

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH

TRƯỜNG ĐẠI HỌC BÁCH KHOA



MÔN HỌC: KIẾN TRÚC MÁY TÍNH (THỰC HÀNH) (CO2008)

Bài tập/Thực hành 2

CHƯƠNG 2 KIẾN TRÚC TẬP LỆNH MIPS: Lệnh đại số, luận lý, truy xuất dữ liệu

LỚP THỰC HÀNH L03 – HỌC KỲ 212

Giảng viên hướng dẫn: Vũ Trọng Thiên

Sinh viên thực hiện

Phạm Duy Quang

Mã số sinh viên

2011899

Thành phố Hồ Chí Minh, tháng 03 năm 2022

Bài tập và Thực hành

Bài 1: Syscall.

Tham khảo manual của lệnh syscall trong phần help của công cụ MARS và hiện thực các yêu cầu dưới đây dùng lệnh syscall.

(a) Viết chương trình **nhập vào 3 số nguyên a, b, c rồi xuất ra màn hình giá trị của hàm $f(a,b,c) = (a - b) + c$.**

❖ Code:

```
.data
str1: .asciiz "a = "
str2: .asciiz "b = "
str3: .asciiz "c = "
str4: .asciiz "f(a,b,c) = (a - b) + c = "
.text
    la $a0, str1
    li $v0, 4
    syscall
    li $v0, 5
    syscall
    move $s1, $v0

    la $a0, str2
    li $v0, 4
    syscall
    li $v0, 5
    syscall
    move $s2, $v0

    la $a0, str3
    li $v0, 4
    syscall
    li $v0, 5
```

```

syscall

move $s3, $v0

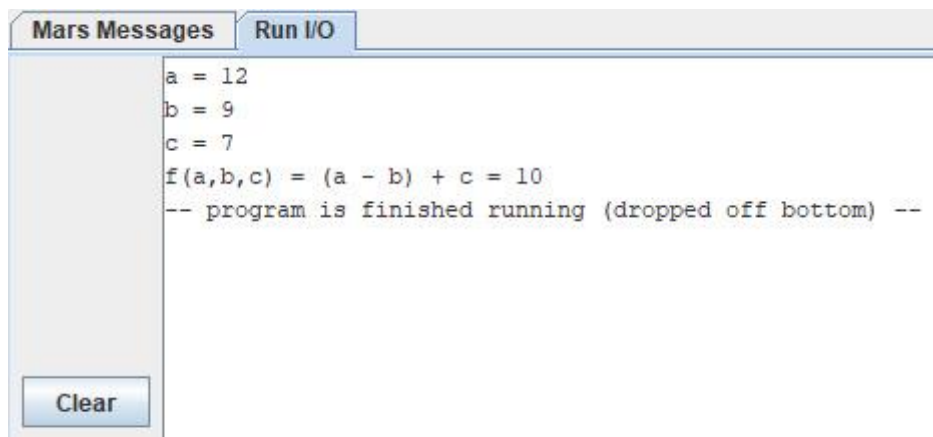
sub $t0, $s1, $s2
add $s0, $t0, $s3

la $a0, str4
li $v0, 4
syscall

li $v0, 1
move $a0, $s0
syscall

```

❖ **Run I/O:**



(b) Viết chương trình xuất ra chuỗi **"Kien Truc May Tinh 2020"**. (giống ví dụ HelloWorld!)

❖ **Code:**

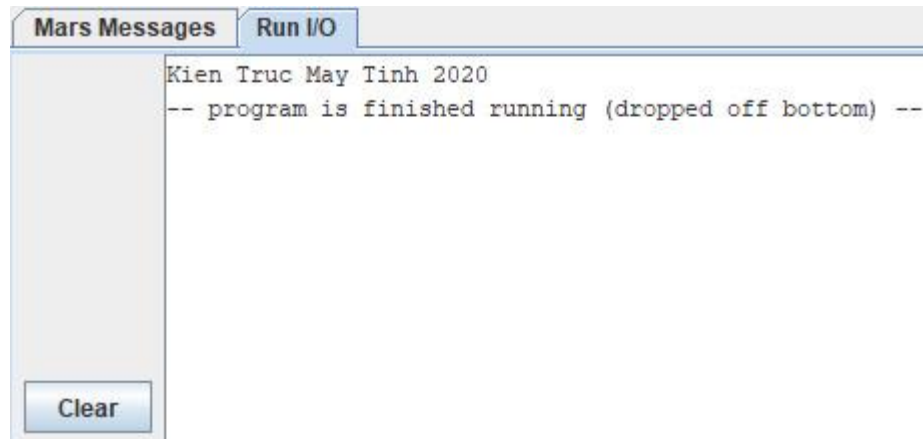
```

.data
str: .asciiz "Kien Truc May Tinh 2020"

.text
li $v0, 4
la $a0, str
syscall

```

❖ **Run I/O:**



(c) Viết chương trình đọc vào một chuỗi 10 ký tự sau đó xuất ra màn hình chuỗi ký tự đó.

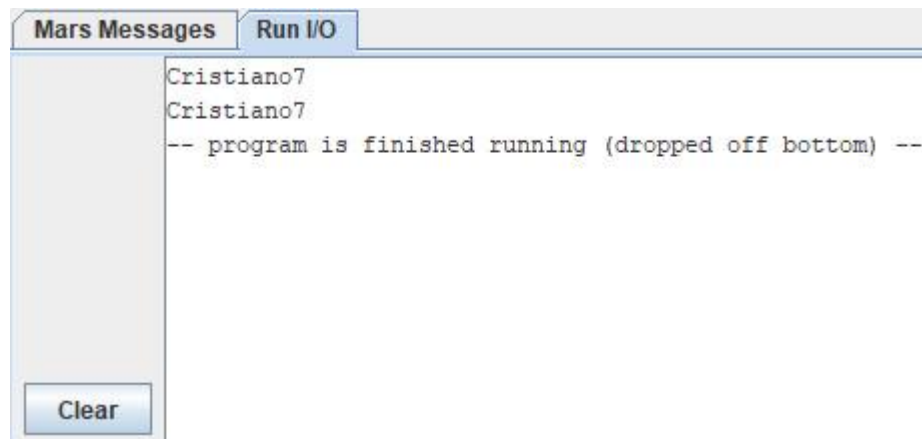
❖ **Code:**

```
.data
    str: .space 11
    newline: .asciiz "\n"
.text
    li $v0, 8
    la $a0, str
    li $a1, 11
    syscall

    la $a0, newline
    addi $v0, $0, 4
    syscall

    li $v0, 4
    la $a0, str
    syscall
```

❖ **Run I/O:**



Bài 2: Các lệnh số học luận lý.

Viết chương trình dùng các lệnh add, addi, sub, subi, or, ori . . . để thực hiện phép tính bên dưới.

```
100000 # This immediate number is greater than 16-bit
+ 5000
- 400
```

Kết quả chứa vào thanh ghi \$s0 và xuất kết quả ra màn hình (console).

❖ Code:

```
.data
    str: .asciiz "100000 + 5000 - 400 = "
    number1: .word 100000
    number2: .word 5000
    number3: .word 400
.text
    lw $s1, number1
    lw $s2, number2
    lw $s3, number3

    add $t1, $s1, $s2
    sub $s0, $t1, $s3

    la $a0, str
    li $v0, 4
```

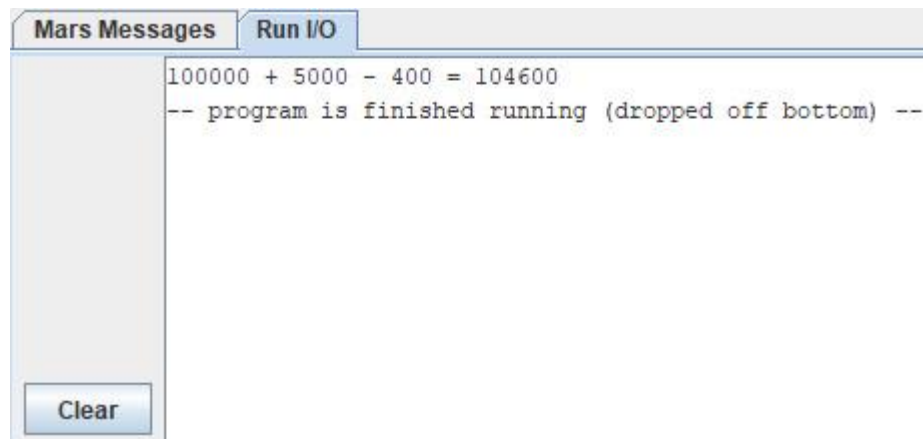
```
syscall
```

```
li $v0, 1
```

```
move $a0, $s0
```

```
syscall
```

❖ **Run I/O:**



Bài 3: Các lệnh về số học, phép nhân.

Viết chương trình tính giá trị biểu thức $f(x)$ bên dưới. Kết quả lưu vào thanh ghi s_0 và xuất ra màn hình.

$$f = a.x^3 - b.x^2 + c.x - d$$

Dùng syscall để nhập a, b, c, d, x và xuất kết quả ra màn hình.

Gợi ý: (theo phương pháp Horner's Method, sinh viên có thể làm theo cách của riêng mình)

- Nhân a với x rồi lưu kết quả vào thanh ghi tạm. $t = a.x$
- Thực hiện phép số tính giữa thanh ghi tạm với b . $t = t - b$ // $t = a.x - b$
- Nhân thanh ghi tạm với x . $t = t.x$ // $t = (a.x - b)x$
- Thực hiện phép số tính giữa thanh ghi tạm với c . $t = t + c$ // $t = a.x^2 - b.x + c$
- Nhân thanh ghi tạm với x . $t = t.x$ // $t = (a.x^2 - b.x + c)x$
- Thực hiện phép số tính giữa thanh ghi tạm với d . $t = t - d$ // $t = a.x^3 - b.x^2 + c.x - d$

❖ **Code:**

```

.data
str1: .asciiz "a = "
str2: .asciiz "b = "
str3: .asciiz "c = "
str4: .asciiz "d = "
str5: .asciiz "x = "
str6: .asciiz "f = a.x^3 - b.x^2 + c.x - d = "
.text

    la $a0, str1
    li $v0, 4
    syscall
    li $v0, 5
    syscall
    move $s1, $v0

    la $a0, str2
    li $v0, 4
    syscall
    li $v0, 5
    syscall
    move $s2, $v0

    la $a0, str3
    li $v0, 4
    syscall
    li $v0, 5
    syscall
    move $s3, $v0

    la $a0, str4
    li $v0, 4
    syscall

```

```

li $v0, 5
syscall
move $s4, $v0

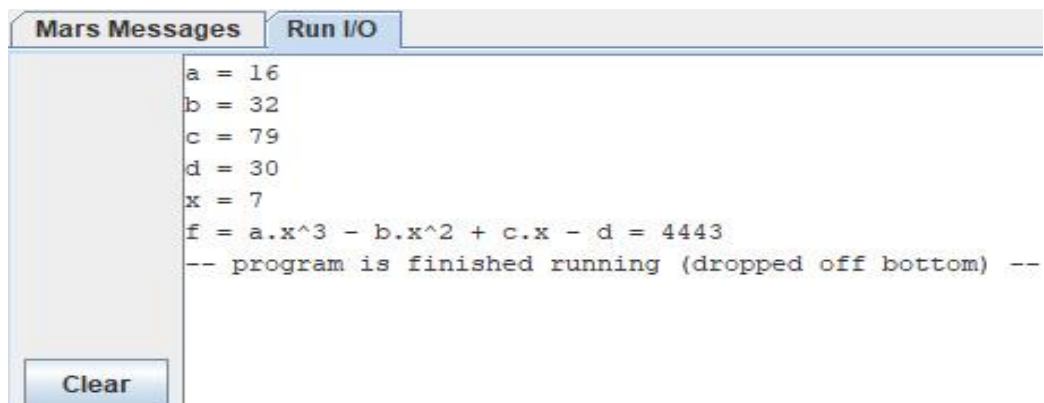
la $a0, str5
li $v0, 4
syscall
li $v0, 5
syscall
move $s5, $v0

mulo $t0, $s1, $s5
sub $t0, $t0, $s2
mulo $t0, $t0, $s5
add $t0, $t0, $s3
mulo $t0, $t0, $s5
sub $s0, $t0, $s4

la $a0, str6
li $v0, 4
syscall
li $v0, 1
move $a0, $s0
syscall

```

❖ Run I/O:



The screenshot shows a window titled "Mars Messages" with a tab labeled "Run I/O". The output text is as follows:

```

a = 16
b = 32
c = 79
d = 30
x = 7
f = a.x^3 - b.x^2 + c.x - d = 4443
-- program is finished running (dropped off bottom) --

```

At the bottom left of the window is a button labeled "Clear".

Bài 4: Lệnh load/store.

(a) Cho dãy **số nguyên** 10 phần tử, xuất ra kết quả là HIỆU của phần tử thứ 2 và 5. Mảng bắt đầu từ phần tử thứ 0.

❖ Code:

```
.data
x:          .word 7,1,10,4,6,77,8,5,2,43
iterator:   .word 0
size:       .word 9

.text
main:
    la $t0, x
    lw $t1, iterator
    lw $t2, size

begin_loop1:
    bgt $t1, 2, exit_loop1

    sll $t3, $t1, 2

    addu $t3, $t3, $t0

    addi $t1, $t1, 1

    j begin_loop1
exit_loop1:
    lw $t4, 0($t3)

begin_loop2:
    bgt $t1, 5, exit_loop2
```

```

sll $t3, $t1, 2

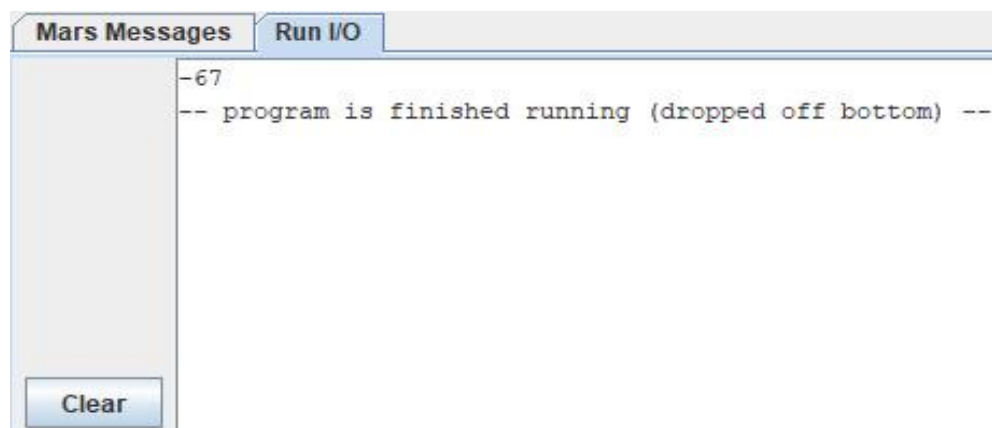
addu $t3, $t3, $t0

addi $t1, $t1, 1

j begin_loop2
exit_loop2:
lw $t5, 0($t3)
sub $s0, $t4, $t5
li $v0, 1
la $a0, ($s0)
syscall

```

❖ **Run I/O:**



(b) Chuyển đổi vị trí cuối và đầu của chuỗi "MSSV - Ho-Ten". Ví dụ chuỗi "123456 - Nguyen Van A" sẽ chuyển thành "A23456 - Nguyen Van 1". **Sinh viên thay tên và mã số sinh viên của mình vào chuỗi trên.**

❖ **Code:**

```

.data
string: .asciiz "2011899 - Pham Duy Quang"
.text
main:
    la $a0, string

```

```
lb $t0, ($a0)
lb $t1, 23($a0)
sb $t0, 23($a0)
sb $t1, ($a0)

la $a0, string
li $v0, 4
syscall

li $v0, 10
syscall
```

❖ **Run I/O:**

