

# BÁO CÁO THỰC HÀNH KIẾN TRÚC MÁY TÍNH

## LAB 4

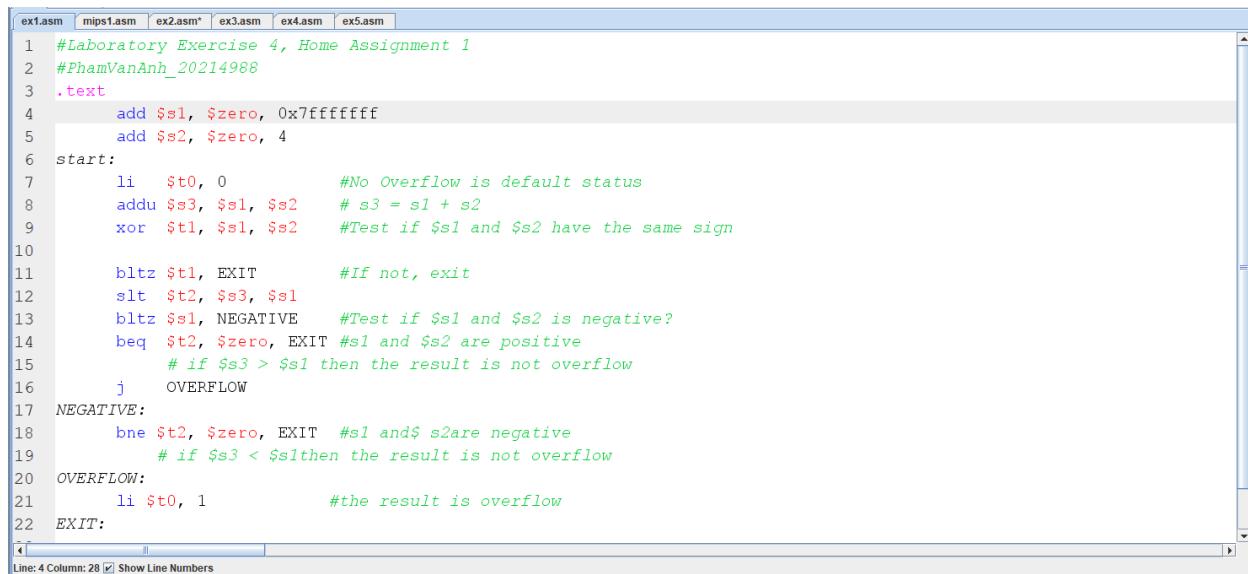
Họ và tên: Phạm Văn Anh

MSSV: 20214988

Mã lớp: 139365

### **ASSIGNMENT 1:**

#### **1. Code**



The screenshot shows a window titled "mips1.asm" containing MIPS assembly code. The code is as follows:

```
1 #Laboratory Exercise 4, Home Assignment 1
2 #PhamVanAnh_20214988
3 .text
4     add $s1, $zero, 0x7fffffff
5     add $s2, $zero, 4
6 start:
7     li $t0, 0          #No Overflow is default status
8     addu $s3, $s1, $s2    # s3 = s1 + s2
9     xor $t1, $s1, $s2    #Test if $s1 and $s2 have the same sign
10
11    bltz $t1, EXIT      #If not, exit
12    slt $t2, $s3, $s1
13    bltz $t2, NEGATIVE   #Test if $s1 and $s2 is negative?
14    beq $t2, $zero, EXIT #s1 and $s2 are positive
15    # if $s3 > $s1 then the result is not overflow
16    j OVERFLOW
17 NEGATIVE:
18    bne $t2, $zero, EXIT  #s1 and $s2 are negative
19    # if $s3 < $s1 then the result is overflow
20 OVERFLOW:
21    li $t0, 1            #the result is overflow
22 EXIT:
```

Line: 4 Column: 28  Show Line Numbers

Thực hiện gõ chương trình vào công cụ MARS

#### **2. Thực hiện dòng lệnh số 4 và 5**

- Gán \$s1 = 0x7fffffff và \$s2 = 4

### 3. Thực hiện dòng lệnh số 8: \$s3 = \$s1 + \$s2.

- Addu là phép cộng nhưng sẽ không báo tràn số

### 4. Thực hiện dòng lệnh số 9: XOR \$s1 và \$s2 xem có cùng dấu hay không?

- \$s1 và \$s2 khác dấu → exit
- \$s1 và \$s2 cùng dấu → chạy tiếp
- Sau khi XOR, lưu vào thanh ghi \$t1

## 5. Thực hiện dòng lệnh số 11: Kiểm tra t1 là âm hoặc dương

- \$t1 âm → exit do cộng 2 số khác dấu
- \$t1 dương → chạy tiếp

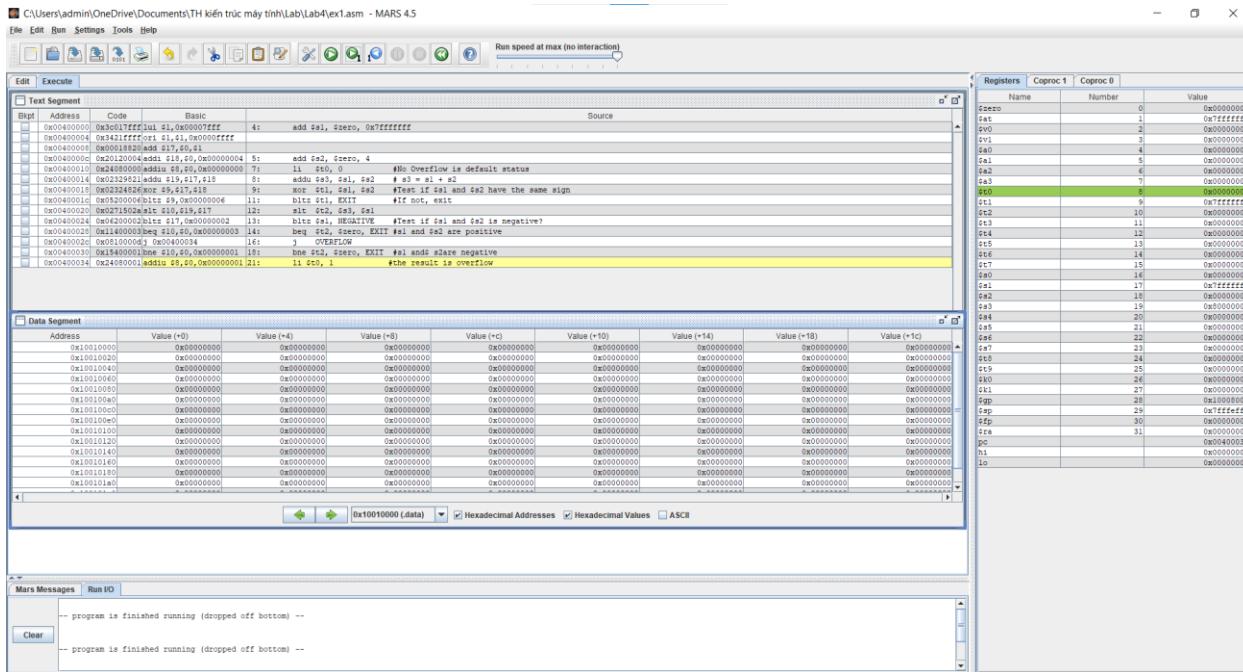
## 6. Thực hiện dòng lệnh số 12: So sánh \$s3 và \$s1

- \$s3 > \$s1 → \$t2 = 0
- \$s3 < \$s1 → \$t2 = 1

The screenshot shows the MARS 4.5 assembly debugger interface. The assembly code for Lab4/ex1.asm is displayed in the main window. The code includes instructions for addition, subtraction, and comparison, along with conditional branches based on the results. The Registers window on the right shows the initial values of registers \$s0 to \$s7 and \$t0 to \$t7. The Data Segment window shows memory starting at address 0x10010000. The Mars Messages window at the bottom indicates the program has finished running.

## 7. Thực hiện dòng lệnh số 13: Kiểm tra \$s1 âm hoặc dương

- \$s1 âm → Nếu \$t2 != 0 → Exit
- \$s1 dương → Kiểm tra \$t2
  - \$t2 = 0 → exit
  - \$t2 != 0 → gán \$t0 = 1



## 8. Trường hợp $\$s1 = -1$ , $\$s2 = 10$

### - Code

```

ex1.asm  mips1.asm  ex2.asm*  ex3.asm  ex4.asm  ex5.asm
1  #Laboratory Exercise 4, Home Assignment 1
2  #PhamVanAnh_20214988
3  .text
4      add $s1, $zero, -1
5      add $s2, $zero, 10
6  start:
7      li $t0, 0      #No Overflow is default status
8      addu $s3, $s1, $s2   # s3 = s1 + s2
9      xor $t1, $s1, $s2   #Test if $s1 and $s2 have the same sign
10
11     bltz $t1, EXIT      #If not, exit
12     slt $t2, $s3, $s1   #Test if $s3 < $s1
13     bltz $t2, NEGATIVE    #Test if $s1 and $s2 is negative?
14     beq $t3, $zero, EXIT   #s1 and $s2 are positive
15     # if $s3 > $s1 then the result is not overflow
16     j OVERFLOW
NEGATIVE:
17     bne $t3, $zero, EXIT   #s1 and $s2 are negative
18     # if $s3 < $s1 then the result is not overflow
19
20  OVERFLOW:
21     li $t0, 1      #the result is overflow
22  EXIT:
23
24

```

- Kết quả: tràn số không xảy ra

Registers	Coproc 1	Coproc 0
Name	Number	Value
\$zero	0	0
\$at	1	0
\$v0	2	0
\$v1	3	0
\$a0	4	0
\$a1	5	0
\$a2	6	0
\$a3	7	0
\$t0	8	0
\$t1	9	-11
\$t2	10	0
\$t3	11	0
\$t4	12	0
\$t5	13	0
\$t6	14	0
\$t7	15	0
\$s0	16	0
\$s1	17	-1
\$s2	18	10
\$s3	19	9
\$s4	20	0
\$s5	21	0
\$s6	22	0
\$s7	23	0
\$t8	24	0
\$t9	25	0
\$k0	26	0
\$k1	27	0
\$gp	28	268468224
\$sp	29	2147479548
\$fp	30	0
\$ra	31	0
pc		4194352
hi		0
lo		0

## ASSIGNMENT 2:

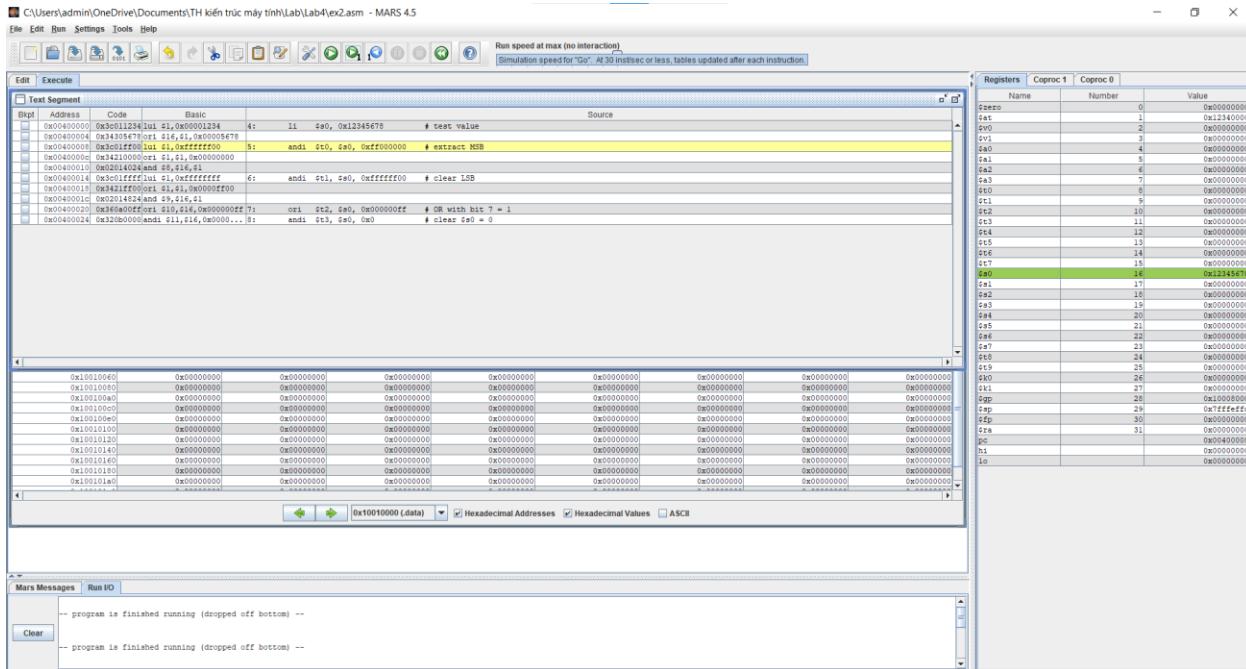
### 1. Code

```
1 #Laboratory Exercise 4, Home Assignment 2
2 #PhamVanAnh_20214988
3 .text
4     li    $s0, 0x12345678      # test value
5     andi $t0, $s0, 0xff000000  # extract MSB
6     andi $t1, $s0, 0xfffffff00 # clear LSB
7     ori  $t2, $s0, 0x000000ff # OR with bit 7 = 1
8     andi $t3, $s0, 0x00          # clear $s0 = 0
9
10
11
```

Line: 11 Column: 1 Show Line Numbers

Thực hiện gõ chương trình vào công cụ MARS

### 2. Thực hiện câu lệnh 4: Gán giá trị \$s0 = 0x12345678



### 3. Thực hiện câu lệnh 5: Extract MSB.

- AND 0x12345678 và 0xff000000
- Vì muốn extract “12” nên để “12” AND “ff” và các số khác AND với 0
- Gán giá trị tìm được vào thanh ghi \$t0
- Kết quả:

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

```

    .Text Segment
    B0f0
    ADDRESS  Code      Basic
    0x00400000 0x3a011234 lui $1,0x0000001234   t: li $s0, 0x12345678 # test value
    0x00400004 0x34305671 ori $16,$1,0x00000078
    0x00400008 0x3a01ff00 lui $1,0xfffffff000
    0x0040000c 0x34210000 ori $1,$1,0x00000000
    0x00400010 0x3a01ff00 lui $1,0xfffffff000
    0x00400014 0x3a01ffff lui $1,0xffffffff       d: andi $t1,$s0,0xfffffff00 # extract MSB
    0x00400018 0x3a210000 ori $1,$1,0x0000ff00
    0x0040001c 0x3a214124 andi $t1,$1,16,11
    0x00400020 0x3a010000 andi $t1,$1,0x000000ff    t: ori $t2,$s0,0x000000ff # OR with bit 3 = 1
    0x00400024 0x3a010000 andi $t1,$16,0x0000... 8: andi $t3,$s0,$0 # clear $s0 = 0

```

The registers window shows the following state:

Name	Number	Value
\$s0	0	0x00000000
\$at	1	0xff000000
\$v0	2	0x00000000
\$v1	3	0x00000000
\$a0	4	0x00000000
\$a1	5	0x00000000
\$a2	6	0x00000000
\$a3	7	0x00000000
\$t0	8	0xa12345678
\$t1	9	0x00000000
\$s2	10	0x00000000
\$t3	11	0x00000000
\$t4	12	0x00000000
\$t5	13	0x00000000
\$t6	14	0x00000000
\$t7	15	0x00000000
\$s0	16	0x12345678
\$s1	17	0x00000000
\$s2	18	0x00000000
\$s3	19	0x00000000
\$s4	20	0x00000000
\$s5	21	0x00000000
\$s6	22	0x00000000
\$s7	23	0x00000000
\$t8	24	0x00000000
\$t9	25	0x00000000
\$s1	26	0x00000000
\$k1	27	0x00000000
\$gp	28	0x10000000
\$sp	29	0x000ffffc
\$fp	30	0x00000000
\$ra	31	0x00000000
pc		0x00400014
hi		0x00000000
lo		0x00000000

The messages window shows the message: "program is finished running (dropped off bottom) --".

#### 4. Thực hiện câu lệnh 6: Clear LSB of \$s0

- AND 0x12345678 và 0x000000ff
- Vì muốn clear “78” nên để “78” AND “00” và các số khác AND với f
- Gán giá trị tìm được vào thanh ghi \$t1
- Kết quả:

C:\Users\admin\OneDrive\Documents\TH kiến trúc máy tính\Lab\Lab4\ex2.asm - MARS 4.5

The screenshot shows the MARS 4.5 assembly editor interface. The assembly code in the text segment is:

```

    .Text Segment
    B0pt Address Code Basic Source
    0x00400000 0x3c011234 lui $1,0x000001234
    0x00400004 0x3130347fori $1,1,0x0000047f8
    0x00400008 0x30000000000000000000000000000000
    0x0040000c 0x3421ff00ori $1,1,0xffffffff00
    0x00400010 0x30201402and $9,16,1
    0x00400014 0x3c01fffffdli $1,0xffffffffffff
    0x00400018 0x3421ff00ori $1,1,0xffffffff00
    0x0040001c 0x30201424and $9,16,1
    0x00400020 0x346000ffori $10,16,0x000000ff7
    0x00400024 0x3120b000andi $11,$16,0x00000000...$1: andi $t3,$s0,0x0
  
```

The Registers window shows the following values:

Name	Number	Value
zero	0	0x00000000
\$at	1	0xfffffff000
\$v0	2	0x00000000
\$v1	3	0x00000000
\$a0	4	0x00000000
\$a1	5	0x00000000
\$a2	6	0x00000000
\$a3	7	0x00000000
\$t0	8	0x1123456700
\$t1	9	0x1234567800
\$t2	10	0x00000000
\$t3	11	0x00000000
\$t4	12	0x00000000
\$t5	13	0x00000000
\$t6	14	0x00000000
\$t7	15	0x00000000
\$t8	16	0x12345678
\$t9	17	0x00000000
\$s0	18	0x00000000
\$s1	19	0x00000000
\$s2	20	0x00000000
\$s3	21	0x00000000
\$s4	22	0x00000000
\$s5	23	0x00000000
\$s6	24	0x00000000
\$t9	25	0x00000000
\$k0	26	0x00000000
\$k1	27	0x00000000
\$gp	28	0x10000000
\$sp	29	0x7fffffe000
\$fp	30	0x00000000
\$ra	31	0x00000000
pc		0x00400020
hi		0x00000000
lo		0x00000000

The Mars Messages window shows two entries:

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

## 5. Thực hiện câu lệnh số 7: Set LSB of \$s0 (bits 7 to 0 are set to 1)

- OR 0x12345678 và 0x000000ff
- Vì muốn bit 7 → 0 thành 1 nên OR 8 bit cuối với ff
- Gán giá trị tìm được vào thanh ghi \$t2
- Kết quả:

C:\Users\admin\OneDrive\Documents\TH kiến trúc máy tính\Lab\Lab4\ex2.asm - MARS 4.5

The screenshot shows the MARS 4.5 assembly editor interface. The assembly code in the text segment is identical to the previous screenshot.

The Registers window shows the following values:

Name	Number	Value
zero	0	0x00000000
\$at	1	0xfffffff000
\$v0	2	0x00000000
\$v1	3	0x00000000
\$a0	4	0x00000000
\$a1	5	0x00000000
\$a2	6	0x00000000
\$a3	7	0x00000000
\$t0	8	0x1123456700
\$t1	9	0x1234567800
\$t2	10	0x12345678
\$t3	11	0x00000000
\$t4	12	0x00000000
\$t5	13	0x00000000
\$t6	14	0x00000000
\$t7	15	0x00000000
\$t8	16	0x12345678
\$t9	17	0x00000000
\$s0	18	0x00000000
\$s1	19	0x00000000
\$s2	20	0x00000000
\$s3	21	0x00000000
\$s4	22	0x00000000
\$s5	23	0x00000000
\$s6	24	0x00000000
\$t9	25	0x00000000
\$k0	26	0x00000000
\$k1	27	0x00000000
\$gp	28	0x10000000
\$sp	29	0x7fffffe000
\$fp	30	0x00000000
\$ra	31	0x00000000
pc		0x00400020
hi		0x00000000
lo		0x00000000

The Mars Messages window shows two entries:

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

## 6. Thực hiện câu lệnh số 8: Clear \$s0

- ADD 0x12345678 và 0x00000000
- Vì để clear \$s0 cần add \$s0 và 0
- Gán giá trị tìm được vào thanh ghi \$t3
- Kết quả:

C:\Users\admin\OneDrive\Documents\TH kiến trúc máy tính\Lab\Lab4\ex2.asm - MARS 4.5

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

```

Text Segment
B1pt Address Code Basic Source
0x00000000 0x00000000 li $s0, 0x12345678 # test value
0x00000004 0x00000000 addi $s0, $s0, 0xffffffff7F
0x00000008 0x00000000 lui $t1, $fffffe00
0x0000000C 0x00000000 andi $t0, $s0, $fffffe00 # extract MSB
0x00000010 0x00000000 ori $t1, $t1, $00000000
0x00000014 0x00000000 andi $t1, $t1, 0x11111111
0x00000018 0x00000000 andi $t1, $t1, 0x00000000
0x0000001C 0x00000000 andi $t1, $t1, 0x00000000
0x00000020 0x00000000 ori $t2, $s0, 0x000000ff # OR with bit 7 = 1
0x00000024 0x00000000 andi $t3, $t2, $00

```

The Registers window shows the following register values:

Registers	Name	Number	Value
gzero		0	0x00000000
\$s0		1	0xffffffff00
\$v0		2	0x00000000
\$v1		3	0x00000000
\$a0		4	0x00000000
\$a1		5	0x00000000
\$a2		6	0x00000000
\$a3		7	0x00000000
\$t0		8	0x12345678
\$t1		9	0x12345678
\$t2		10	0x12345678
\$t3		11	0x00000000
\$t4		12	0x00000000
\$t5		13	0x00000000
\$t6		14	0x00000000
\$t7		15	0x00000000
\$a0		16	0x12345678
\$s1		17	0x00000000
\$a2		18	0x00000000
\$a3		19	0x00000000
\$a4		20	0x00000000
\$a5		21	0x00000000
\$a6		22	0x00000000
\$t8		23	0x00000000
\$t9		24	0x00000000
\$t10		25	0x00000000
\$t11		26	0x00000000
\$t12		27	0x00000000
\$gp		28	0x10000000
\$sp		29	0x7fffffe0
\$fp		30	0x00000000
\$ra		31	0x00000000
pc			0x00400024
hi			0x00000000
lo			0x00000000

The Memory dump window shows memory starting at address 0x10010000.

The Mars Messages window shows the message: "program is finished running (dropped off bottom) --".

## ASSIGNMENT 3:

### 1. abs \$s0, s1

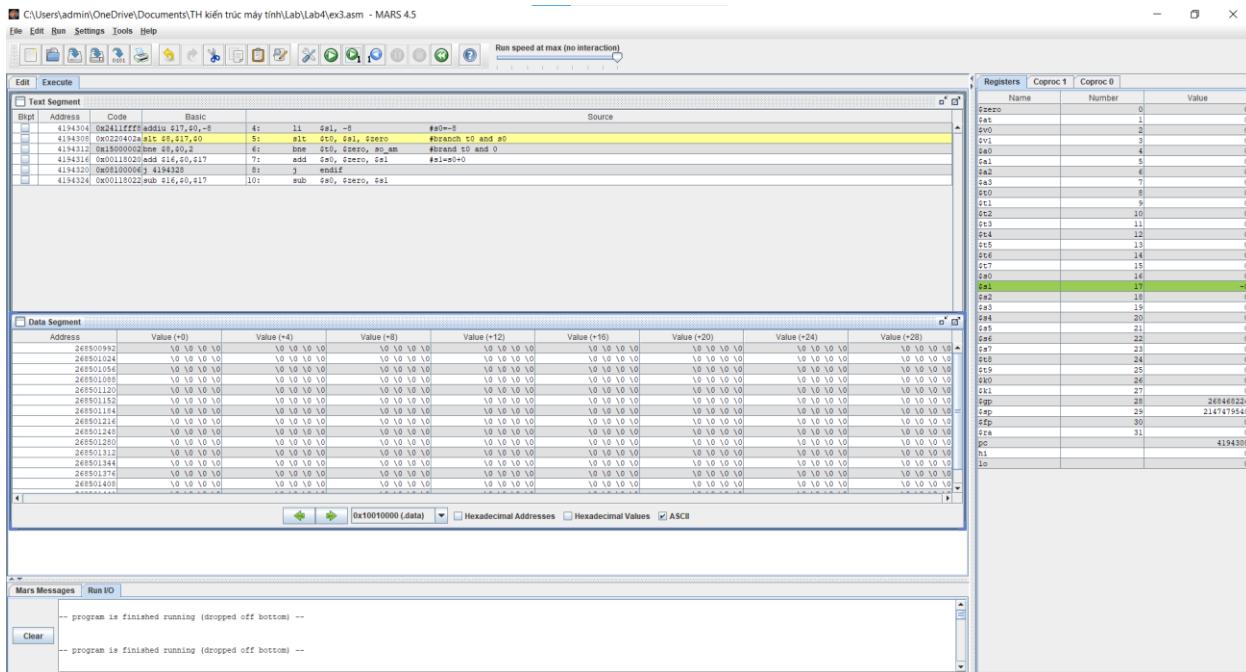
- Code

```

ex1.asm  mips1.asm*  ex2.asm  ex3.asm  ex4.asm  ex5.asm
1  #Laboratory Exercise 4, Home Assignment 3
2  #PhamVanAnh_20214988
3  .text
4      li      $s1, -8          #s0=-8
5      slt    $t0, $s1, $zero   #branch t0 and s0
6      bne    $t0, $zero, so_am #branch t0 and 0
7      add    $s0, $zero, $s1    #s1=s0+0
8      j      endif
9 so_am:
10     sub   $s0, $zero, $s1
11 endif:
12
13
14
15

```

- Thực hiện câu lệnh 4: Gán \$s1 thành -8



- Thực hiện câu lệnh 5: So sánh s1 và 0.

➤ Nếu  $s1 < 0 \rightarrow t0 = 1 \rightarrow so\_am: s0 = 0 - s1$

C:\Users\admin\OneDrive\Documents\TH kiến trúc máy tính\Lab\Lab4\ex3.asm - MARS 4.5

```

1 #Laboratory Exercise 4, Home Assignment 3
2 #PhamVanAnh_20214988
3 .text
4 li    $s1, -8      #s0=-8
5 slt  $t0, $s1, $zero   #branch t0 and s0
6 bne  $t0, $zero, so_am #branch t0 and 0
7 add  $s0, $zero, $s1  #s1=s0+0
8 j     endif
9 so_am:
10 sub  $s0, $zero, $s1
11 endif:
12
13
14
15
16
17
18
19
20
  
```

Mars Messages: -- program is finished running (dropped off bottom) --  
Run IO: -- program is finished running (dropped off bottom) --

C:\Users\admin\OneDrive\Documents\TH kiến trúc máy tính\Lab\Lab4\ex3.asm - MARS 4.5

Mars Messages: -- program is finished running (dropped off bottom) --  
Run IO: -- program is finished running (dropped off bottom) --

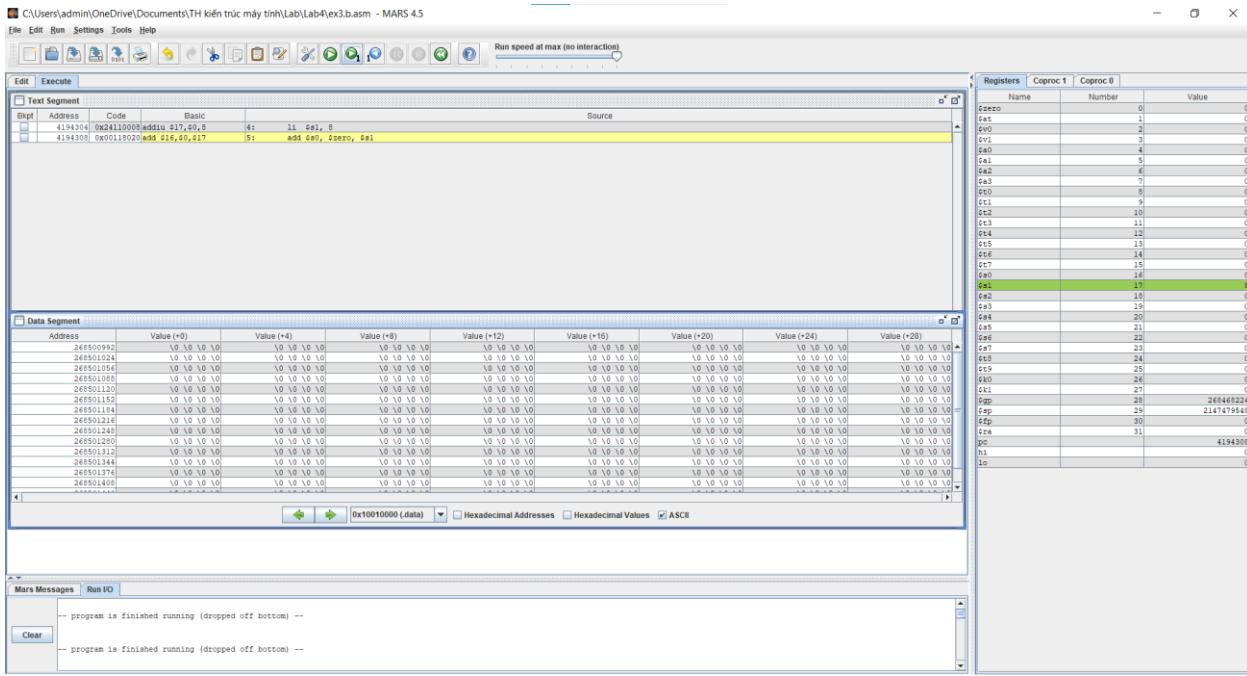
➤ Nếu  $s1 > 0 \rightarrow t0 = 1 \rightarrow$  Gán  $s0 = s1$

## 2. move \$s0, \$t1

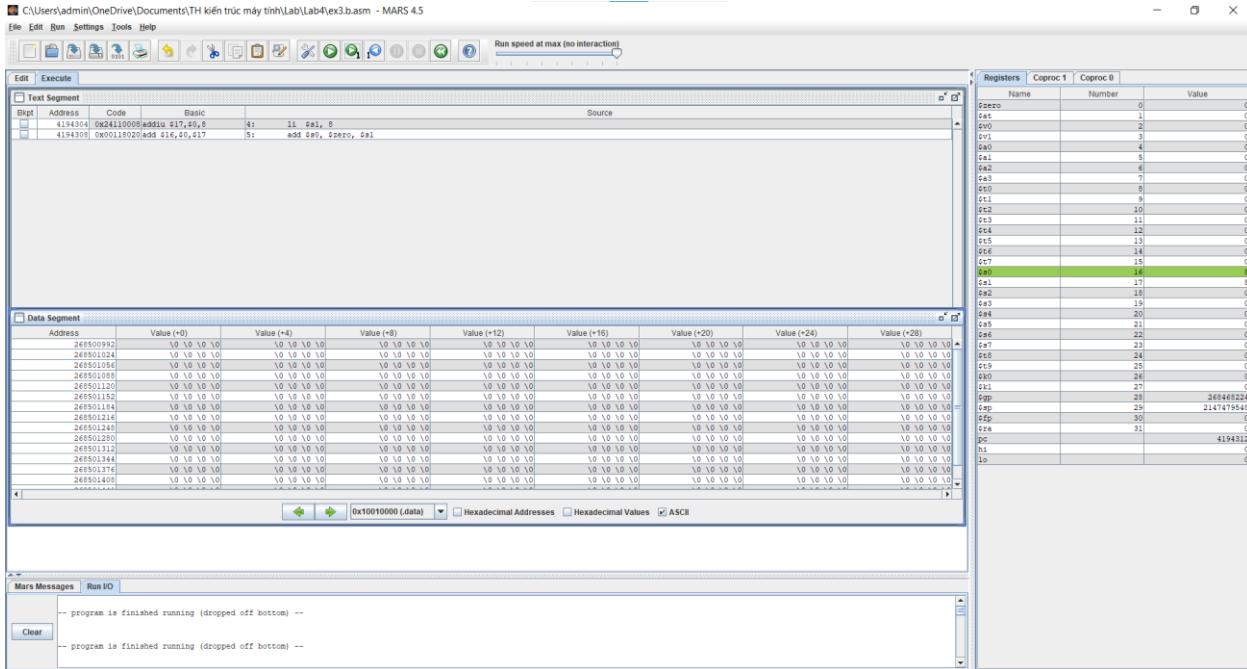
- Code

```
1 #Laboratory Exercise 4, Home Assignment 3
2 #PhamVanAnh_20214988
3 .text
4     li    $t1, 8
5     add   $s0, $zero, $t1
6
7
```

- Thực hiện câu lệnh 4: Gán giá trị \$t1 = 8



- Thực hiện câu lệnh 5: \$s0 = 0 + \$s1



### 3. not \$s0, s1

- Code

```

1 #Laboratory Exercise 4, Home Assignment 3
2 #PhamVanAnh_20214988
3 .text
4     li      $s1, 0x20214988
5     xor   $s0, $s1, 0xffffffff
6
7
8

```

- Thực hiện dòng lệnh số 4: Gán s1 = 0x20214988

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

```

    .text
    li    $s1, 0x20214988
    xor $s0, $s1, 0xffffffff

```

The registers window shows the following values:

Name	Number	Value
zero	0	0x00000000
\$at	1	0x21000000
\$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
\$a0	16	0x00000000
\$s1	17	0x20214988
\$a2	18	0x00000000
\$a3	19	0x00000000
\$t1	20	0x00000000
\$a5	21	0x00000000
\$t2	22	0x00000000
\$t3	23	0x00000000
\$t4	24	0x00000000
\$t5	25	0x00000000
\$t6	26	0x00000000
\$t7	27	0x00000000
\$gp	28	0x10000000
\$sp	29	0x7fffffe0
\$fp	30	0x00000000
\$ra	31	0x00000000
\$pc		0x00000000
\$hi		0x00000000
\$lo		0x00000000

The memory dump window shows the data segment from address 0x10010000 to 0x10010010. The messages window shows two entries: "program is finished running (dropped off bottom) --" repeated twice.

- Thực hiện dòng lệnh số 5: XOR \$s1 với 0xffffffff → đảo bit 1 thành 0 và ngược lại

The screenshot shows the MARS 4.5 assembly debugger interface. The assembly window displays the same code as the previous screenshot:

```

    .text
    li    $s1, 0x20214988
    xor $s0, $s1, 0xffffffff

```

The registers window shows the following values:

Name	Number	Value
zero	0	0x00000000
\$at	1	0x21000000
\$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
\$a0	16	0x00000000
\$s1	17	0x20214988
\$a2	18	0x00000000
\$a3	19	0x00000000
\$t1	20	0x00000000
\$t2	21	0x00000000
\$t3	22	0x00000000
\$t4	23	0x00000000
\$t5	24	0x00000000
\$t6	25	0x00000000
\$t7	26	0x00000000
\$gp	27	0x10000000
\$sp	28	0x7fffffe0
\$fp	29	0x00000000
\$ra	30	0x00000000
\$pc		0x00000000
\$hi		0x00000000
\$lo		0x00000000

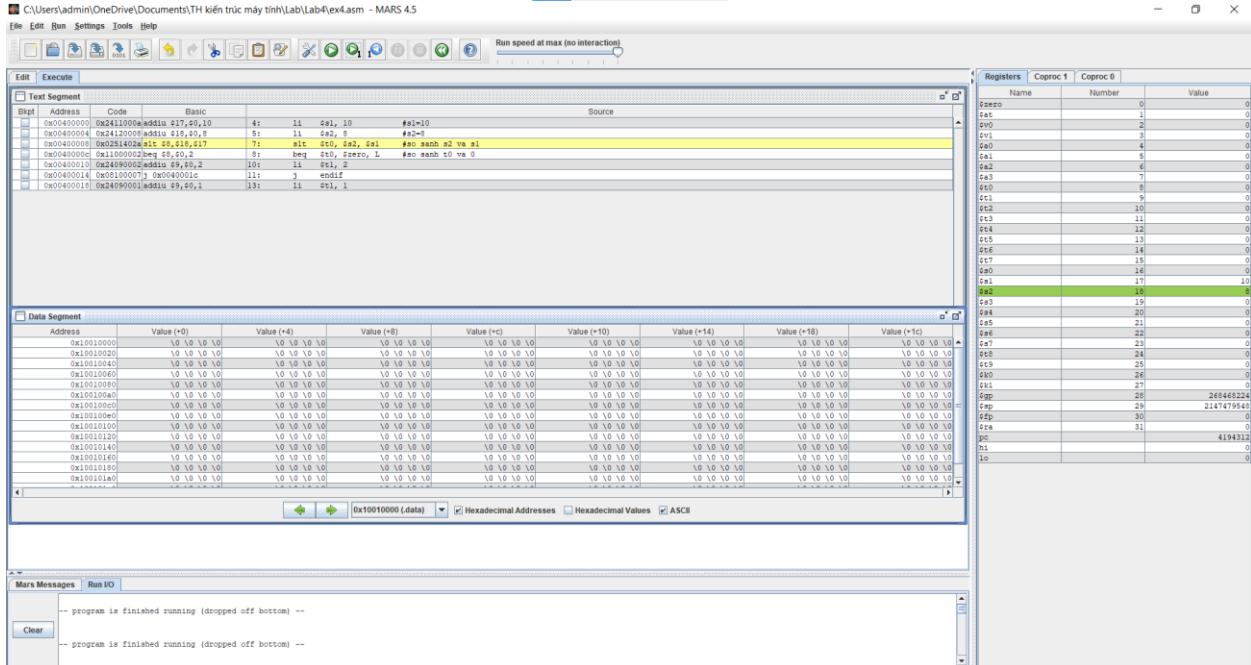
The memory dump window shows the data segment from address 0x10010000 to 0x10010010. The messages window shows two entries: "program is finished running (dropped off bottom) --" repeated twice.

## 4. ble \$s1, s2, L

### - Code

```
1 #Laboratory Exercise 4, Home Assignment 3
2 #PhamVanAnh_20214988
3 .text
4     li    $s1, 10          #s1=10
5     li    $s2, 8           #s2=8
6
7     slt   $t0, $s2, $s1      #so sanh s2 va s1
8     beq   $t0, $zero, L       #so sanh t0 va 0
9
10    li    $t1, 2
11    j     endif
12 L:
13    li    $t1, 1
14 endif:
15
16
17
18
```

- Thực hiện câu lệnh 4+5: Gán giá trị s1=10, s2=8



- Thực hiện câu lệnh 7: So sánh s1 và s2

- Thực hiện câu lệnh 8: So sánh t0 và 0
  - Nếu  $t0 = 0 \rightarrow$  Gán  $t1 = 2$
  - Nếu  $t0 = 1 \rightarrow$  L: gán  $t1 = 1$
- Kết quả:

The screenshot shows the MARS 4.5 assembly debugger interface. The assembly code in the editor is as follows:

```

1 #Laboratory Exercise 4, Home Assignment 4
2 #PhamVanAnh_20214988
3 .text
4 li $s1, 10      #s1=10
5 li $s2, 8       #s2=8
6
7 slt $t0, $s2, $s1    #so sanh s2 va s1
8 beq $t0, $zero, L   #so sanh t0 va 0
9
10 li $t1, 2
11 j endif
12 L: li $t1, 1
13 endif:
14
15
16
17
18
19
20

```

The Registers window shows the following register values:

Name	Number	Value
\$zero	0	0
\$at	1	0
\$v0	2	0
\$v1	3	0
\$a0	4	0
\$a1	5	0
\$a2	6	0
\$a3	7	0
\$t0	8	1
\$t1	9	2
\$t2	10	0
\$t3	11	0
\$t4	12	0
\$t5	13	0
\$t6	14	0
\$t7	15	0
\$s0	16	0
\$s1	17	10
\$s2	18	8
\$s3	19	0
\$s4	20	0
\$s5	21	0
\$s6	22	0
\$s7	23	0
\$t8	24	0
\$t9	25	0
\$t10	26	0
\$t11	27	0
\$gp	28	267469224
\$sp	29	2147479541
\$fp	30	0
\$ra	31	0
pc		4194332
hi		0
lo		0

The Mips Messages window displays the message: '-- program is finished running (dropped off bottom) --'.

## ASSIGNMENT 4:

### 1. Code

The screenshot shows the MARS 4.5 assembly debugger interface. The assembly code in the editor is as follows:

```

1 #Laboratory Exercise 4, Home Assignment 4
2 #PhamVanAnh_20214988
3
4 .text
5     add $s1, $zero, 0xffffffff
6     add $s2, $zero, 4
7 start:
8     li $t0, 0          #No Overflow is default status
9     addu $s3, $s1, $s2 # s3 = s1 + s2
10    xor $t1, $s1, $s2 #Test if $s1 and $s2 have the same sign
11    bltz $t1, EXIT    #If not, exit
12
13    xor $t2, $s3, $s1 #Test if $s3 and $s1 have the same sign
14    blt $t2, $zero, OVERFLOW #If t2 < 0 --> OVERFLOW
15 OVERFLOW:
16    li $t0, 8
17 EXIT:
18
19

```

2. Thực hiện câu lệnh số 5+6: Gán \$s1 = 0xffffffff, \$s2 = 4

The screenshot shows the MARS 4.5 assembly debugger interface. The top menu bar includes File, Edit, Run, Settings, Tools, Help, and a Run speed at max (no interaction) button. The left sidebar has sections for Text Segment and Data Segment. The Text Segment pane displays assembly code with columns for Bp#, Address, Code, Basic, Source, and Registers/Coproc 1/Coproc 0. The Data Segment pane shows memory starting at address 0x10010000 with various registers listed. The bottom pane displays Mars Messages and Run IO, with the message "program is finished running (dropped off bottom)" appearing twice.

3. Thực hiện câu lệnh số 9: Gán s3 = s1 + s2

The screenshot shows the MARS 4.5 assembly debugger interface. The top menu bar includes File, Edit, Run, Settings, Tools, Help, and a Run speed dropdown. Below the menu is a toolbar with various icons for file operations and assembly manipulation. The main window is divided into several panes:

- Text Segment:** Shows assembly code with columns for Bkpt, Address, Code, Basic, and Source. The code includes instructions like lui, ori, add, sub, and bit operations.
- Data Segment:** Shows memory starting at address 0x10010000 with values for .data, .bss, and .text sections.
- Registers:** A table showing the state of general-purpose registers (\$zero through \$t0-\$t7, \$s0-\$s1, \$t8-\$t9, \$k0-\$k1, \$gp-\$sp, \$fp-\$ra, \$pc, \$hi, \$lo).
- Coproc 1:** Shows floating-point register values.
- Coproc 0:** Shows floating-point register values.
- Mars Messages:** Displays log messages indicating the program has finished running.
- Run IO:** A pane for monitoring I/O operations.

#### 4. Thực hiện câu lệnh số 10:

- XOR \$s1 và \$s2 → Gán giá trị vào \$t1
    - Nếu \$s1 và \$s2 khác dấu → EXIT
    - Nếu \$s1 và \$s2 cùng dấu → chạy tiếp

C:\Users\SinhVien\Documents\va\lab4\assign4.asm - MARS 4.5

File Edit Run Settings Tools Help

Run speed at max (no interaction)

Registers Coproc 1 Coproc 0

Name	Number	Value
\$zero	0	0x00000000
\$at	1	0xffffffff
\$v0	2	0x00000000
\$v1	3	0x00000000
\$a0	4	0x00000000
\$a1	5	0x00000000
\$a2	6	0x00000000
\$a3	7	0x00000000
\$t0	8	0x00000000
\$t1	9	0xfffffffffb
\$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	0x00000003
\$s4	20	0x00000000
\$s5	21	0x00000000
\$s6	22	0x00000000
\$s7	23	0x00000000
\$t8	24	0x00000000
\$t9	25	0x00000000
\$t0	26	0x00000000
\$t1	27	0x00000000
\$gp	28	0x10008000
\$sp	29	0xfffffffffc
\$fp	30	0x00000000
\$ra	31	0x00000000
pc		0x0400001c
hi		0x00000000
lo		0x00000000

Mars Messages Run I/O

```
-- program is finished running (dropped off bottom) --
Clear
-- program is finished running (dropped off bottom) --
```

## 5. Thực hiện câu lệnh số 13:

- XOR \$s3 và \$s1 → gán giá trị vào \$t2

C:\Users\SinhVien\Documents\va\lab4\assign4.asm - MARS 4.5

File Edit Run Settings Tools Help

Run speed at max (no interaction)

Registers Coproc 1 Coproc 0

Name	Number	Value
\$zero	0	0x00000000
\$at	1	0xffffffff
\$v0	2	0x00000000
\$v1	3	0x00000000
\$a0	4	0x00000000
\$a1	5	0x00000000
\$a2	6	0x00000000
\$a3	7	0x00000000
\$t0	8	0x00000000
\$t1	9	0xfffffffffb
\$t2	10	0xfffffffffc
\$t3	11	0x00000000
\$t4	12	0x00000000
\$t5	13	0x00000000
\$t6	14	0x00000000
\$t7	15	0x00000000
\$s0	16	0x00000000
\$s1	17	0xffffffff
\$s2	18	0x00000004
\$s3	19	0x00000003
\$s4	20	0x00000000
\$s5	21	0x00000000
\$s6	22	0x00000000
\$s7	23	0x00000000
\$t8	24	0x00000000
\$t9	25	0x00000000
\$t0	26	0x00000000
\$t1	27	0x00000000
\$gp	28	0x10008000
\$sp	29	0xfffffffffc
\$fp	30	0x00000000
\$ra	31	0x00000000
pc		0x04000024
hi		0x00000000
lo		0x00000000

Mars Messages Run I/O

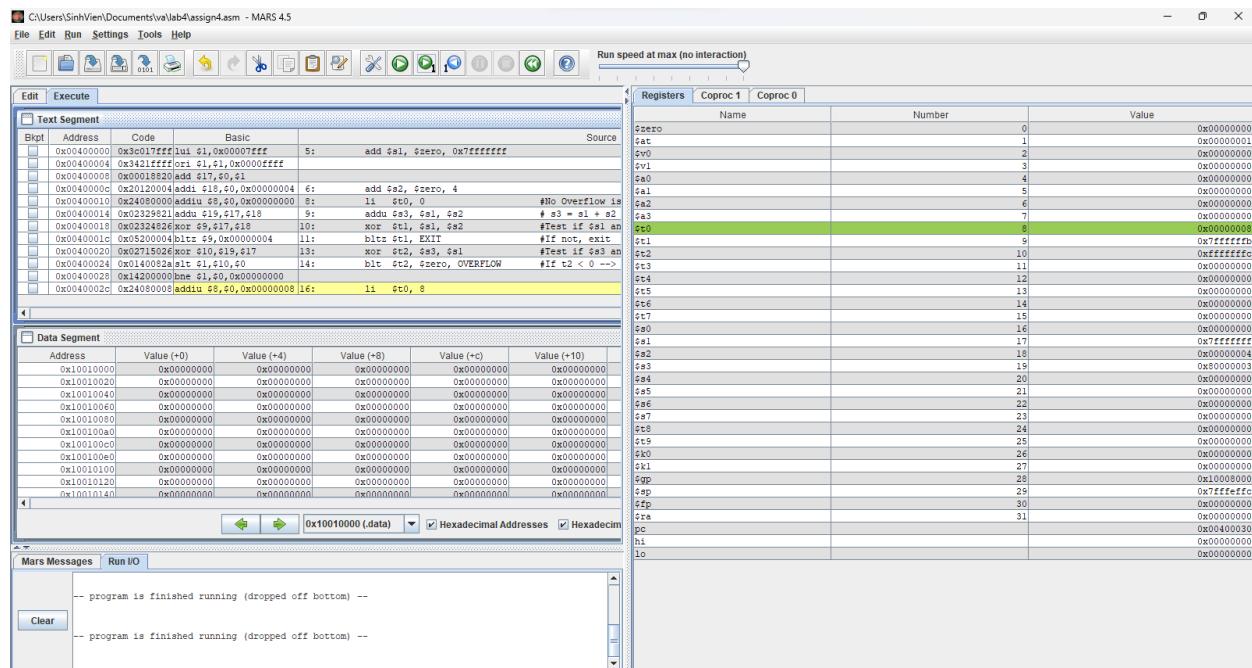
```
-- program is finished running (dropped off bottom) --
Clear
-- program is finished running (dropped off bottom) --
```

## 6. Thực hiện câu lệnh số 14:

- So sánh t2 và 0

➤ Nếu  $t2 < 0 \rightarrow$  OVFLOW  $\rightarrow t0 = 8$

➤ Nếu  $t2 > 0 \rightarrow$  END



## ASSIGNMENT 5:

### 1. Code

```
ex5.asm ex5().asm*
1 #Laboratory Exercise 4, Home Assignment 5
2 #PhamVanAnh_20214988
3 .text
4     li    $s0, 12
5     sll  $t1, $s0 ,1      # s0 * (2^1)
6     sll  $t2, $s0 ,2      # s0 * (2^2)
7     sll  $t3, $s0 ,3      # s0 * (2^3)
8     sll  $t4, $s0 ,4      # 12 * (2^4)
9
10
11
12
13
14
15
```

## 2. Thực hiện câu lệnh 4:

- Gán giá trị s0 = 12

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

```

Text Segment
B1pt Address Code Basic Source
0x00400000 0x2410000c addi $16, $0,12 4: li $s0, 12
0x00400004 0x00104141 sll $t1, $s0, 1 5: sll $t1, $s0, 1 # s0 ^ (2^1)
0x00400008 0x00105580 sll $t1, $t1, 2 6: sll $t1, $t1, 2 # s0 ^ (2^2)
0x0040000c 0x00105580 sll $t1, $t1, 3 7: sll $t1, $t1, 3 # s0 ^ (2^3)
0x00400010 0x00104100 sll $t1, $t1, 4 8: sll $t1, $t1, 4 # s0 ^ (2^4)

```

The Registers window shows the state of registers after execution:

Name	Number	Value
\$s0	0	0
\$at	1	0
\$v0	2	0
\$v1	3	0
\$s1	4	0
\$s2	5	0
\$s3	6	0
\$s4	7	0
\$s5	8	0
\$s6	9	0
\$s7	10	0
\$t1	11	0
\$t2	12	0
\$t3	13	0
\$t4	14	0
\$t5	15	0
\$t6	16	12
\$t7	17	0
\$s8	18	0
\$s9	19	0
\$s10	20	0
\$s11	21	0
\$s12	22	0
\$s13	23	0
\$s14	24	0
\$s15	25	0
\$s16	26	0
\$s17	27	0
\$sp	28	268446224
\$fp	29	214747545
\$tp	30	0
\$ra	31	0
pc		4194395
hi		0
lo		0

## 3. Thực hiện câu lệnh số 5: Dịch sang trái 1 bit của S0 → \$t1

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

```

Text Segment
B1pt Address Code Basic Source
0x00400000 0x2410000c addi $16, $0,12 4: li $s0, 12
0x00400004 0x00104141 sll $t1, $s0, 1 5: sll $t1, $s0, 1 # s0 ^ (2^1)
0x00400008 0x00105580 sll $t1, $t1, 2 6: sll $t1, $t1, 2 # s0 ^ (2^2)
0x0040000c 0x00105580 sll $t1, $t1, 3 7: sll $t1, $t1, 3 # s0 ^ (2^3)
0x00400010 0x00104100 sll $t1, $t1, 4 8: sll $t1, $t1, 4 # s0 ^ (2^4)

```

The Registers window shows the state of registers after execution:

Name	Number	Value
\$s0	0	0
\$at	1	0
\$v0	2	0
\$v1	3	0
\$s1	4	0
\$s2	5	0
\$s3	6	0
\$s4	7	0
\$s5	8	0
\$s6	9	24
\$s7	10	0
\$t1	11	0
\$t2	12	0
\$t3	13	0
\$t4	14	0
\$t5	15	0
\$t6	16	12
\$t7	17	0
\$s8	18	0
\$s9	19	0
\$s10	20	0
\$s11	21	0
\$s12	22	0
\$s13	23	0
\$s14	24	0
\$s15	25	0
\$s16	26	0
\$s17	27	0
\$sp	28	268446224
\$fp	29	214747545
\$tp	30	0
\$ra	31	0
pc		4194312
hi		0
lo		0

#### 4. Thực hiện câu lệnh số 6: Dịch sang trái 2 bit của s0 → \$t2

The screenshot shows the MARS 4.5 assembly editor interface. The assembly code for Lab4\ex50.asm is displayed in the main window. The registers window shows various registers with their names, numbers, and current values. The \$t2 register is highlighted in yellow, indicating it is the target of the current operation. The data segment window shows memory starting at address 0x10010000 with all values set to 0.

#### 5. Thực hiện câu lệnh số 7: Dịch sang trái 3 bit của \$s0 → \$t3

The screenshot shows the MARS 4.5 assembly editor interface. The assembly code for Lab4\ex50.asm is displayed in the main window. The registers window shows various registers with their names, numbers, and current values. The \$t3 register is highlighted in yellow, indicating it is the target of the current operation. The data segment window shows memory starting at address 0x10010000 with all values set to 0.

## 6. Thực hiện câu lệnh số 8: Dịch sang trái 4 bit của \$s0 → \$t4

The screenshot shows the MARS 4.5 assembly debugger interface. The assembly code in the Test Segment window is:

```
Blpt    Address      Code      Bytes
0x00400000 0x41000000 addi    $t0,$s0,12
0x00400004 0x70104440 sll    $t1,$t0,1    # s0 ^ (2^1)
0x00400008 0x70104000 sll    $t2,$s0,2    # s0 ^ (2^2)
0x0040000C 0x70104000 sll    $t3,$s0,3    # s0 ^ (2^3)
0x00400010 0x70104000 sll    $t4,$s0,4    # s0 ^ (2^4)
```

The Registers window shows the state of the registers after the execution of the fourth instruction:

Name	Number	Value
\$zero	0	0
\$at	1	0
\$v0	2	0
\$v1	3	0
\$s0	4	0
\$s1	5	0
\$s2	6	0
\$s3	7	0
\$t0	8	0
\$t1	9	24
\$t2	10	48
\$t3	11	96
<b>\$t4</b>	<b>12</b>	<b>192</b>
\$t5	13	0
\$t6	14	0
\$t7	15	0
\$t8	16	12
\$t9	17	0
\$t10	18	0
\$t11	19	0
\$t12	20	0
\$t13	21	0
\$t14	22	0
\$t15	23	0
\$t16	24	0
\$t17	25	0
\$t18	26	0
\$t19	27	0
\$sp	28	201446224
\$gp	29	2147475451
\$tp	30	0
\$ra	31	0
pc		4194324
hi		0
lo		0

The Data Segment window shows memory starting at address 0x10010000. The Registers window also shows the value 192 in register \$t4.