

Báo cáo tuần 3

Thực hành kiến trúc máy tính

Họ tên: Phan Minh Anh Tuấn
MSSV: 20205227

Mục lục

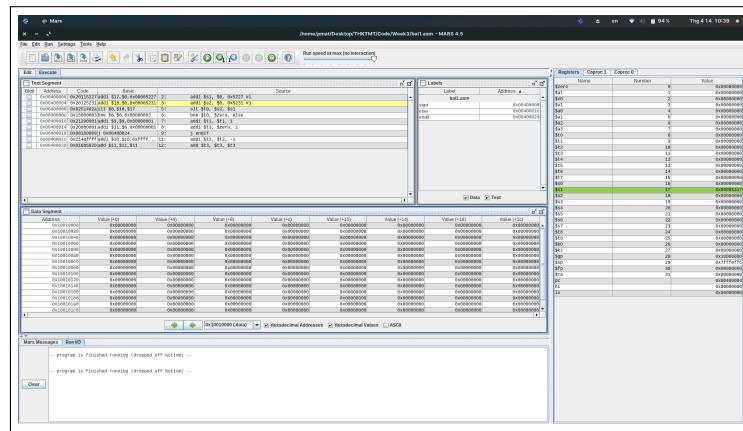
1 Assignment 1	2
2 Assignment 2	5
3 Assignment 3	8
4 Assignment 4	12
4.1 Assignment 4a	12
4.2 Assignment 4b	13
4.3 Assignment 4c	14
4.4 Assignment 4d	15
5 Assignment 5	16
5.1 Assignment 5a	16
5.2 Assignment 5b	17
5.3 Assignment 5c	18
5.4 Assignment 5d	19
6 Assignment 6	20

1 Assignment 1

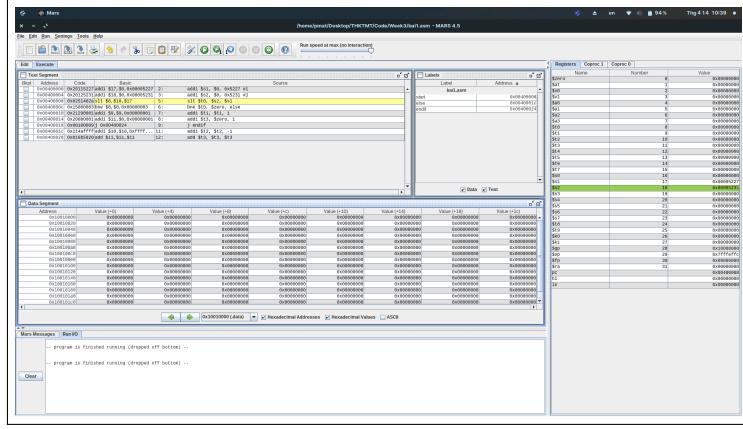
Mã số sinh viên của em: 20205227, mã số sinh viên bạn ngồi cạnh: 20205231
Đặt i = 5227, j = 5231

```
.text
    addi $s1, $0, 0x5227 #i
    addi $s2, $0, 0x5231 #j
start:
    slt $t0, $s2, $s1
    bne $t0, $zero, else
    addi $t1, $t1, 1
    addi $t3, $zero, 1
    j endif
else:
    addi $t2, $t2, -1
    add $t3, $t3, $t3
endif:
```

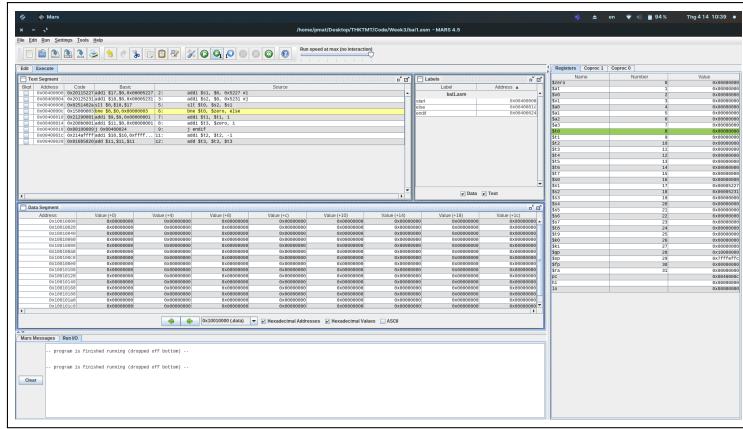
Hình 1: Code của Assignment 1



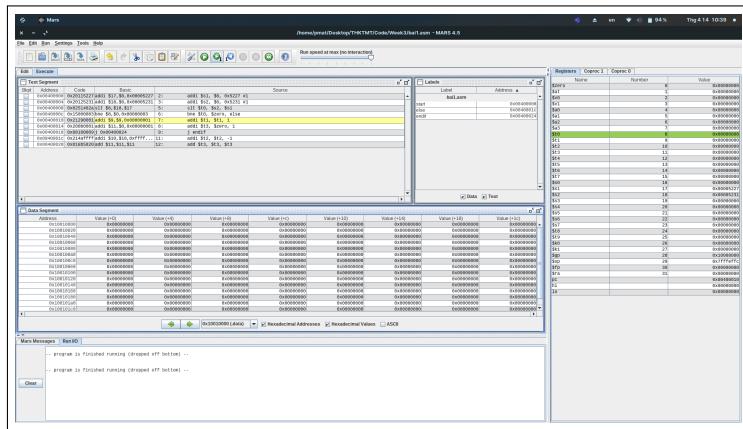
Step 1: Gán thanh ghi \$s1 (i) giá trị 5227



Step 2: Gán thanh ghi $\$s2$ (j) giá trị 5231



Step 3: So sánh j và j, $\$t0 = 0$ do $\$s2 > \$s1$



Step 4: So sánh $\$t0$ với 0, nếu $\$t0 \neq 0$ nhảy đến else

The screenshot shows the MARS 4.5 assembly editor interface. The assembly window displays the following code:

```

    .32bit
    .code32
    .org $00000000
    .data
    .text
    .intel_syntax noprefix
    .global _start
_start:
    movl $1, %eax
    movl %eax, %t1
    movl %t1, %t2
    movl %t2, %t3
    addl $1, %t1
    addl $1, %t2
    addl $1, %t3
    jneq %t1, %t3, .loop
    .loop:
        addl $1, %t1
        addl $1, %t2
        addl $1, %t3
        jneq %t1, %t3, .loop
    .exit:
        movl $0, %eax
        movl %eax, %t1
        movl %t1, %t2
        movl %t2, %t3
        addl $1, %t1
        addl $1, %t2
        addl $1, %t3
        jneq %t1, %t3, .loop
        .end:
        .exit

```

The registers window shows the following values:

Name	Value
\$t0	0
\$t1	1
\$t2	1
\$t3	1
\$t4	0
\$t5	0
\$t6	0
\$t7	0
\$t8	0
\$t9	0
\$t10	0
\$t11	0
\$t12	0
\$t13	0
\$t14	0
\$t15	0
\$t16	0
\$t17	0
\$t18	0
\$t19	0
\$t20	0
\$t21	0
\$t22	0
\$t23	0
\$t24	0
\$t25	0
\$t26	0
\$t27	0
\$t28	0
\$t29	0
\$t30	0
\$t31	0

Step 5: Do $\$t0 == 0$, thực hiện câu lệnh dưới: $\$t1 = \$t1 + 1 = 0x1$

The screenshot shows the MARS 4.5 assembly editor interface. The assembly window displays the same code as before. The registers window shows the following values:

Name	Value
\$t0	0
\$t1	2
\$t2	1
\$t3	1
\$t4	0
\$t5	0
\$t6	0
\$t7	0
\$t8	0
\$t9	0
\$t10	0
\$t11	0
\$t12	0
\$t13	0
\$t14	0
\$t15	0
\$t16	0
\$t17	0
\$t18	0
\$t19	0
\$t20	0
\$t21	0
\$t22	0
\$t23	0
\$t24	0
\$t25	0
\$t26	0
\$t27	0
\$t28	0
\$t29	0
\$t30	0
\$t31	0

Step 6: $\$t3 = \$0 + 1 = 0x1$

The screenshot shows the MARS 4.5 assembly editor interface. The assembly window displays the same code as before. The registers window shows the following values:

Name	Value
\$t0	0
\$t1	2
\$t2	2
\$t3	2
\$t4	0
\$t5	0
\$t6	0
\$t7	0
\$t8	0
\$t9	0
\$t10	0
\$t11	0
\$t12	0
\$t13	0
\$t14	0
\$t15	0
\$t16	0
\$t17	0
\$t18	0
\$t19	0
\$t20	0
\$t21	0
\$t22	0
\$t23	0
\$t24	0
\$t25	0
\$t26	0
\$t27	0
\$t28	0
\$t29	0
\$t30	0
\$t31	0

Step 7: Nhảy đến endif, kết thúc chương trình

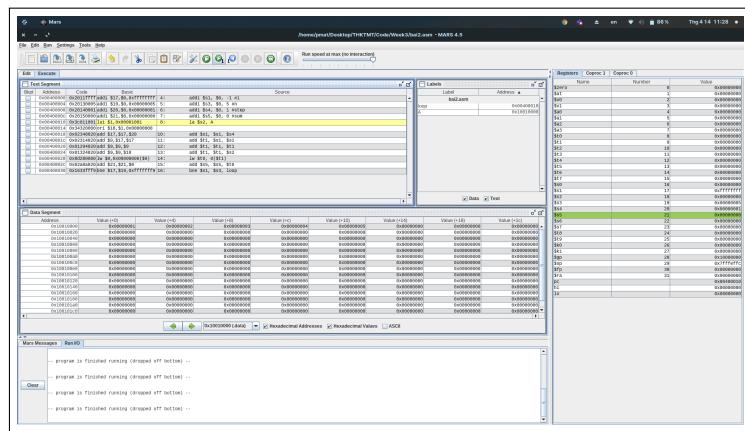
2 Assignment 2

```

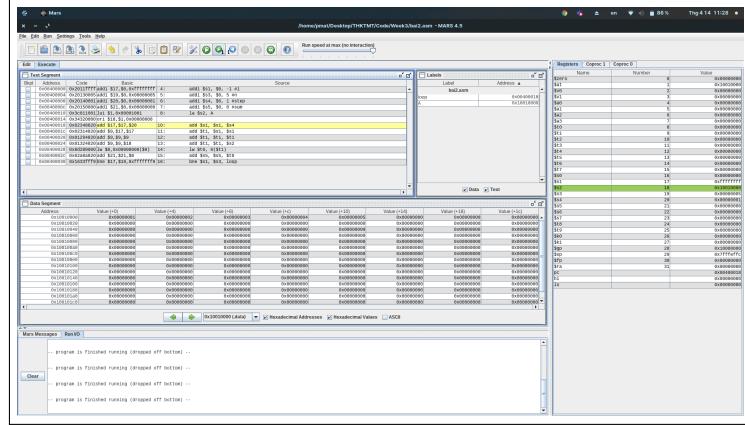
.data
    A: .word 1,2,3,4,5
.text
    addi $s1, $0, -1 #i
    addi $s3, $0, 5 #n
    addi $s4, $0, 1 #step
    addi $s5, $0, 0 #sum
    la $s2, A
loop:
    add $s1, $s1, $s4
    add $t1, $s1, $s1
    add $t1, $t1, $t1
    add $t1, $t1, $s2
    lw $t0, 0($t1)
    add $s5, $s5, $t0
    bne $s1, $s3, loop

```

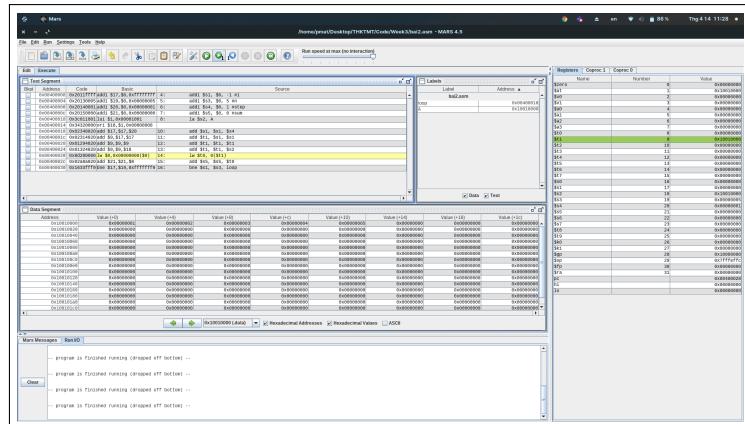
Hình 2: Code của Assignment 2



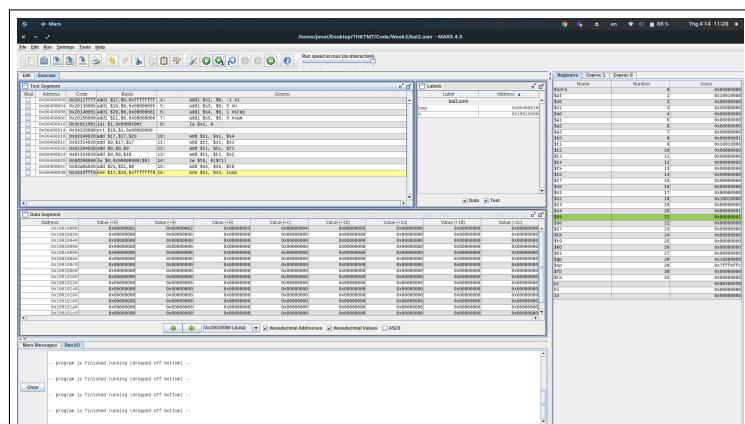
Step 1: Gán giá trị thanh ghi lần lượt \$s1 = -1, \$s3 = 5, \$s4 = 1, \$s5 = 0



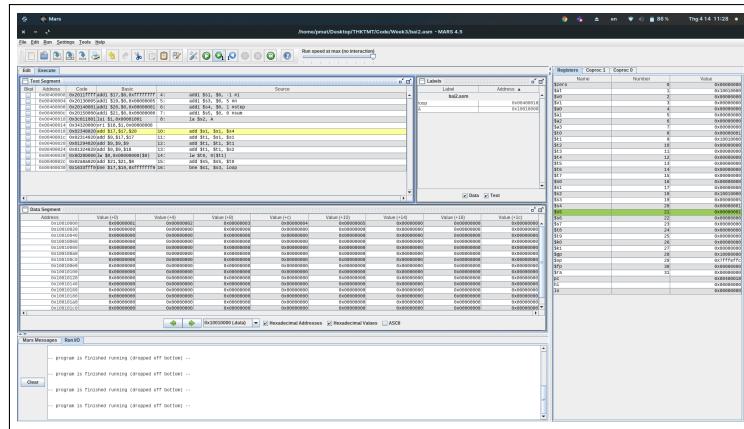
Step 2: Load địa chỉ mảng A vào \$s2



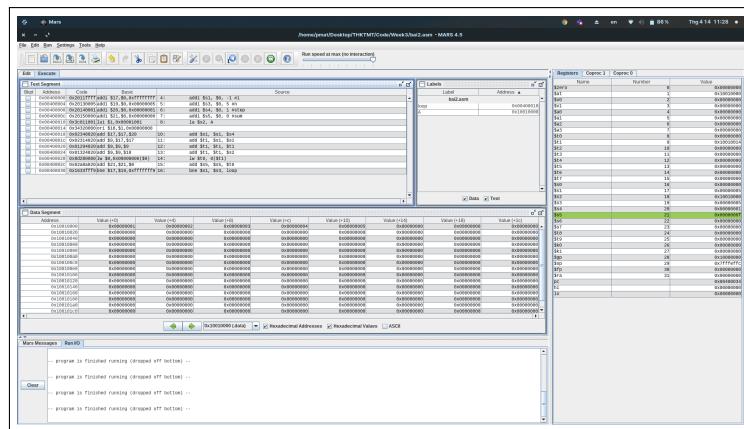
Step 3: \$t1 lưu trữ địa chỉ trả đến phần tử hiện tại của mảng



Step 4: Cộng vào \$s5 (sum) giá trị hiện thời của mảng



Step 5: $i \neq n$, tiếp tục nhảy đến loop

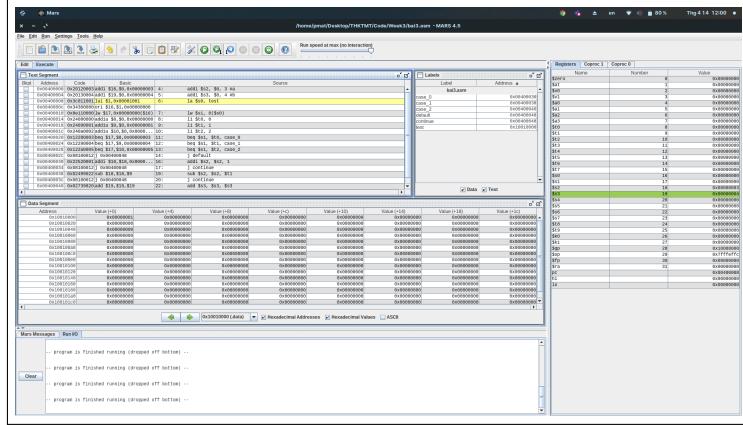


Step 6: qua n vòng lặp, tổng của mảng là $0xf = 15 = 1+2+3+4+5$

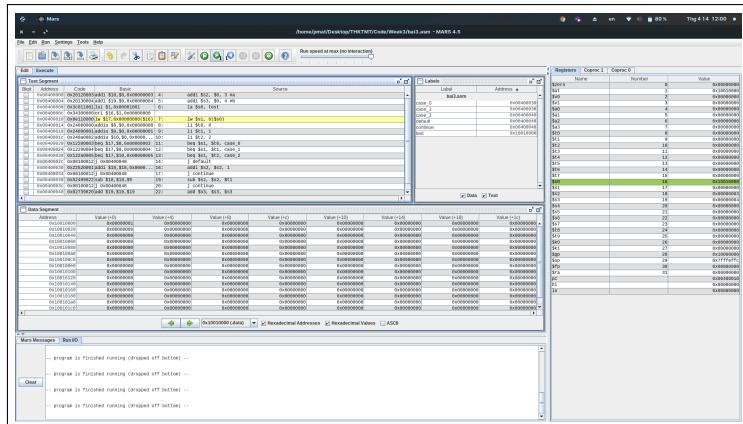
3 Assignment 3

```
.data
test: .word 1
.text
    addi $s2, $0, 3 #a
    addi $s3, $0, 4 #b
    la $s0, test
    lw $s1, 0($s0)
    li $t0, 0
    li $t1, 1
    li $t2, 2
    beq $s1, $t0, case_0
    beq $s1, $t1, case_1
    beq $s1, $t2, case_2
    j default
case_0:
    addi $s2, $s2, 1
    j continue
case_1:
    sub $s2, $s2, $t1
    j continue
case_2:
    add $s3, $s3, $s3
    j continue
default:
continue:
```

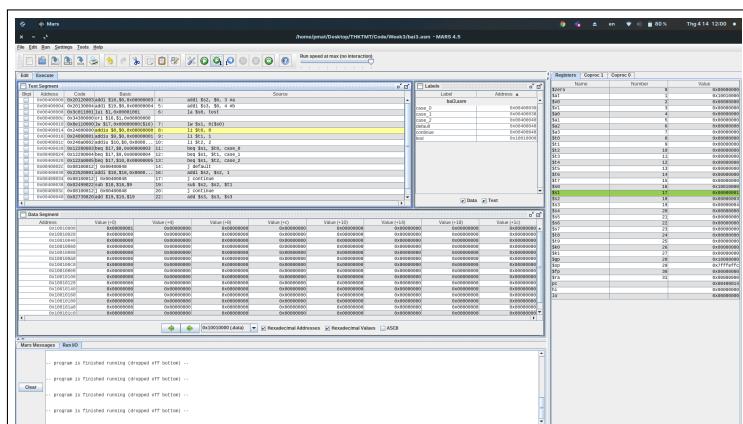
Hình 3: Code của Assignment 3



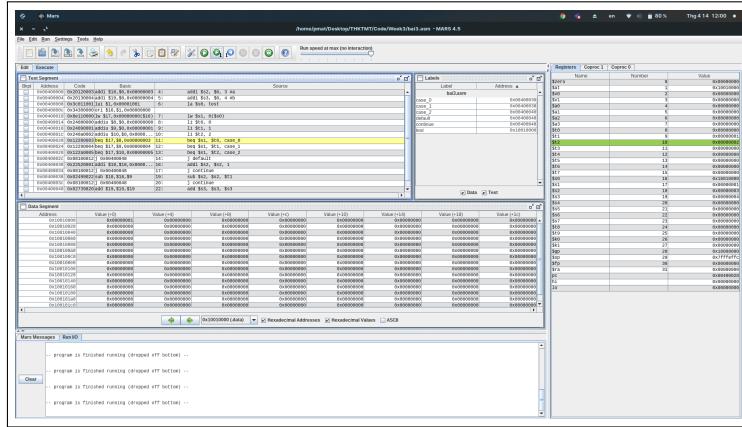
Step 1: Gán giá trị thanh ghi lần lượt \$s2 = 3, \$s3 = 4



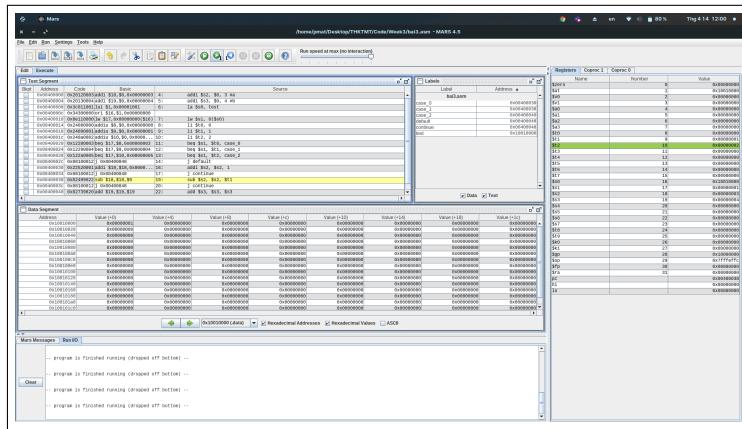
Step 2: Load địa chỉ test vào \$s0



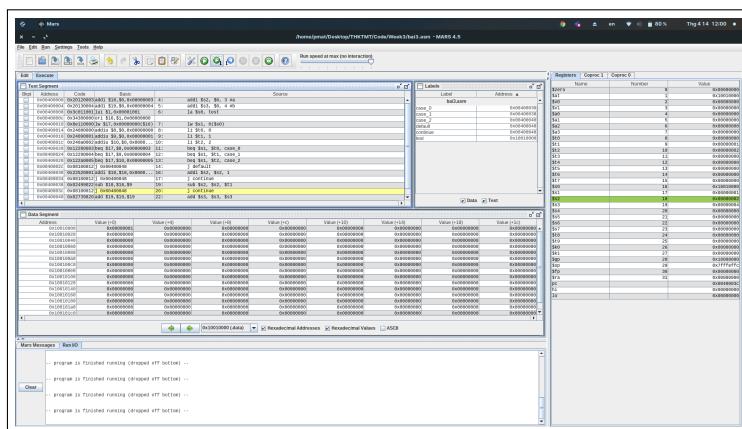
Step 3: Lấy giá trị của \$s1 cho vào \$s0



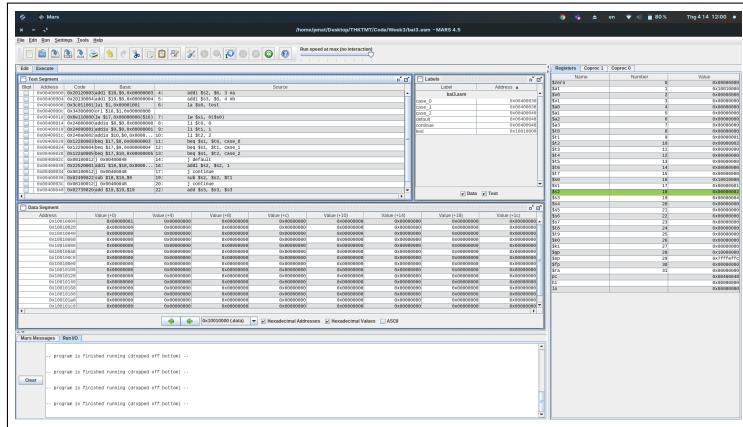
Step 4: Gán tạm thời $\$t0 = 0$, $\$t1 = 1$, $\$t2 = 2$



Step 5: So sánh giá trị $\$s1$ với $\$t0$, $\$t1$, $\$t2$ để nhảy đến case tương ứng



Step 6: do $\$s1 = \$t1 = 1$, nhảy đến case 1, thực hiện $\$s2 = \$s2 - \$t1 = 3 - 1 = 2$



Step 7: Thoát chương trình

Tương tự, khi test = 0, chương trình thực hiện case 0, \$s2 từ 3 tăng lên 4 qua biểu thức addi \$s2, \$s2, 1. Khi test = 2, chương trình thực hiện case 2, \$s3 từ 4 tăng lên 8 qua biểu thức add \$s3, \$s3, \$s3.

4 Assignment 4

4.1 Assignment 4a

```
.text
    addi $s1, $0, 0x5227 #i
    addi $s2, $0, 0x5231 #j
    addi $t4, $0, 1

start:
    slt $t0, $s1, $s2
    bne $t0, $t4, else
    addi $t1, $t1, 1
    addi $t3, $zero, 1
    j endif

else:
    addi $t2, $t2, -1
    add $t3, $t3, $t3

endif:
```

Hình 4: Code của Assignment 4a

Thay đổi dòng 5 và dòng 6, so sánh \$s1 (i) và \$s2(j) để trả về \$t0 sau đó thực hiện so sánh với \$t4 có giá trị bằng 1. Nếu $i < j$, $$t0 = 1$, thực hiện dòng lệnh dưới không nhảy đến else, ngược lại, nhảy đến else và thực hiện lệnh.

4.2 Assignment 4b

```
.text
    addi $s1, $0, 0x5227 #i
    addi $s2, $0, 0x5231 #j
start:
    slt $t0, $s1, $s2
    bne $t0, $zero, else
    addi $t1, $t1, 1
    addi $t3, $zero, 1
    j endif
else:
    addi $t2, $t2, -1
    add $t3, $t3, $t3
endif:
```

Hình 5: Code của Assignment 4b

Đảo chỗ \$s1 \$s2 ở dòng 5 so với assignment 1.

4.3 Assignment 4c

```
.text
    addi $s1, $0, 0x5227 #i
    addi $s2, $0, 0x5231 #j
    add $s3, $s1, $s2 # i + j
start:
    slt $t0, $0, $s3
    bne $t0, $zero, else
    addi $t1, $t1, 1
    addi $t3, $zero, 1|
    j endif
else:
    addi $t2, $t2, -1
    add $t3, $t3, $t3
endif:
```

Hình 6: Code của Assignment 4c

Gán giá trị \$s3 là tổng \$s1 và \$s2, thực hiện so sánh với 0.

4.4 Assignment 4d

```
.text
    addi $s1, $0, 0x5227 #i
    addi $s2, $0, 0x5231 #j
    add $s3, $s1, $s2 # i + j
    li $t1, 5 # m
    li $t2, 6 #n
    add $t3, $t1, $t2
.start:
    slt $t0, $s3, $t3
    bne $t0, $zero, else
    addi $t1, $t1, 1
    addi $t3, $zero, 1
    j endif
.else:
    addi $t2, $t2, -1
    add $t3, $t3, $t3
endif:
```

Hình 7: Code của Assignment 4d

Gán giá trị \$s3 là tổng \$s1 và \$s2. Khởi tạo m là 5, n là 6. Gán giá trị \$t3 là m + n. Sau đó thực hiện so sánh \$s3 và \$t3.

5 Assignment 5

5.1 Assignment 5a

```
.data
    A: .word 1,2,3,4,5
.text
    addi $s1, $0, -1 #i
    addi $s3, $0, 5 #n
    addi $s4, $0, 1 #step
    addi $s5, $0, 0 #sum
    la $s2, A
    addi $t3, 1
loop:
    add $s1, $s1, $s4
    add $t1, $s1, $s1
    add $t1, $t1, $t1
    add $t1, $t1, $s2
    lw $t0, 0($t1)
    add $s5, $s5, $t0
    slt $t2, $s1, $s3
    bne $t2, $0, loop
```

Hình 8: Code của Assignment 5a

So sánh \$s1 (i) và \$s2 (n). Nếu i < n, \$t2 trả về 1, khác \$0 nên tiếp tục vòng lặp. Ngược lại dừng vòng lặp.

5.2 Assignment 5b

```
.data
    A: .word 1,2,3,4,5
.text
    addi $s1, $0, -1 #i
    addi $s3, $0, 5 #n
    addi $s4, $0, 1 #step
    addi $s5, $0, 0 #sum
    la $s2, A
    addi $t3, 1
loop:
    add $s1, $s1, $s4
    add $t1, $s1, $s1
    add $t1, $t1, $t1
    add $t1, $t1, $s2
    lw $t0, 0($t1)
    add $s5, $s5, $t0
    slt $t2, $s3, $s1
    beq $t2, $0, loop
```

Hình 9: Code của Assignment 5b

So sánh \$s1 (i) và \$s2 (n). Nếu n < i, \$t2 trả về 1, ngược lại \$t2 trả về 0. So sánh \$t2 và 0, nếu bằng nhau thực hiện loop, ngược lại dừng vòng lặp

5.3 Assignment 5c

```
.data
    A: .word 1,2,3,4,5
.text
    addi $s1, $0, -1 #i
    addi $s3, $0, 5 #n
    addi $s4, $0, 1 #step
    addi $s5, $0, 0 #sum
    la $s2, A
loop:
    add $s1, $s1, $s4
    add $t1, $s1, $s1
    add $t1, $t1, $t1
    add $t1, $t1, $s2
    lw $t0, 0($t1)
    slt $t2, $s5, $0
    beq $t2, $0, loop
```

Hình 10: Code của Assignment 5c

So sánh 0 và sum. Nếu sum < 0, \$t2 trả về 1 != 0 dừng vòng lặp. Ngược lại tiếp tục vòng lặp.

5.4 Assignment 5d

```
.data
    A: .word 1,2,3,4,5
.text
    addi $s1, $0, -1 #i
    addi $s3, $0, 5 #n
    addi $s4, $0, 1 #step
    addi $s5, $0, 0 #sum
    la $s2, A
loop:
    add $s1, $s1, $s4
    add $t1, $s1, $s1
    add $t1, $t1, $t1
    add $t1, $t1, $s2
    lw $t0, 0($t1)
    bne $t0, $0, loop
```

Hình 11: Code của Assignment 5d

So sánh \$t0 là giá trị trả đến mạng tại ngay lúc đó. Nếu \$t0 khác 0 tiếp tục vòng lặp, ngược lại dừng vòng lặp.

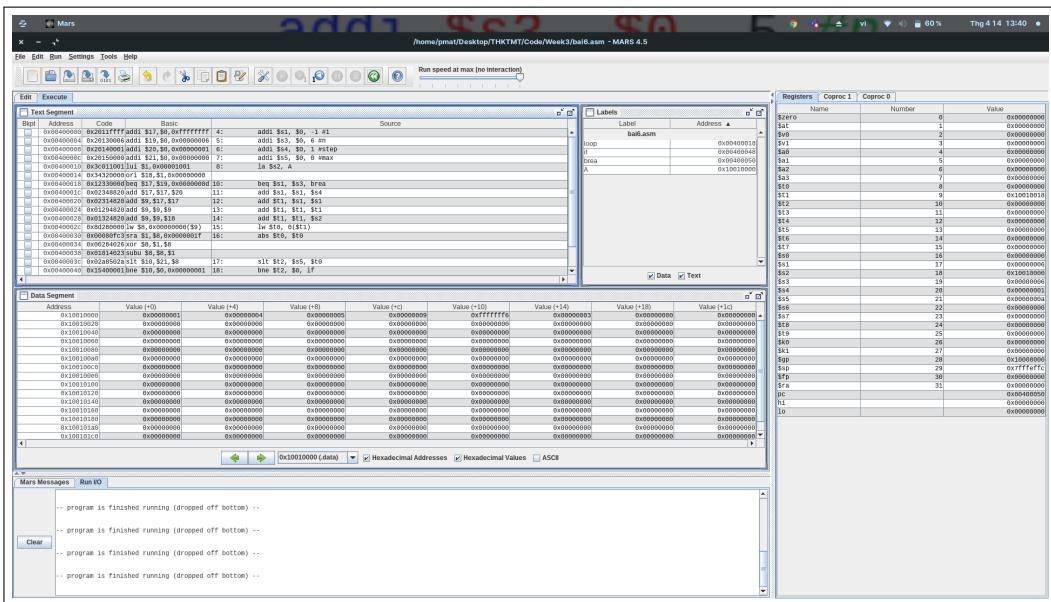
6 Assignment 6

```
.data
    A: .word 1,4,5,9,-10,3
.text
    addi $s1, $0, -1 #i
    addi $s3, $0, 6 #n
    addi $s4, $0, 1 #step
    addi $s5, $0, 0 #max
    la $s2, A
loop:
    beq $s1, $s3, brea
    add $s1, $s1, $s4
    add $t1, $s1, $s1
    add $t1, $t1, $t1
    add $t1, $t1, $s2
    lw $t0, 0($t1)
    abs $t0, $t0
    slt $t2, $s5, $t0
    bne $t2, $0, if
    j loop
if:
    add $s5, $t0, $0
    j loop
brea:
```

Hình 12: Code của Assignment 6

Giải thích thuật toán:

- Tại mỗi vòng lặp, kiểm tra xem $i \neq n$ không, nếu không rời vòng lặp
- Tại dòng 16, lấy trị tuyệt đối của giá trị hiện thời của mảng
- Nếu $\text{tổng } \$s5 < \text{giá trị hiện thời } \$t0$ thì tiến hành gán $\$s5 = \$t0$. Nếu không tiếp tục vòng lặp



Hình 13: Kết quả tại thanh ghi \$s5