

KIẾN TRÚC MÁY TÍNH

WEEK 5

NGUYỄN HẢI DƯƠNG - 20194530

Assignment 1:

- Code:

```
1 #Laboratory Exercise 5, Assignment 1
2 .data
3 test: .asciiz "Nguyen Hai Duong - 20194530"
4 .text
5 li $v0, 4      # Nạp 4 vào $v0 để chỉ ra một chuỗi in.
6 la $a0, test   # Lưu địa chỉ của "Hello World" vào $a0.
7 syscall        # In ra message
8
```

- Chạy chương trình:

The screenshot displays the Mips simulator interface. The 'Text Segment' window shows the assembly code with labels 5, 6, and 7. The 'Labels' window shows the label 'test' at address 0x10010000. The 'Data Segment' window shows the memory layout with the string 'y u g N H n e D i a g n o u 2 - 4 9 1 0' stored at address 0x10010000. The 'Data Segment' window also shows the memory layout for the string 'Nguyen Hai Duong - 20194530' starting at address 0x10010004.

- Debug từng dòng:

Step	\$pc	Giá trị thanh ghi thay đổi
1	0x00400004	\$v0 = 0x00000004
2	0x00400008	\$at = 0x10010000
3	0x0040000c	\$a0 = 0x10010000

Theo Home Assignment 1:

Thanh ghi \$v0 được nạp giá trị 4, \$a0 được nạp địa chỉ của chuỗi kí tự lưu ở biến test. Với lệnh gọi hệ thống syscall thì chương trình nhận được giá trị 4 từ \$v0 -> thực hiện lệnh in ra màn hình console giá trị được lưu tại địa chỉ của thanh ghi \$a0

Lưu trữ chuỗi:

Tại địa chỉ 0x10010000 chuỗi “Nguyen Hai Duong - 20194530” được lưu theo cụm kí tự tương ứng với 4 byte địa chỉ, tại mỗi 1 cụm 4 kí tự, các kí tự được lưu theo thứ tự ngược ví dụ cụm “Nguy” thì được lưu thành “yugn”. Khi hết chuỗi, tất cả các địa chỉ ở phía sau đều có giá trị “\0”.

Assignment 2:

- Chương trình

```
1 #Laboratory Exercise 5, Assignment 2
2 .data
3 text1: .ascii "The sum of "
4 text2: .ascii " and"
5 text3: .ascii " is "
6
7 .text
8 li $v0, 4
9 li $s0, 2019
10 li $s1, 4530
11 add $t1, $s0, $s1
12 la $a0, text1
13 syscall
14 li $v0, 1
15 la $a0, 0($s0)
16 syscall
17 li $v0, 4
18 la $a0, text2
19 syscall
20 li $v0, 1
21 la $a0, 0($s1)
22 syscall
23 li $v0, 4
24 la $a0, text3
25 syscall
26 li $v0, 1
27 la $a0, 0($t1)
28 syscall
```

⇒ Kết quả:

```
The sum of 2019 and 4530 is 6549
-- program is finished running (dropped off bottom) --
```

- Debug từng dòng:

Step	\$pc	Giá trị thanh ghi thay đổi
1	0x00400004	\$v0 = 0x00000004
2	0x00400008	\$s0 = 0x000007e3
3	0x0040000c	\$s1 = 0x000011b2
4	0x00400010	\$t1 = 0x00001195
5	0x00400014	\$at = 0x10010000
6	0x00400018	\$a0 = 0x10010000
7	0x00400020	\$v0 = 0x00000001
8	0x00400024	\$a0 = 0x000007e3
9	0x00400030	\$v0 = 0x00000004
10	0x00400034	\$at = 0x10010000
11	0x00400038	\$a0 = 0x1001000c
12	0x00400040	\$v0 = 0x00000001
13	0x00400044	\$at = 0x00000000
14	0x00400048	\$a0 = 0x000011b2
15	0x00400050	\$v0 = 0x00000004
16	0x00400054	\$at = 0x10010000
17	0x00400058	\$a0 = 0x10010012
18	0x00400060	\$v0 = 0x00000001
19	0x00400064	\$at = 0x00000000
20	0x00400068	\$a0 = 0x00001995

- tính toán trước tổng của hai số và lưu vào thanh ghi \$t1
- in ra từng đoạn của cụm câu “The sum of (s0) and (s1) is (result)”
- In các chuỗi liên mạch và in giá trị số nguyên tiếp theo đó

Assignment 3:

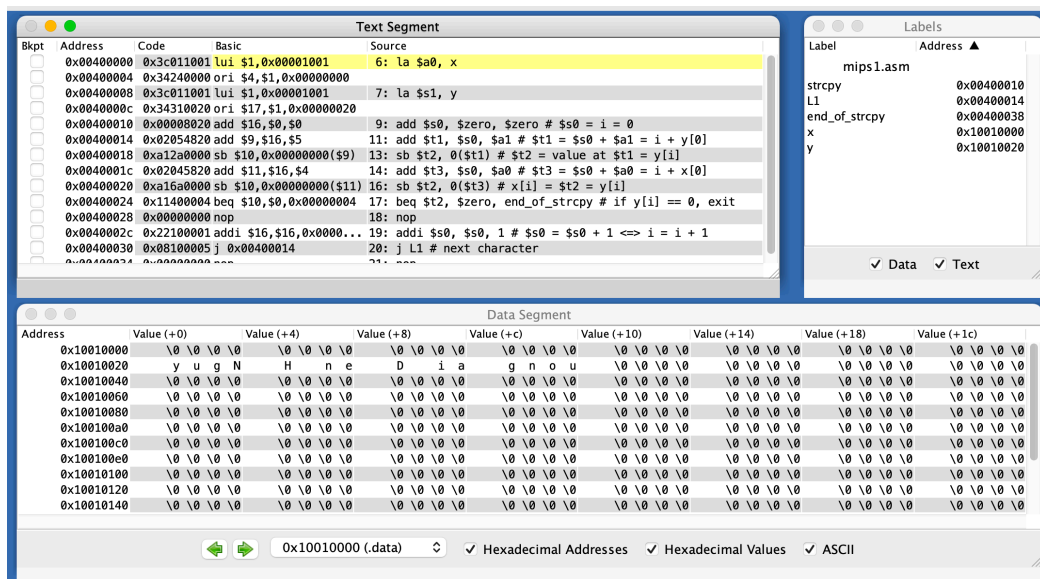
- Code

```

1  #Laboratory Exercise 5, Assignment 3
2  .data
3  x: .space 32 # destination string x, empty
4  y: .asciiz "Nguyen Hai Duong" # source string y
5  .text
6  la $a0, x
7  la $s1, y
8  strcpy:
9  add $s0, $zero, $zero # $s0 = i = 0
10 L1:
11 add $t1, $s0, $a1 # $t1 = $s0 + $a1 = i + y[0]
12 # = address of y[i]
13 sb $t2, 0($t1) # $t2 = value at $t1 = y[i]
14 add $t3, $s0, $a0 # $t3 = $s0 + $a0 = i + x[0]
15 # = address of x[i]
16 sb $t2, 0($t3) # x[i] = $t2 = y[i]
17 beq $t2, $zero, end_of_strcpy # if y[i] == 0, exit
18 nop
19 addi $s0, $s0, 1 # $s0 = $s0 + 1 ==> i = i + 1
20 j L1 # next character
21 nop
22 end_of_strcpy:

```

- Chương trình:



Name	Number	Value
\$zero	0	0x00000000
\$at	1	0x00000000
\$v0	2	0x00000000
\$v1	3	0x00000000
\$a0	4	0x00000000
\$a1	5	0x00000000
\$a2	6	0x00000000
\$a3	7	0x00000000
\$t0	8	0x00000000
\$t1	9	0x00000000
\$t2	10	0x00000000
\$t3	11	0x00000000
\$t4	12	0x00000000
\$t5	13	0x00000000
\$t6	14	0x00000000
\$t7	15	0x00000000
\$s0	16	0x00000000
\$s1	17	0x00000000
\$s2	18	0x00000000
\$s3	19	0x00000000
\$s4	20	0x00000000
\$s5	21	0x00000000
\$s6	22	0x00000000
\$s7	23	0x00000000
\$s8	24	0x00000000
\$t9	25	0x00000000
\$k0	26	0x00000000
\$k1	27	0x00000000
\$gp	28	0x10000000
\$sp	29	0x7ffffeffc
\$fp	30	0x00000000
\$ra	31	0x00000000
pc		0x00400000
hi		0x00000000
lo		0x00000000

Khi chạy từng vòng lặp ta thấy rõ từng kí tự được copy từ thanh ghi \$a1 sang \$a0 tại mỗi khối 4 byte theo chiều từ phải qua trái

Assignment 4:

- Code:

```

1 #Laboratory Exercise 5, Assignment 4
2 .data
3 string: .space 50
4 Message1: .asciiz "Nhap xau: "
5 Message2: .asciiz "Do dai xau la: "
6 .text
7 main:
8 get_string:
9     li $v0, 54
10    la $a0, Message1
11    la $a1, string
12    li $a2, 22
13    syscall
14
15 get_length:    la $a0, string # $a0 = address(string[0])
16                add $t0, $zero, $zero # $t0 = i = 0
17 check_char:    add $t1, $a0, $t0 # $t1 = $a0 + $t0
18                # = address(string[i])
19                lb $t2, 0($t1) # $t2 = string[i]
20                beq $t2, $zero, end_of_str # is null char?
21                addi $t0, $t0, 1 # $t0 = $t0 + 1 -> i = i + 1
22                j check_char
23
24 end_of_str:
25 end_of_get_length:
26 print_length:
27     li $v0, 56
28     la $a0, Message2
29     add $a1, $t0, -1
30     syscall

```

- Chương trình:

Assignment 5:

```
1 #Laboratory Exercise 5, Assignment 5
2 .data
3 Message: .asciiz "Nhap tung ki tu"
4 string: .space 20
5 .text
6 init:
7     add $s0, $zero, $zero      # i = 0
8     la $s1, string             # dua dia chi string vao $s1
9 read_str:
10 read_char:
11     li $v0, 12                 # doc ky tu nhap tu ban phim
12     syscall
13     nop
14 check_char:
15     beq $v0, 10, print_revert_str # neu ben enter LF = 10 -> Kiem tra
16     add $t1, $s0, $s1          # lay dia chi cua string[i]
17     sb $v0, 0($t1)             # luu $v0 vao string[i]
18     addi $s0, $s0, 1           # i += 1
19     slti $t0, $s0, 20          # i < 20? 1 : 0
20     beq $t0, $zero, print_revert_str # neu khong thi ket thuc viec doc
21     j read_char
22 print_revert_str:
23     slt $t0, $s0, $zero        # neu i < 0 ? 1 : 0
24     bne $t0, $zero, exit       # if true -> exit
25     add $t1, $s0, $s1          # lay dia chi string[i]
26     lb $t2, 0($t1)             # load vao $t2 string[i]
27     li $v0, 11                 # in chuoi
28     add $a0, $zero, $t2        # $a0 = string[i]
29     syscall
30     addi $s0, $s0, -1          # s0 -= 1
31     j print_revert_str
32 exit:
```

⇒ **Kết quả**

aabbccdd

ddccbbaa

⇒ -- program is finished running (dropped off bottom) --