

Báo cáo thực hành tuần 4

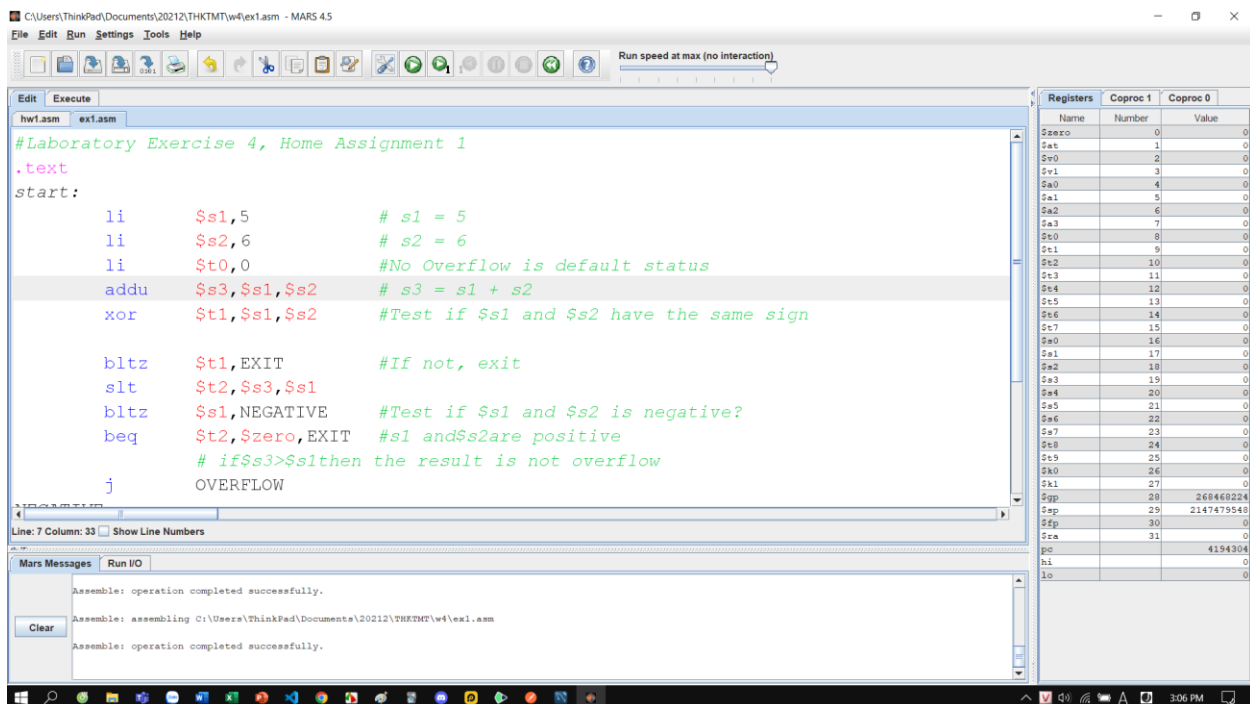
Phùng Ngọc Vinh – 20194719

Bài 1:

*Trạng thái Not overflow:

- 2 số dương:

\$s1 = 5; \$s2 = 6



The screenshot shows the MARS MIPS simulator interface. The main window displays assembly code for a program that adds two positive numbers, \$s1 (5) and \$s2 (6), and stores the result in \$t0. The code includes comments explaining each step. The 'Registers' window on the right shows the current state of the MIPS registers, with \$t0 containing the value 8. The 'Mars Messages' window at the bottom shows the successful completion of the assembly and execution process.

```
#Laboratory Exercise 4, Home Assignment 1
.text
start:
    li    $s1,5           # s1 = 5
    li    $s2,6           # s2 = 6
    li    $t0,0           #No Overflow is default status
    addu  $s3,$s1,$s2     # s3 = s1 + s2
    xor   $t1,$s1,$s2     #Test if $s1 and $s2 have the same sign

    bltz  $t1,EXIT        #If not, exit
    slt   $t2,$s3,$s1
    bltz  $s1,NEGATIVE    #Test if $s1 and $s2 is negative?
    beq   $t2,$zero,EXIT  #s1 and $s2 are positive
    # if $s3 > $s1 then the result is not overflow
    j     OVERFLOW

EXIT:
    j     OVERFLOW
```

Register	Value
\$zero	0
\$at	0
\$v0	0
\$v1	0
\$a0	0
\$a1	0
\$a2	0
\$a3	0
\$t0	8
\$t1	0
\$t2	0
\$t3	0
\$t4	0
\$t5	0
\$t6	0
\$t7	0
\$s0	0
\$s1	5
\$s2	6
\$s3	11
\$s4	0
\$s5	0
\$s6	0
\$s7	0
\$s8	0
\$s9	0
\$s10	0
\$s11	0
\$gp	268460224
\$ap	2147479548
\$fp	0
\$ra	0
pc	4194304
hi	0
lo	0

Mars Messages: Run I/O

Assembly: operation completed successfully.

Assembly: assembling C:\Users\ThinkPad\Documents\20212\THKMT\w4\ex1.asm

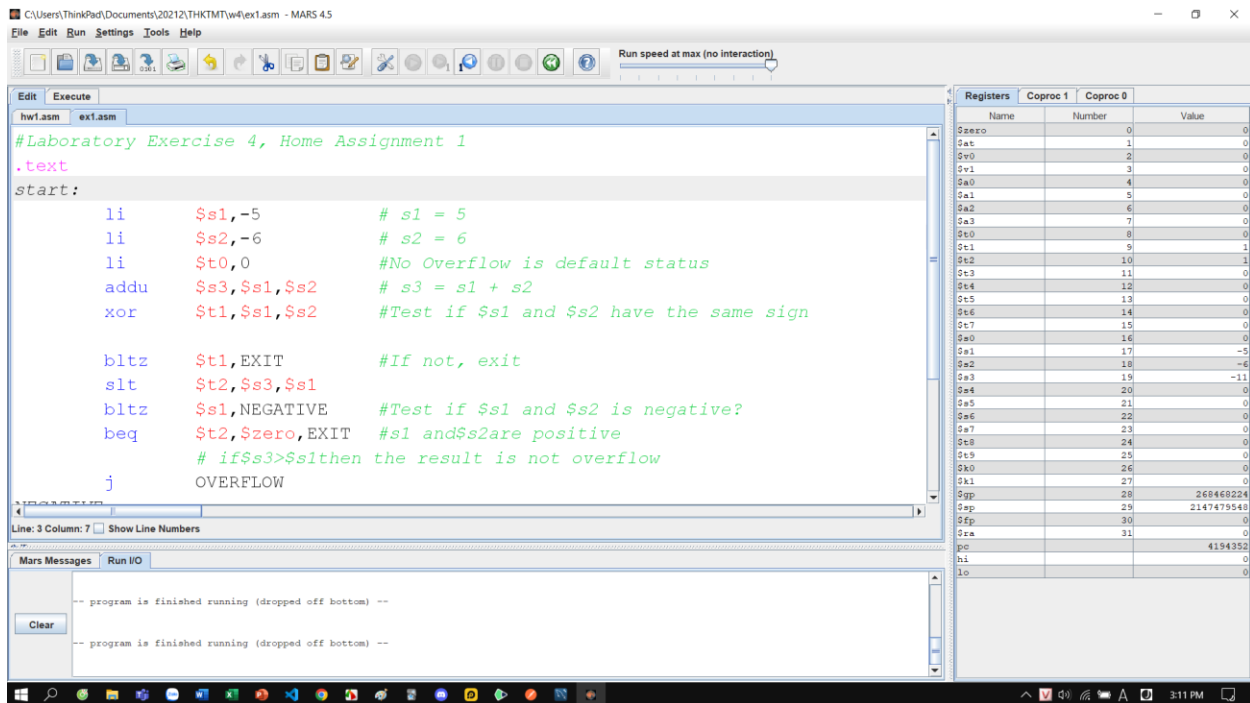
Assembly: operation completed successfully.

Kết quả:

\$t0	8	0
------	---	---

- 2 số âm:

\$s1 = -5; \$s2 = -6

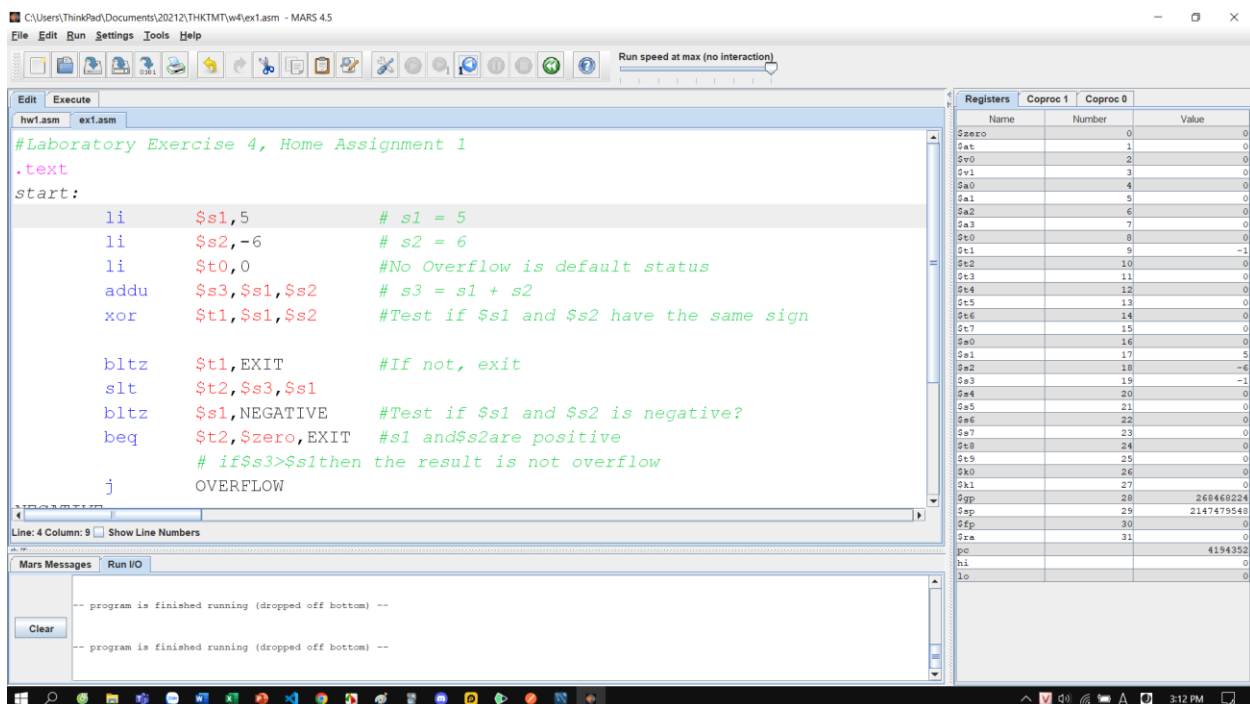


Kết quả:

\$t0	8	0
------	---	---

- 2 số khác dấu:

\$s1 = 5; \$s2 = -6



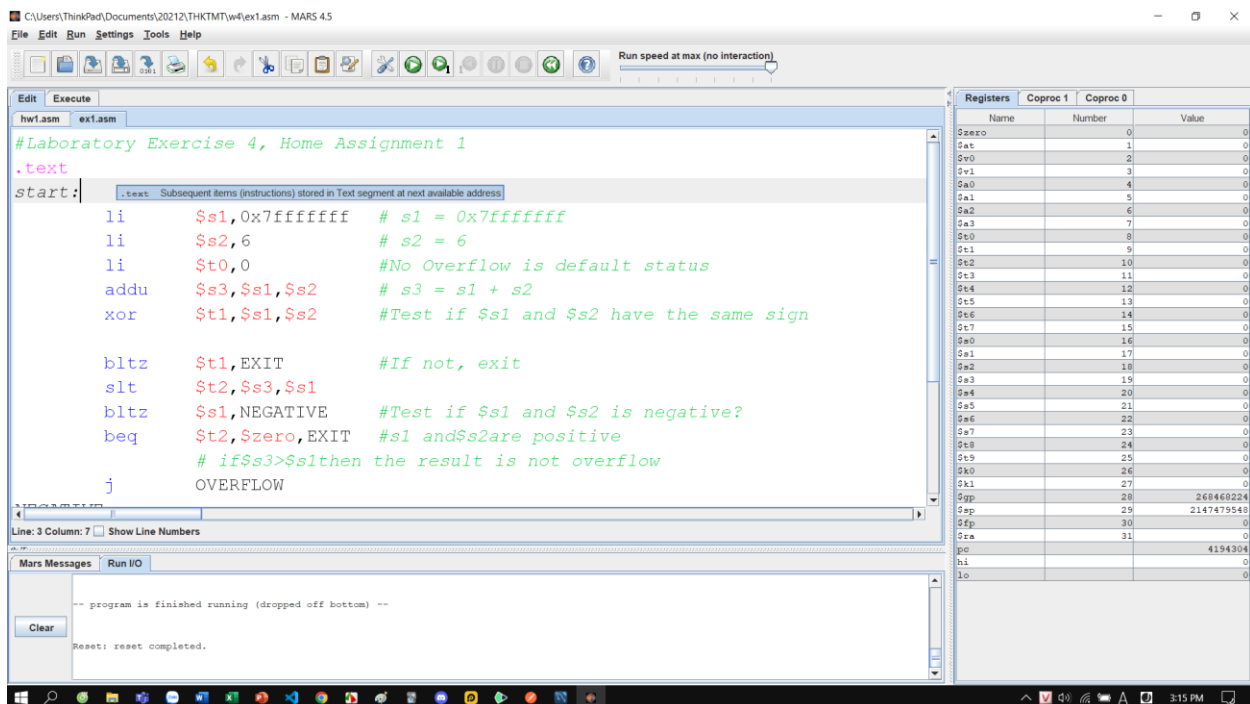
Kết quả:

\$t0	8	0
------	---	---

* Trạng thái Overflow:

- 2 số dương:

\$s1 = 0x7fffffff; \$s2 = 6

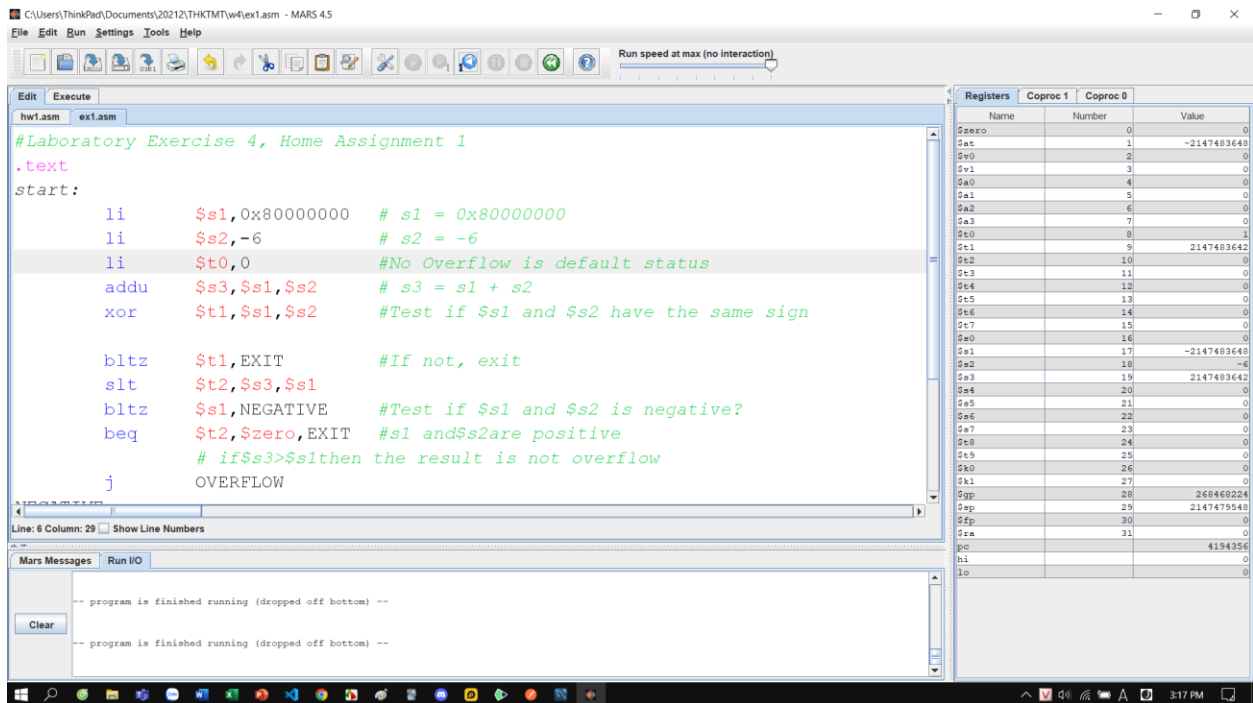


Kết quả:

\$t0	8	1
------	---	---

- 2 số âm:

\$s1 = 0x80000000; \$s2 = -6



Kết quả:

\$t0	8	1
------	---	---

Bài 2:

.text

```

li    $s0,0x12345678

andi  $t0,$s0,0xff000000    #Extract MSB of $s0

andi  $t1,$s0,0xffffffff00  #Clear LSB of $s0

ori    $t2,$s0,0x000000ff    #Set LSB of $s0(bits 7 to 0 are set to 1)

xor    $s0,$s0,$s0           #Clear $s0($s0=0, must use logical
                              instructions)

```

*Kết quả:

Extract MSB of \$s0

\$t0	8	0x12000000
------	---	------------

Clear LSB of \$s0

\$t1	9	0x12345600
------	---	------------

Set LSB of \$s0(bits 7 to 0 are set to 1)

\$t2	10	0x123456ff
------	----	------------

Clear \$s0(\$s0=0, must use logical instructions)

\$s0	16	0x00000000
------	----	------------

Bài 3:

a. abs \$s0,\$s1

.text

```
li    $s1,26          #gan s1 = -26
li    $s2,1
bltz  $s1,SOAM         #kiem tra dau cua s1
add   $s0,$s1,$0
j     EXIT
```

SOAM:

```
sub   $t0,$s1,$s2     #tru s1 cho 1
not   $s0,$t0         #dao bit
```

EXIT:

b. move \$s0,\$s1

.text

```
li    $s1, 26         #gan s1 = 26
add   $s0,$s1,$0      #s0 = s1
```

c. not \$s0, \$s1

.text

li \$s1, 26 #gan s1 = 26

nor \$s0, \$s1, \$0

d. ble \$s1, \$s2, label

.text

li \$s1, 26 #gan s1 = 26

li \$s2, 75 #gan s2 = 7

sub \$t0, \$s1, \$s2 #t0 = s1 - s2

blez \$t0, LABEL #t0 <= 0 ?

j EXIT

LABEL:

li \$s3, 2001 #kiem tra

EXIT:

Câu 4:

.text

addi \$s1, \$zero, 0x7fffffff

addi \$s2, \$zero, 1

li \$t0, 0 # trang thai Not overflow

addu \$s3, \$s1, \$s2

xor \$t1, \$s1, \$s2 # check dau s1 va s2

bltz \$t1, EXIT

xor \$t1, \$s1, \$s3 # check dau s1 va s3

bgtz \$t1, EXIT

OVERFLOW:

```
li $t0, 1      #Overflow
```

EXIT:

Câu 5:

```
$s3 = $s1 * $s2
```

.text

```
li    $s1,26      #gan s1 = 26
```

```
li    $s2,8       #gan s2 = 8
```

```
srl   $t0,$s2,1   #dich phai 1 bit
```

```
li    $t1,1       #count = 1
```

```
loop:  beq $t0,1,enloop  #dk dung
```

```
srl   $t0,$t0,1   #tipe tuc dich phai 1 bit
```

```
addi  $t1,$t1,1   # t1 = t1 + 1
```

```
j     loop
```

enloop:

```
sllv  $s3,$s1,$t1  #s3 = s1 * (2^count)
```

EXIT:

