# Thực hành kiến trúc máy tính

## Báo cáo thực hành

## Bài 3. Các lệnh nhảy và lệnh rẽ nhánh

Họ Tên	Lê Thành An
MSSV	20235631

#### **ASSIGNMENT 1**

## ĐOẠN MÃ:

```
TH1(else):
```

```
# Laboratory Exercise 3, Home Assignment 1
.text
start:
# TODO:
# Khởi tạo giá trị i vào thanh ghi s1
addi s1, zero, 4
# Khởi tạo giá trị j vào thanh ghi s2
addi s2, zero, 3
# Cách 1:
blt s2, s1, else \# if j < i then jump else
# Cách 2:
# slt t0, s2, s1
\# set t0 = 1 if j < i else clear t0 = 0 t0, zero, else \# t0 != 0 means t0
= 1, jump else
then:
                # then part: x=x+1
addi t1, t1, 1
                    # z=1
addi t3, zero, 1
j endif
                    # skip else part
else:
endif:
```

Lần	s1	s2	t1	t2	t3
1	4	0	0	0	0
2	4	3	0	0	0
3	4	3	0	0	0
4	4	3	0	0xfffffff	0
5	4	3	0	0xfffffff	0+0=0

### TH2(then):

```
# Laboratory Exercise 3, Home Assignment 1
  .text
  start:
  # TODO:
  # Khởi tạo giá trị i vào thanh ghi s1
  addi s1, zero, 4
  # Khởi tạo giá trị j vào thanh ghi s2
  addi s2, zero, 3
  # Cách 1:
  blt s2, s1, then \# if j < i then jump then
  # Cách 2:
          t0, s2, s1
  # slt
  # bne
  \# set t0 = 1 if j < i else clear t0 = 0 t0, zero, else \# t0 != 0 means t0
= 1, jump else
  then:
  addi t1, t1, 1 # then part: x=x+1
  addi t3, zero, 1  # z=1
  j endif
                           # skip else part
  else:
  addi t2, t2, -1
                          # begin else part: y=y-1
  add t3, t3, t3
                           \# z = 2 * z
  endif:
   Lần
                                                     t2
                s1
                            s2
                                        t1
                                                                 t3
   1
                4
                            0
                                        0
                                                     0
                                                                 0
   2
                4
                            3
                                        0
                                                     0
                                                                 0
   3
                            3
                                                     0
                                                                 0
                4
                                        1
   4
                            3
                                                     0
                4
                                        1
                                                                 0
```

1

0

1=0+1

### **ASSIGNMENT 2:**

4

3

TH1:

5

```
CODE:
# Laboratory 3, Home Assigment 2
A: .word 1, 3, 2, 5, 4, 7, 8, 9, 6
.text
# TODO: Khởi tạo giá trị các thanh ghi s2, s3, s4
li s1, 0
\# i = 0
li s5, 0
\# sum = 0
loop:
bge s1, s3, endloop \# if i >= n then end loop
add t1, s1, s1
# t1 = 2 * s1
add t1, t1, t1
\# t1 = 4 * s1 => t1 = 4*i
add t1, t1, s2
# t1 store the address of A[i]
lw t0, 0(t1)
# load value of A[i] in t0
add s5, s5, t0
\# sum = sum + A[i]
add s1, s1, s4
\# i = i + step
j loop
# go to loop
endloop:
```

Sự thay đổi của s1,s5 và t0:

Iteration	s1 (i)	s5 (sum)	t1 (Address of A[i])	t0 (A[i])
Initial	0	0	-	-
1	1	1	0x10010000	1
2	2	4	0x10010004	3
3	3	6	0x10010008	2
4	4	11	0x1001000C	5
5	5	15	0x10010010	4
6	6	22	0x10010014	7
7	7	30	0x10010018	8
8	8	39	0x1001001C	9
9	9	45	0x10010020	6

```
TH2:
```

```
# Laboratory 3, Home Assigment 2
.data
A: .word 10, 3
.text
# TODO: Khởi tạo giá trị các thanh ghi s2, s3, s4
la s2, A
addi s3, zero, 2
addi s3, zero, 1
li s1, 0 #i = 0
li s5, 0 \#sum = 0;
loop:
bge s1, s3, endloop \# if i \ge n then end loop
                # t1 = 2 * s1
add t1, s1, s1
add t1, t1, t1
                  # t1 = 4 * s1 => t1 = 4*i
add t1, t1, s2
                 # t1 store the address of A[i]
1w + 0, 0(t1)
                 # load value of A[i] in t0
add s5, s5, t0
                  \# sum = sum + A[i]
add s1, s1, s4
                 # i = i + step
j loop
                  # go to loop
endloop:
Iteration s1 (i) s5 (sum) t1 (Address of A[i])
                                             t0 (A[i])
Initial
        0
               0
1
        1
               10
                        0x10010000
                                             10
2
        2
               13
                        0x10010004
                                             3
  ⇒ Đúng.
```

#### **ASSIGNMENT 3:**

```
TH1: test = 0:
# Laboratory Exercise 3, Home Assignment 3
.data
test: .word 0
.text
la s0, test
# Nạp địa chỉ của biến test vào s0
lw s1, 0(s0) # Nạp giá trị của biến test vào s1
addi s3, zero, 1
li t0, 0
              # Nạp giá trị cần kiểm tra
li t1, 1
              # Nạp giá trị cần kiếm tra
li t2, 2
              # Nạp giá trị cần kiếm tra
beq s1, t0, case 0
beq s1, t1, case 1
beq s1, t2, case 2
j default
case 0:
addi s2, s2, 1 \# a = a + 1
j continue
case 1:
sub s2, s2, t1 \# a = a - 1
j continue
```

```
case 2:
add s3, s3, s3 \# b = 2 * b
j continue
default:
continue:
TH2: test = 1:
# Laboratory Exercise 3, Home Assignment 3
.data
test: .word 1
.text
la s0, test
              # Nạp địa chỉ của biến test vào s0
lw s1, 0(s0) # Nạp giá trị của biến test vào s1
addi s3, zero, 1
li t0, 0
              # Nạp giá trị cần kiếm tra
li t1, 1
              # Nạp giá trị cần kiểm tra
li t2, 2
            # Nạp giá trị cần kiểm tra
beq s1, t0, case 0
beq s1, t1, case 1
beq s1, t2, case 2
j default
case 0:
addi s2, s2, 1 \# a = a + 1
j continue
case 1:
sub s2, s2, t1 \# a = a - 1
j continue
case 2:
add s3, s3, s3 \# b = 2 * b
j continue
default:
continue:
TH3: test = 2:
# Laboratory Exercise 3, Home Assignment 3
.data
test: .word 2
.text
la s0, test
              # Nạp địa chỉ của biến test vào s0
             # Nạp giá trị của biến test vào s1
lw s1, 0(s0)
addi s3, zero, 1
li t0, 0
               # Nạp giá trị cần kiếm tra
              # Nạp giá trị cần kiểm tra
li t1, 1
li t2, 2
              # Nạp giá trị cần kiểm tra
beq s1, t0, case 0
beq s1, t1, case 1
beq s1, t2, case_2
j default
case 0:
addi s2, s2, 1 \# a = a + 1
j continue
case 1:
sub s2, s2, t1 \# a = a - 1
j continue
case 2:
add s3, s3, s3 \# b = 2 * b
```

j continue
default:
Continue:

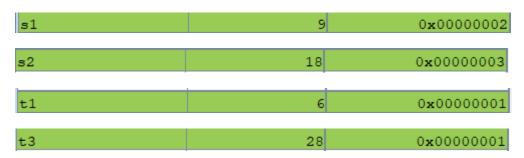
test	s0	s1	s2	s3
0	0x10010000	0x00000000	0x0000001	0x0000001
1	0x10010000	0x0000001	0xfffffff	0x0000001
2	0x10010000	0x00000002	0x00000000	0x00000002

### **ASSIGNMENT 4:**

```
a) i < j :
```

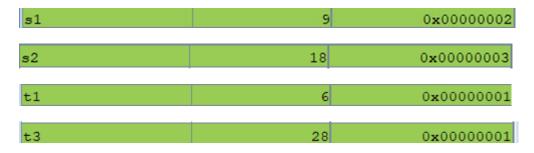
```
# Laboratory Exercise 3, Home Assignment 1
start:
       # TODO:
       addi s2, s2, 3
                             # Khởi tạo giá trị i vào thanh ghi s1
                             # Khởi tạo giá trị j vào thanh ghi s2
       blt s1, s2, then
                             # if i < j then jump then</pre>
       then:
       addi t1, t1, 1
                             # then part:x=x+1
       addi t3, zero, 1
                             # z=1
       j endif
                              # skip else part
       else:
       addi t2, t2, -1
                            # y=y-1
                             # z=2*z
       add t3, t3, t3
       endif:
```

## Giá trị các thanh ghi:



```
b) i >= j :
```

```
# Laboratory Exercise 3, Home Assignment 1
.text
start:
       # TODO:
       addi s1, s1, 2
                             # Khởi tạo giá trị i vào thanh ghi s1
       addi s2, s2, 3
                             # Khởi tạo giá trị j vào thanh ghi s2
       bge s1, s2, else
                             # if i >= j then jump else
then:
       addi t1, t1, 1
                              # then part: x=x+1
       addi t3, zero, 1
                             # z=1
       j endif
                              # skip else part
else:
                             # begin else part: y=y-1
       addi t2, t2, -1
       add t3, t3, t3
                              \# z = 2 * z
endif:
       Giá trị các thanh ghi:
```



### c) i + j <= 0:

```
# Laboratory Exercise 3, Home Assignment 1
.text
start:
       # TODO:
       addi s1, s1, -1
                             # Khởi tạo giá tri i vào thanh ghi s1
       addi s2, s2, -1
                             # Khởi tạo giá trị j vào thanh ghi s2
       add s3, s1, s2
                             # s3 = i + j
       blt zero, s3, else
                             \# if s3 <= 0 (0 >= s3) then jump else
then:
       addi t1, t1, 1
                             # then part: x=x+1
       addi t3, zero, 1
                              \# z=1
       j endif
                              # skip else part
else:
       addi t2, t2, -1
                             # begin else part: y=y-1
       add t3, t3, t3
                             \# z = 2 * z
endif:
```

Giá trị các thanh ghi:

```
s2
                                                       0xffffffff
                                          18
                                         19
                                                       0xfffffffe
     s3
     t1
                                           6
                                                       0x00000001
     t3
                                                       0x00000001
d) i + j > m + n:
# Laboratory Exercise 3, Home Assignment 1
.text
start:
       # TODO:
       addi s1, s1, 2
                            # Khởi tạo giá trị i vào thanh ghi s1
       addi s2, s2, 3
                             # Khởi tạo giá trị j vào thanh ghi s2
```

# s5 = i + j

# s6 = m + n

add s6, s4, s3 bge s6, s5, else

addi s3, s3, 4

addi s4, s4, 0

add s5, s1, s2

# if m + n >= i+j jump else

# Khởi tạo giá trị m vào thanh ghi s3

# Khởi tạo giá trị n vào thanh ghi s4

then:

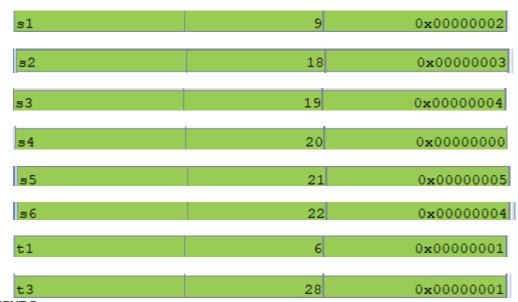
addi t1, t1, 1 # then part: x=x+1 addi t3, zero, 1 # z=1 # skip else part

else:

addi t2, t2, -1 # begin else part: y=y-1 add t3, t3, t3 # z=2\*z

endif:

Giá trị các thanh ghi:



## **ASSIGNMENT 5:**

## a) TH1:i>n:

# Laboratory 3, Home Assigment 2

```
.data
    A: .word 10, 3
.text
# TODO: Khởi tạo giá trị các thanh ghi s2, s3, s4
    la s2, A
    addi s3, zero, 2
    addi s4, zero, 1
    li s1, 0 # i = 0
    li s5, 0 # sum = 0

loop:
    blt s3, s1, endloop # if n < i then loop
    add t1, s1, s1 # t1 = 2 * s1
    add t1, t1, t1 # t1 = 4 * s1 => t1 = 4*i
```

```
add t1, t1, s2 # t1 store the address of A[i]
lw t0, 0(t1) # load value of A[i] in t0
add s5, s5, t0 # sum = sum + A[i]
add s1, s1, s4 # i = i + step
j loop # go to loop
```

endloop:

Giá trị của các thanh ghi:

s2	18	0x10010000
s3	19	0x00000002
<b>s</b> 3	19	0x00000001
s4	20	0x00000001
s1	9	0x00000000
s5	21	0x00000000
t1	6	0x00000000
t1	6	0x10010000
t0	5	0x0000000a
s5	21	0x0000000a
s1	9	0x00000001
t1	6	0x00000002
t1	6	0x00000004
t1	6	0x10010004
t0	5	0x00000003
s5	21	D0000000x0
s1	9	0x00000002

```
TH2 : sum < 0 :
```

```
# Laboratory 3, Home Assignment 2
.data
   A: .word 10, -13, 1
.text
```

# TODO: Khởi tạo giá trị các thanh ghi s2, s3, s4

```
la s2, A
  addi s3, zero, 3
  addi s4, zero, 1
  li s1, 0 # i = 0
  li s5, 0 # sum = 0

loop:

  blt s5, zero, endloop # if sum < 0 then end loop
  add t1, s1, s1 # t1 = 2 * s1
  add t1, t1, t1 # t1 = 4 * s1 => t1 = 4*i
  add t1, t1, s2 # t1 store the address of A[i]
  lw t0, 0(t1) # load value of A[i] in t0
  add s5, s5, t0 # sum = sum + A[i]
  add s1, s1, s4 # i = i + step
  j loop # go to loop
endloop:
```

## Giá trị các thanh ghi:

s2	18	0x10010000
s3	19	0x00000003
s4	20	0x00000001
s1	9	0x00000000
<b>s</b> 5	21	0 <b>x</b> 00000000
t1	6	0x00000000
t0	5	0x0000000a
s5	21	0x0000000a
s1	9	0x00000001
t1	6	0x00000002
t1	6	0x0000004
t0	5	0xfffffff3
s5	21	0xfffffffd
s1	9	0x00000002

```
TH3: A[i] == 0:
# Laboratory 3, Home Assigment 2
.data
  A: .word 10, 0, 1
.text
# TODO: Khởi tạo giá trị các thanh ghi s2, s3, s4
  la s2, A
  addi s3, zero, 3
  addi s4, zero, 1
  li s1, 0 # i = 0
  li s5, 0 \# sum = 0
loop:
  add t1, s1, s1 \# t1 = 2 * s1
  add t1, t1, t1 \# t1 = 4 * s1 => t1 = 4*i
  add t1, t1, s2 # t1 store the address of A[i]
  lw t0, 0(t1) # load value of A[i] in t0
  beq t0, zero, endloop # if A[i] == 0 then end loop
  add s5, s5, t0 \# sum = sum + A[i]
  add s1, s1, s4 # i = i + step
  j loop # go to loop
endloop :
Giá trị các thanh ghi:
s2
                                  18
                                               0x10010000
s3
                                  19
                                               0x00000003
s4
                                  20
                                               0x00000001
s1
                                                0x00000000
s5
                                  21
                                               0x00000000
t1
                                   6
                                               0x00000000
t1
                                               0x10010000
t0
                                   5
                                               0x0000000a
```

21

0x0000000a

**s**5

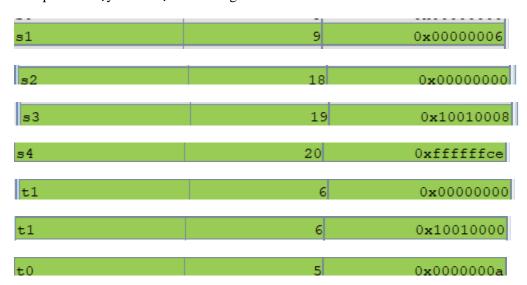
s1	9	0x00000001
t1	6	0x00000002
t1	6	0x00000004
t0	5	0x00000000

### **ASSIGNMENT 6**

## CODE:

```
.data
     A: .word 10, 2, -11, 4, 7, 8
.text
     addi s1, zero, 6
                         \# n = 6
     addi s2, s2, 0
                         \# i = 0
     la s3, A
                          # s3 luu dia chi cua A
     addi s4, s4, -50
                         # gia tri max
loop:
     bge s2, s1, endloop
                         # 1 >= n then endloop
     add t1, s2,s2
     add t1, t1, t1
                          # 4*i
                         # luu dia chi cua phan tu tiep theo
     add t1, t1, s3
     lw t0, 0(t1) # luu gia tri cua phan tu tiep
     blt t0, zero, negative # neu gia tri am
continue:
     blt s4, t0, max
tiep:
     addi s2, s2, 1
     j loop
negative:
     sub t0, zero, t0
     j continue
max:
     add s4, zero, t0
     j tiep
endloop:
```

## Kết quả khi chạy – Giá trị các thanh ghi:



s4	20	0x0000000a
s2	18	0x00000001
t1	6	0x00000002
t1	6	0x00000004
t1	6	0x10010004
t0	5	0x00000002
s2	18	0x00000002
t1	6	0x00000004
t1	6	0x00000008
t0	5	0xfffffff5
t0	5	0x0000000b
s4	20	0x0000000b
s2	18	0x00000003
t1	6	0x00000006
t1	6	0x000000c
t1	6	0x1001000c
t0	5	0x00000004
s2	18	0x00000004
t1	6	0x00000008
t1	6	0x00000010
t0	5	0x00000007
s2	18	0x00000005
t1	6	0x0000000a
t1	6	0x00000014

s2 18 0x00000006