



# CO2008 - KIẾN TRÚC MÁY TÍNH

Khoa Khoa học và kỹ thuật máy tính  
Đại học Bách Khoa - ĐHQG Tp.HCM

09/2024

## Bài thực hành 2

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

#### Mục tiêu

- Sử dụng thành thạo công cụ mô phỏng MARS. Biết cấu trúc một chương trình hợp ngữ MIPS.
- Sử dụng syscall xuất/nhập dữ liệu để hiển thị kết quả.
- Dùng thành thạo các lệnh luận lý, đại số trong tập lệnh MIPS.
- Biết cách định nghĩa biến thuộc các kiểu dữ liệu khác nhau và sử dụng được các lệnh về truy xuất dữ liệu (load/store).

#### Yêu cầu

- Tìm hiểu công cụ MARS, hệ thống syscall và thực hành trên máy cá nhân.
- Hiểu các lệnh trong tập lệnh hợp ngữ MIPS.
- Biết sử dụng công cụ tham khảo nhanh tập lệnh.

#### Bài tập và thực hành

##### Bài 1. Syscall

Tham khảo thẻ syscall trong phần help của công cụ MARS và hiện thực các yêu cầu dưới đây bằng syscall.

- Viết chương trình xuất ra chuỗi "Kien Truc May Tinh 2022.\n" ra màn hình.
- Viết chương trình nhập vào 3 số nguyên a, b, c rồi xuất ra màn hình giá trị của hàm  $f(a,b,c) = a - b + c$
- Viết chương trình đọc vào một chuỗi tối đa 10 ký tự, sau đó xuất chuỗi đã nhập ra màn hình.

##### Bài 2. Các lệnh đại số, 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
+ 1000
- 100
```

Kết quả chứa vào thanh ghi \$t0 và xuất ra màn hình (console).

##### Bài 3. Các lệnh về phép toán nhân.

Viết chương trình nhập a, b, c, d, x, tính giá trị biểu thức  $f(x)$  bên dưới. Kết quả lưu vào thanh ghi \$t0 và xuất ra màn hình.

$$f(x) = a \cdot x^3 + b \cdot x^2 - c \cdot x - d$$

Gợi ý: (phương pháp Horner hạ bậc x) tính theo trình tự từ trong ra ngoài:

$$f(x) = ((a \cdot x + b) \cdot x - c) \cdot x - d$$

- $t = a \cdot x$
- $t = t + b$        $[= a \cdot x + b]$
- $t = t \cdot x$        $[= (a \cdot x + b) \cdot x]$
- $t = t - c$        $[= (a \cdot x + b) \cdot x - c]$
- $t = t \cdot x$        $[= ((a \cdot x + b) \cdot x - c) \cdot x]$
- $t = t - d$        $[= ((a \cdot x + b) \cdot x - c) \cdot x - d]$

**Bài 4.** Lệnh load/store.

- a) Định nghĩa dãy số nguyên 10 phần tử có trị ban đầu. In ra kết quả là HIỆU của phần tử 7 và 3. Mảng bắt đầu từ phần tử 0.
- b) Đổi chỗ ký tự cuối và đầu của chuỗi "MSSV - HoTen". Ví dụ chuỗi "123456 - Nguyen Van A" sẽ chuyển thành "A23456 - Nguyen Van 1". Sinh viên sử dụng tên và mã số sinh viên của chính mình khi viết chương trình.

**Làm thêm**

1. Xác định các trường (OP, Rs, Rt, Rd, shamt, function, immediate) của các lệnh sau và chuyển các lệnh đó qua mã máy (dạng số hex)

```
add $t0,$s0,$a0    # add register to register
addi $v0,$a1,200    # add register to immediate
lw  $t0,4($a0)      # load word
sw  $t0,4($a0)      # store word
lb  $t0,4($a0)      # load byte
sb  $t0,4($a0)      # store byte
sll $t1,$s0,5        # shift left logic (5-bit)
```