"
msg_solved: .asciiz "Mang sau khi sap xep la: "

.text
input_length:
    li $v0, 4
    la $a0, msg_length
    syscall
    li $v0, 5      # $v0 chua so nhap vao
    syscall
    add $a1, $zero, $v0  # $a1 chua kich thuoc cua A[]

input_element:
    li $v0, 4
    la $a0, msg_element
    syscall

loop:
    beq $s1, $a1, end_loop
    li $v0, 5      # Nhap A[i]
    syscall
    slt $t4, $v0, $zero
    bne $t4, $zero, continue
    add $a3, $zero, $v0  # Khoi tao gia tri nho nhat

continue:
    sw $v0, 0($sp)    # Luu gia tri vua nhap vao A[i]
    add $s1, $s1, 1
```

```
    sub $sp, $sp, 4      # Tro den phan tu A[i+1]
    j    loop
    nop

end_loop:

main:

end_main:
    li   $v0, 10
    syscall
```

Bài 3:

Nhập vào chuỗi ký tự

In ra màn hình số ký tự nguyên âm có trong chuỗi.

Code:

```
# Nhập vào chuỗi ký tự, in ra màn hình số ký tự nguyên âm có trong chuỗi.

.data
vowel:      .ascii "aeiou"
msg_prompt: .ascii "Nhap xau ky tu: "
msg_out:    .ascii "So ky tu nguyen am la: "
msg_nl:     .ascii "\n"
str:        .space 256      # Buffer 100 byte chua string
.text

main:
    # In "Nhap xau ky tu: "
    li      $v0, 4
    la      $a0, msg_prompt
    syscall

    # Nhap string
    li      $v0, 8
    la      $a0, str
    li      $a1, 100
    syscall

    li      $s2, 0           # vowelCount = 0
    la      $s0, str         # Dia chi tro den string

string_loop:
    lb      $t0, 0($s0)      # t0 = str[i]
    addiu   $s0, $s0, 1      # i = i + 1
    beqz    $t0, string_done # Da duyet het xau?

    la      $s1, vowel       # Tro den vowel

vowel_loop:
    lb      $t1, 0($s1)      # t1 = vowel[j]
```

```

    beqz    $t1, string_loop      # Duyệt hết string vowel?
    addiu   $s1, $s1, 1          # j = j + 1

    # Kiểm tra có phải nguyên âm hay không?
    bne     $t0, $t1, vowel_loop # str[i] == vowel[j]
    addi    $s2, $s2, 1          # Là nguyên âm -> vowelCount++
    j       string_loop         # Duyệt ký tự tiếp theo

string_done:                      # Đã duyệt hết xâu
    # In "Số ký tự nguyên âm là: "
    li     $v0, 4
    la     $a0, msg_out
    syscall

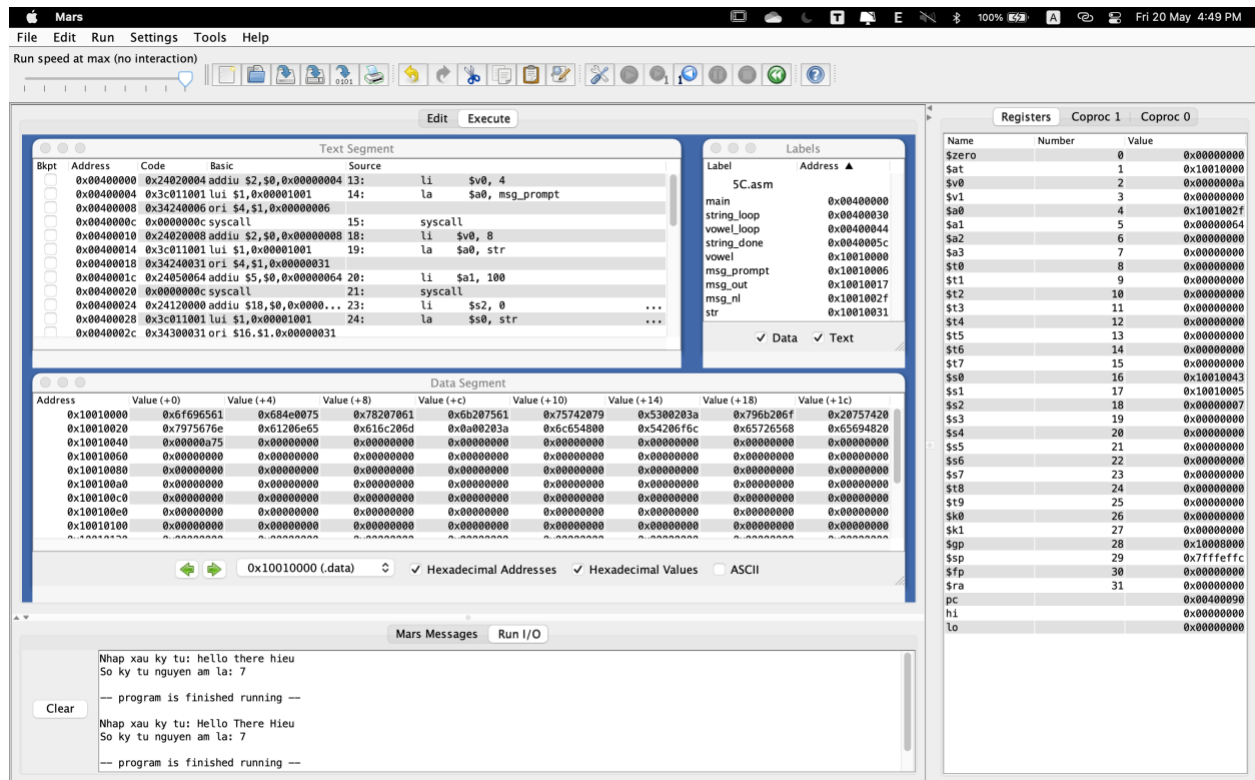
    # In số ký tự nguyên âm
    li     $v0, 1
    move   $a0, $s2
    syscall

    # Xuống dòng mới
    li     $v0, 4
    la     $a0, msg_nl
    syscall

    # Thoát chương trình
    li     $v0, 10
    syscall

```


Chạy thử chương trình:



Lần thử 1:

Nhập xâu ký tự: hello there hieu
Số ký tự nguyên âm là: 7

Lần thử 2:

Nhập xâu ký tự: Hello There Hieu
Số ký tự nguyên âm là: 7