

CE119-Lab03/IT012-Lab05

1. Thao tác với mảng

Mảng với n phần tử là một chuỗi n phần tử liên tiếp nhau trong bộ nhớ. Thao tác với mảng trong MIPS là thao tác trực tiếp với byte/word trong bộ nhớ.

- Để cấp phát chuỗi word hoặc byte trong bộ nhớ, có giá trị khởi tạo sử dụng “.word” hoặc “.byte” trong “.data”
- Để cấp phát chuỗi byte không có giá trị khởi tạo trước, sử dụng “.space” trong “.data”

Cho ba mảng với cấp phát dữ liệu trong bộ nhớ như sau:

```
.data
array1: .word 5, 6, 7, 8, 1, 2, 3, 9, 10, 4
size1:  .word 10

array2: .byte 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16
size2:  .word 16

array3: .space 8
size3:  .word 8
```

Mảng array1 có 10 word, kích thước được lưu trong size1; Mảng array2 có 16 byte, kích thước được lưu trong size2; Mảng array3 có 8 byte, kích thước được lưu trong size3.

Viết code trong phần “.text” thực hiện riêng từng phần việc:

- ✓ In ra cửa sổ I/O của MARS tất cả các phần tử của mảng array1 và array2
- ✓ Gán các giá trị cho mảng array3 sao cho
$$array3[i] = array2[i] + array2[size2 - 1 - i]$$
- ✓ Người sử dụng nhập vào mảng thứ mấy và chỉ số phần tử cần lấy trong mảng đó, chương trình xuất ra phần tử tương ứng.

2. Thao tác với con trỏ

Con trỏ là một biến lưu địa chỉ của một biến khác. Thao tác với con trỏ trong MIPS là thao tác trực tiếp với địa chỉ bộ nhớ.

Thực hiện lại các yêu cầu của nội dung 1 với con trỏ.

3. Bài tập (chỉ sử dụng con trỏ)

a. **Nhập một mảng các số nguyên n phần tử (nhập vào số phần tử và giá trị của từng phần tử), xuất ra cửa sổ I/O của MARS theo từng yêu cầu sau:**

- ✓ Xuất ra giá trị lớn nhất và nhỏ nhất của mảng
- ✓ Tổng tất cả các phần tử của mảng
- ✓ Người sử dụng nhập vào chỉ số của một phần tử nào đó và giá trị của phần tử đó được in ra cửa sổ

b. **Nhập một mảng các số nguyên n phần tử (nhập vào số phần tử và giá trị của từng phần tử). Mảng này gọi là A.**

Chuyển dòng lệnh C dưới đây sang mã assembly của MIPS. Với các biến nguyên i, j được gán lần lượt vào thanh ghi $\$s0, \$s1$; và địa chỉ nền của mảng số nguyên A được lưu trong thanh ghi $\$s3$

$$\begin{aligned} & \text{if } (i < j) \ A[i] = i; \\ & \text{else } A[i] = j; \end{aligned}$$

c. **Nhập vào hai mảng số nguyên A và B chứa lần lượt n_A và n_B phần tử (nhập vào số phần tử và giá trị của từng phần tử cho từng mảng).**

Chuyển dòng lệnh C dưới đây sang mã Assembly của vi xử lý MIPS, với A, B là các mảng số nguyên, i là biến nguyên. Giả sử biến i được gán vào thanh ghi $\$s0$, địa chỉ nền của mảng A và B lần lượt được lưu trong thanh ghi $\$s1$ và $\$s2$.

$$\begin{aligned} & \text{for } (i = 2; i < 10; i++) \\ & \ A[i] = B[A[i-2]]; \end{aligned}$$

-----Hết-----