# BÁO CÁO TUẦN 4

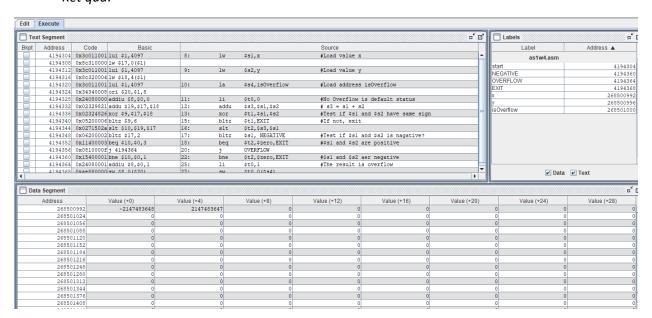
#### Bài 1:

Chương trình:

```
#Laboratory Exercise 4, Home Assignment 1
.data
              .word 0x80000000
x:
               .word 0x7fffffff
у:
isOverflow:
               .word 0
.text
start:
       lw
              $s1,x
                                      #Load value x
       lw
              $s2,y
                                      #Load value y
                                     #Load address isOverflow
       1a
              $s4,isOverflow
                                     #No Overflow is default status
       1i
              $t0,0
              $s3,$s1,$s2
                                     \# s3 = s1 + s2
       addu
       xor
              $t1,$s1,$s2
                                     #Test if $s1 and $s2 have same sign
       bltz
              $t1,EXIT
                                      #If not, exit
       slt
              $t2,$s3,$s1
       bltz
              $s1, NEGATIVE
                                     #Test if $s1 and $s2 is nagative?
              $t2,$zero,EXIT
                                     #$s1 and $s2 are positive
       beq
               #if $s3 > $s1 then the result is not overflow
               OVERFLOW
       j
NEGATIVE:
              $t2,$zero,EXIT
                                     #$s1 and $s2 aer negative
       bne
               \#if \$s3 < \$s1  then the result is not overflow
OVERFLOW:
                                     #The result is overflow
       li
              $t0,1
EXIT:
              $t0,0($s4)
       sw
```

#### TH1: x, y trái dấu

```
#Laboratory Exercise 4, Home Assignment 1
.data
x:
                .word
                        0x80000000
                        0x7fffffff
у:
                .word
isOverflow:
                .word
.text
start:
        lw
                $s1,x
                                         #Load value x
        lw
                                         #Load value y
                $s2,y
        1a
                $s4, is Overflow
                                         #Load address isOverflow
                                         #No Overflow is default status
        1i
                $t0,0
                $s3,$s1,$s2
                                         # s3 = s1 + s2
        addu
                                         #Test if $s1 and $s2 have same sign
        xor
                $t1,$s1,$s2
        bltz
                $t1,EXIT
                                         #If not, exit
        slt
                $t2,$s3,$s1
                $s1, NEGATIVE
                                         #Test if $s1 and $s2 is nagative?
        bltz
        beq
                $t2,$zero,EXIT
                                         #$s1 and $s2 are positive
                #if $s3 > $s1 then the result is not overflow
        j
                OVERFLOW
NEGATIVE:
        bne
                $t2,$zero,EXIT
                                         #$s1 and $s2 aer negative
                #if $s3 < $s1 then the result is not overflow
OVERFLOW:
                $t0,1
                                         #The result is overflow
        1i
EXIT:
        SW
                $t0,0($s4)
```



\$s1	17	-2147483648
\$32	18	2147483647
\$83	19	-1
\$84	20	268501000

### TH2: x, y cùng dương và x+y không tràn

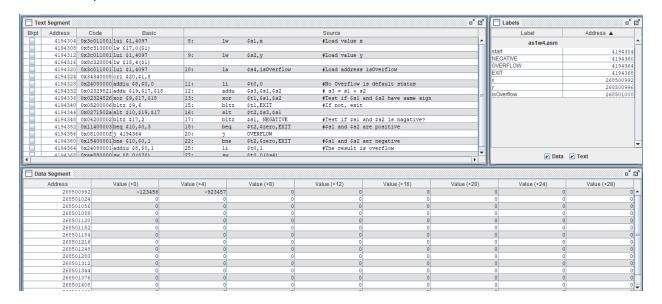
- Chương trình:

```
#Laboratory Exercise 4, Home Assignment 1
.data
x:
                .word
                       123456
                .word
                       123457
у:
isOverflow:
               .word
.text
start:
       lw
               $s1,x
                                       #Load value x
        lw
                                       #Load value y
               $s2,y
               $s4,isOverflow
                                       #Load address isOverflow
        1a
       1i
               $t0,0
                                       #No Overflow is default status
                                        # s3 = s1 + s2
        addu
               $s3,$s1,$s2
               $t1,$s1,$s2
                                       #Test if $s1 and $s2 have same sign
       xor
       bltz
              $t1,EXIT
                                       #If not, exit
               $t2,$s3,$s1
        slt
       bltz
               $s1, NEGATIVE
                                       #Test if $s1 and $s2 is nagative?
        beq
               $t2,$zero,EXIT
                                       #$s1 and $s2 are positive
                #if $s3 > $s1 then the result is not overflow
               OVERFLOW
        j
NEGATIVE:
        bne
               $t2,$zero,EXIT
                                       #$s1 and $s2 aer negative
               #if $s3 < $s1 then the result is not overflow
OVERFLOW:
        li.
               $t0,1
                                       #The result is overflow
EXIT:
               $t0,0($s4)
       sw
```

Tex	t Segment									- □ □	Labels		o <sup>r</sup>
Bkpt	Address C	ode	Basic					Source			Lab	el	Address ▲
	4194304 Ox3			8:	lw	\$sl,x		Load value x		-	.	as1w4.asm	
	4194308 0x8d										start		419430
	4194312 0x3c			9:	lw	\$s2,y		Load value y			NEGATIVE		4194360
	4194316 0x8d										OVERFLOW		4194364
	4194320 0x3c			10:	la	\$s4,isOverflow		Load address	isOverflow		EXIT		4194368
	4194324 0x34										Y		268500992
	4194328 0x24			11:	11	\$t0,0			s default status		v v		268500996
			du \$19,\$17,\$18	12:	addu	\$83,\$81,\$82		s3 = s1 + s2			isOverflow		268501000
	4194336 0x02			13:	xor	\$t1,\$s1,\$s2			nd \$s2 have same si	.gn			
	4194340 0x05			15:	bltz	\$t1,EXIT		If not, exit					
			t \$10,\$19,\$17	16:	slt	\$t2,\$s3,\$s1							
	4194348 0x06			17:	bltz	\$sl, NEGATIVE			nd \$s2 is nagative?				
	4194352 0x1			18:	beq	\$t2,\$zero,EXIT	- 1	\$sl and \$s2 a	re positive				
	4194356 0x08			20:	j	OVERFLOW							
	4194360 0x15			22:	bne	\$t2,\$zero,EXIT		\$sl and \$s2 a					
	4194364 0x2			25:	li	\$t0,1		The result is	overflow				
4	4194368 Oxas	880000192	S8 0/S201	27.	ew.	St0 0(Se4)				)	-[[]	✓ Data	✓ Text
Dat	a Segment										-1		o"
	Address	Va	lue (+0)	Value (+4)		Value (+8)	Val	Je (+12)	Value (+16)	Value (+20)	Valu	e (+24)	Value (+28)
	268500992		123456	12	3457	(	D	0		0	0	0	(
	268501024		0		0	(	0	0		0	0	0	(
	268501056		0		0	(	0	0		0	0	0	(
	268501088		0		0	(	0	0		0	0	0	
	268501120		0		0	(	0	0		0	0	0	1
	268501152		0		0	(	0	0		0	0	0	
	268501184		0		0	(	0	0		0	0	0	
	268501216		0		0	(	0	0		0	0	0	
	268501248		0		0	(	0	0		0	0	0	
	268501280		0		0	(	0	0		0	0	0	
	268501312		0		0	(	0	0		0	0	0	
	268501344		0		0	(	D	0		0	0	0	(
	268501376		0		0	(	0	0		0	0	0	(
	268501408		0		0	(	D	0		0	0	0	(
\$81							17				3456		
\$ <b>s</b> 2							18 19				6913		
\$84							20			26850	1000		

TH3: x, y cùng âm và x+y không tràn

```
#Laboratory Exercise 4, Home Assignment 1
.data
                .word
                        -123456
x:
                 .word
                        -923457
у:
isOverflow:
                .word
.text
start:
        lw
                $s1,x
                                         #Load value x
        lw
                $s2,y
                                         #Load value y
                $s4,isOverflow
                                         #Load address isOverflow
        1a
        1i
                $t0,0
                                         #No Overflow is default status
        addu
                $s3,$s1,$s2
                                         # s3 = s1 + s2
                                         #Test if $s1 and $s2 have same sign
                $t1,$s1,$s2
        xor
        bltz
                $t1,EXIT
                                         #If not, exit
                $t2,$s3,$s1
        slt
                                         #Test if $s1 and $s2 is nagative?
        bltz
                $s1, NEGATIVE
        beq
                $t2,$zero,EXIT
                                         #$s1 and $s2 are positive
                #if $s3 > $s1 then the result is not overflow
                OVERFLOW
        Ť
NEGATIVE:
                $t2,$zero,EXIT
                                         #$s1 and $s2 aer negative
        bne
                #if $s3 < $s1 then the result is not overflow
OVERFLOW:
                $t0,1
                                         #The result is overflow
        li
EXIT:
                $t0,0($s4)
        sw
```

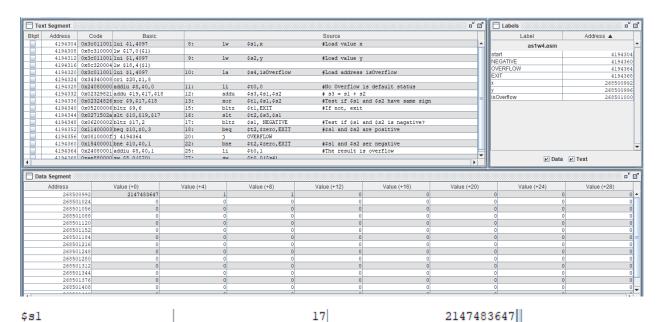


\$s1	17	-123456
\$s2	18	-923457
\$83	19	-1046913
\$84	20	268501000

#### TH4: x, y cùng dương và x+y tràn

- Chương trình:

```
#Laboratory Exercise 4, Home Assignment 1
.data
x:
                .word
                       0x7fffffff
у:
               .word
isOverflow:
               .word
.text
start:
       lw
              $s1,x
                                       #Load value x
              $s2,y
                                       #Load value y
       lw
       1a
               $s4,isOverflow
                                       #Load address isOverflow
                                       #No Overflow is default status
       1i
               $t0,0
              $s3,$s1,$s2
                                       # s3 = s1 + s2
       addu
               $t1,$s1,$s2
                                       #Test if $s1 and $s2 have same sign
       xor
       bltz
              $t1,EXIT
                                       #If not, exit
              $t2,$s3,$s1
       slt
       bltz
              $s1, NEGATIVE
                                       #Test if $s1 and $s2 is nagative?
               $t2,$zero,EXIT
                                       #$s1 and $s2 are positive
       beq
               #if $s3 > $s1 then the result is not overflow
        j
               OVERFLOW
NEGATIVE:
                                       #$s1 and $s2 aer negative
       bne
               $t2,$zero,EXIT
               #if $s3 < $s1 then the result is not overflow
OVERFLOW:
               $t0,1
                                       #The result is overflow
       1i
EXIT:
               $t0,0($s4)
       sw
```



\$s2 18 1 \$s3 19 -2147483648 \$s4 20 268501000

TH5: x, y cùng âm và x+y tràn

```
#Laboratory Exercise 4, Home Assignment 1
.data
                      0x80000000
x:
               .word
               .word -1
у:
isOverflow:
               .word
.text
start:
                                      #Load value x
       lw
              $s1,x
                                     #Load value y
       lw
              $s2,y
                                     #Load address isOverflow
       1a
              $s4,isOverflow
       1i
                                      #No Overflow is default status
              $t0,0
                                      # s3 = s1 + s2
       addu
              $s3,$s1,$s2
                                      #Test if $s1 and $s2 have same sign
       xor
              $t1,$s1,$s2
       bltz
              $t1,EXIT
                                      #If not, exit
              $t2,$s3,$s1
       slt
       bltz
              $s1, NEGATIVE
                                     #Test if $s1 and $s2 is nagative?
              $t2,$zero,EXIT
                                      #$s1 and $s2 are positive
       beq
               #if $s3 > $s1$ then the result is not overflow
               OVERFLOW
       j
NEGATIVE:
               $t2,$zero,EXIT
                                    #$s1 and $s2 aer negative
       bne
               #if $s3 < $s1 then the result is not overflow
OVERFLOW:
              $t0,1
                                     #The result is overflow
       1i
EXIT:
               $t0,0($s4)
       sw
```

kopt	ct Segment					***************************************				Labels	
pt	Address	Code	Basic				Source			Label	Address ▲
Ш		x3c011001 lu		8:	1w	\$sl,x	#Load value x		_	as1w4.asm	
Н		x8c310000 lw								start	4194304
-		x3c011001 lu		9:	lw	\$s2,y	#Load value y			NEGATIVE	4194360
4		x8c320004 lw								OVERFLOW	4194364
+		x3c011001 lu		10:	la	\$s4,isOverflow	#Load address	isOverflow		EXIT	4194368
-		x34340008 or								x	268500992
-		x24080000 ad		11:	li	\$t0,0		is default status		y	268500996
-			du \$19,\$17,\$18	12:	addu	\$83,\$81,\$82	# s3 = s1 + s			isOverflow	268501000
+			r \$9,\$17,\$18	13:	xor	\$t1,\$s1,\$s2		and \$s2 have same sign			
-		x05200006 bl		15:	bltz	\$t1,EXIT	#If not, exit	:			
-		x0271502a s1 x06200002 b1	t \$10,\$19,\$17	17:	slt bltz	\$t2,\$s3,\$s1	Im				
-		x11400003 be		18:	bea	\$s1, NEGATIVE \$t2.\$zero.EXIT	#108t 11 \$81 #\$81 and \$82	and \$s2 is nagative?			
+				20:	peq	St2, Szero, EXII	#\$SI and \$SZ	are positive			
+		x0810000f j x15400001 bn		20:	bne	\$t2,\$zero,EXIT	#\$sl and \$s2				
-		x24080001 ad		22:	li	\$t2,\$Zero,EXII	#\$SI and \$S2 #The result i				
							Fine leading 1	s overilow			
		VARRENOON SW		27.	en.	S+0 0(S94)	Fine resurt 1	s overiiow	V	✓ Data	✓ Text
	4194368 Ox						Fine result 1	s overilow	P	✓ Data	
	4194368 0x	vae880000 qu	58 0 (520)	27.		(1p2)() ()†2					
	4194368 0x ta Segment Address	Va Va	ss 0 (s20)		eu.		Value (+12)	Value (+16)	Value (+20)	✓ Data  Value (+24)	
	4194368 0x ta Segment Address 26850099	Vae880000 sw	58 0 (520)	27.		(1p2)() ()†2					
	A194368 0x ta Segment Address 26850099 26850102	Vae8800000 9W Va 192 124	ss 0 (s20)	27.	eu.	(1,p2) () (1+2					
	4194368 0x  ta Segment  Address 26850099 26850102	Va V	ss 0 (s20)	27.	eu.	(1,p2) () (1+2					
	A194368 0x  ta Segment  Address  26850102 26850105 26850108	Va Va 192 124 156	ss 0 (s20)	27.	eu.	(1,p2) () (1+2					
	A194368 0x  Address 26850109 26850102 26850102 26850112	Va 922 924 956 88	ss 0 (s20)	27.	eu.	(1,p2) () (1+2					
	A194368 0x  ta Segment  Address  26850102 26850105 26850112 26850112	Va 992 924 556 988 920 552	ss 0 (s20)	27.	eu.	(1,p2) () (1+2					
	A194368 Nv ta Segment  Address 26850099 26850102 26850102 26850112 26850118	Va V	ss 0 (s20)	27.	eu.	(1,p2) () (1+2					
	A194368 0v  ta Segment  Address	Va 92 Va 92 24 556 88 20 20 52 88 4 116	ss 0 (s20)	27.	eu.	(1,p2) () (1+2					
	A194368 0x  Address 26850102 26850102 26850112 26850112 2685012 2685012 2685012	Va 192 194 195 196 188 120 152 188 148	ss 0 (s20)	27.	eu.	(1,p2) () (1+2					
	A194368 0x  Ita Segment  Address  26850192 26850102 26850112 26850112 26850112 26850122 26850122 26850122	Va 1992 124 1556 188 1552 184 116 148 180	ss 0 (s20)	27.	eu.	(1,p2) () (1+2					
	A194368 0v  Address 26850195 26850106 268501106 26850112 26850124 26850124 26850124 26850124	Va 1922 124 156 188 120 151 164 164 188 116	ss 0 (s20)	27.	eu.	(1,p2) () (1+2					
	A194368 0v  Address  26850099 26850100 26850110 26850111 26850112 26850122 26850132 26850132	Va 1992 1224 156 188 220 552 84 116 148 180 112	ss 0 (s20)	27.	eu.	(1p2)() ()†2					
	A194368 0v  Address 26850195 26850106 268501106 26850112 26850124 26850124 26850124 26850124	Va 992 124 156 188 20 151 161 161 161 176	ss 0 (s20)	27.	eu.	(1p2)() ()†2					

\$sl	17	-2147483648
\$82	18	-1
\$83	19	2147483647
\$84	20	268501000

#### Bài 2:

### a, Extract MSB of \$s0

- Chương trình:

- Kết quả:

Те	Text Segment										
Bkpt	Address	Code	Basic				Source	П			
	4194304	0x3c012019	lui \$1,0x00002019	3:	li	\$s0,0x20194543	#load test value for these function	_			
	4194308	0x34304543	ori \$16,\$1,0x00004543	3							
	4194312	0x3c01ff00	lui \$1,0xffffff00	4:	andi	\$t0, \$s0, 0xff000000	#Extract MSB of \$s0				
	4194316	0x34210000	ori \$1,\$1,0x00000000								
	4194320	0x02014024	and \$8,\$16,\$1								

\$t0	8	0x20000000	

#### b, Clear LSB of \$s0

Chương trình:

- Kết quả:

Bkpt	Address	Code	Basic				Source
	4194304	0x3c0120191	lui \$1,0x00002019	3:	li	\$s0,0x20194543	#load test value for these function
	4194308	0x34304543	ri \$16,\$1,0x00004543				
	4194312	0x3c01ffff1	lui \$1,0xffffffff	4:	andi	\$t0, \$s0, 0xffffff00	#Clear LSB of \$s0
	4194316	0x3421ff00 c	ri \$1,\$1,0x0000ff00				
	4194320	0x02014024 a	and \$8,\$16,\$1				

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

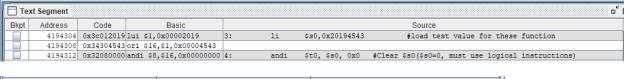
```
#Laboratory Exercise 4, Home Assignment 2
.text
    li $$50,0x20194543  #load test value for these function
    ori $$t0, $$s0, 0x000000ff #Set LSB of $$s0(bits 7 to 0 are set to 1
```

Tex	Text Segment									
Bkpt	Address	Code		Basic				Source		
	4194304	0x3c012019	lui \$1	,0x00002019	3:	li	\$s0,0x20194543	#load test value for these function		
	4194308	0x34304543	ori \$1	6,\$1,0x00004543						
	4194312	0x360800ff	ori \$8	,\$16,0x000000ff	4:	ori	\$t0, \$s0, 0x000000f	f #Set LSB of \$s0(bits 7 to 0 are set to 1		
\$t0						8		0x201945ff		

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

- Chương trình:

- Kết quả:



## \$t0 8 0x00000000

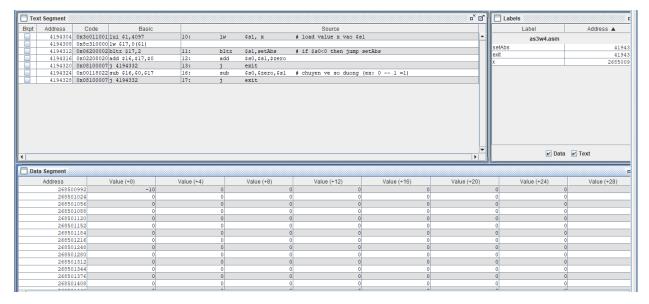
#### Bài 3:

### a, abs \$s0,\$s1 \$s0<= |\$s1|

Chương trình (\$s1 <0):</li>

```
.data
        .word
                 -10
\mathbf{x}:
.text
                 $s1, x
        lw
                                  # load value x vao $s1
                                  # if $s0<0 then jump setAbs
        bltz
                 $s1,setAbs
        add
                 $s0,$s1,$zero
        j
                 exit
setAbs:
                 $s0,$zero,$s1  # chuyen ve so duong (ex: 0 -- 1 =1)
        sub
                 exit
exit:
```

- Kết quả (\$s1<0):

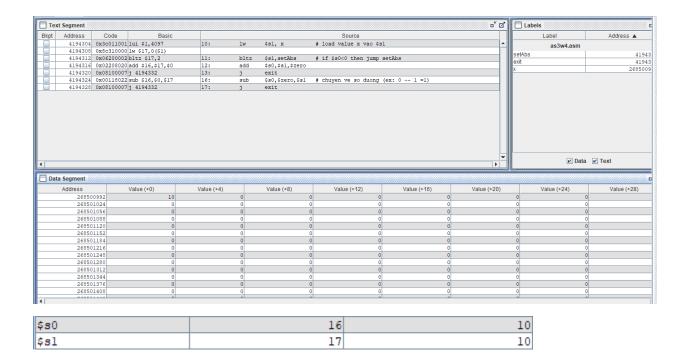


\$80	16	10
\$sl	17	-10

- Chương trình (\$s1>0):

```
.data
x:
      .word 10
.text
      lw $s1, x
                          # load value x vao $s1
            $s1,setAbs
                          # if $s0<0 then jump setAbs
      bltz
       add
             $s0,$s1,$zero
       j
             exit
setAbs:
       sub
            $s0,$zero,$s1  # chuyen ve so duong (ex: 0 -- 1 =1)
       j
            exit
exit:
```

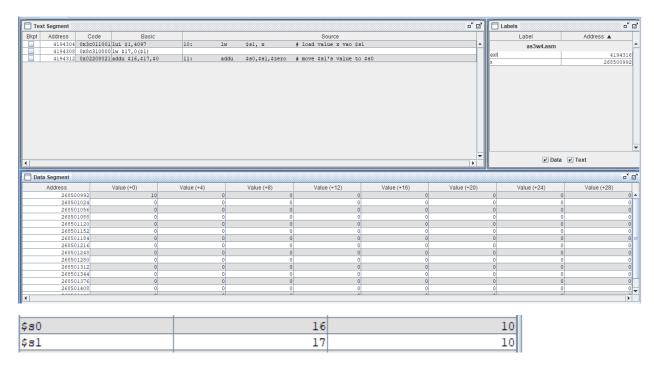
- Kết quả (\$s1>0):



b, move \$s0,\$s1 \$s0 <=\$s1

- Chương trình:

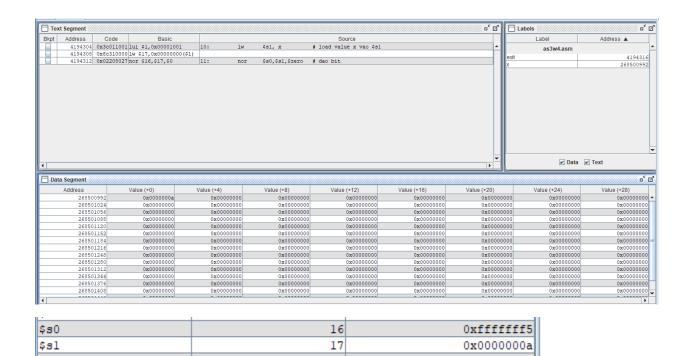
```
.data
x: .word 10
.text
lw $s1, x # load value x vao $s1
addu $s0,$s1,$zero # move $s1's value to $s0
exit:
```



c, not \$s0, \$s1 \$s0 <= bit invert (\$s1)

Chương trình:

```
.data
x: .word 10
.text
lw $$s1, x # load value x vao $$s1
nor $$s0,$$s1,$zero # dao bit
exit:
```



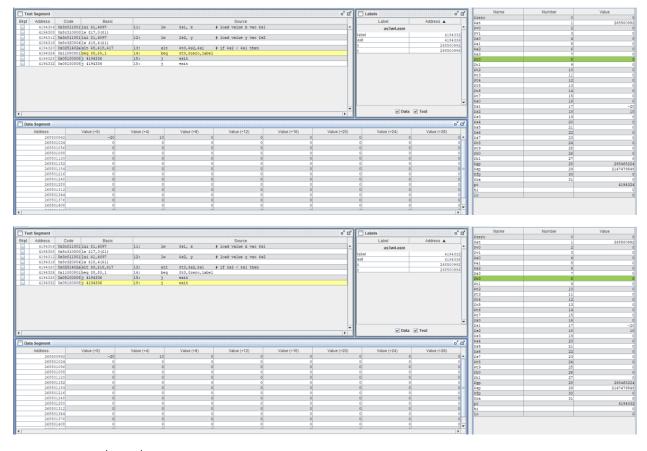
#### d, ble \$s1,\$s2,label if (\$s1 <=\$s2) i label

Chương trình:+ TH1: \$s1 < \$s2</li>

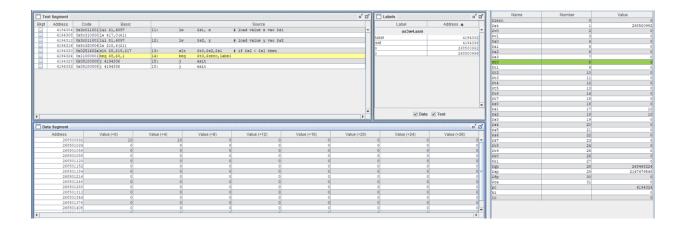
```
.data
       .word -20
\mathbf{x}:
               10
       .word
у:
.text
       lw
           $s1, x
                              # load value x vao $s1
               $s2, y
       lw
                              # load value y vao $s2
               $t0,$s2,$s1 # if $s2 < $s1$ then
        slt
               $t0,$zero,label
       beq
        j
               exit
label:
        j
               exit
exit:
```

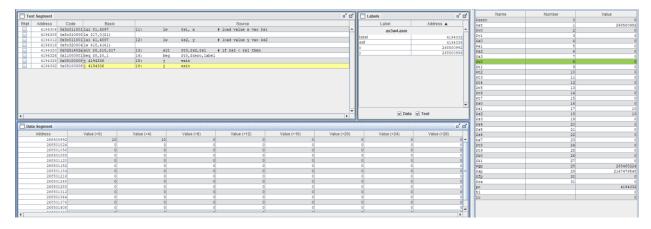
+ TH2: \$s1 = \$s2

```
.data
x: .word 10
у:
      .word
             10
.text
      lw
             $s1, x # load value x vao $s1
             $s2, y
      lw
                          # load value y vao $s2
             $t0,$s2,$s1 # if $s2 < $s1 then
      slt
             $t0,$zero,label
      beq
      j
             exit
label:
      j
         exit
exit:
   + TH3: $s1 > $s2
.data
      .word
             20
x:
у:
      .word
             10
.text
      lw $s1, x # load value x vao $s1
             $s2, y
                           # load value y vao $s2
       lw
             $t0,$s2,$s1 # if $s2 < $s1 then
       slt
             $t0,$zero,label
       beq
             exit
       j
label:
       j
            exit
exit:
 - Kết quả:
 - + TH1: $s1 < $s2
```

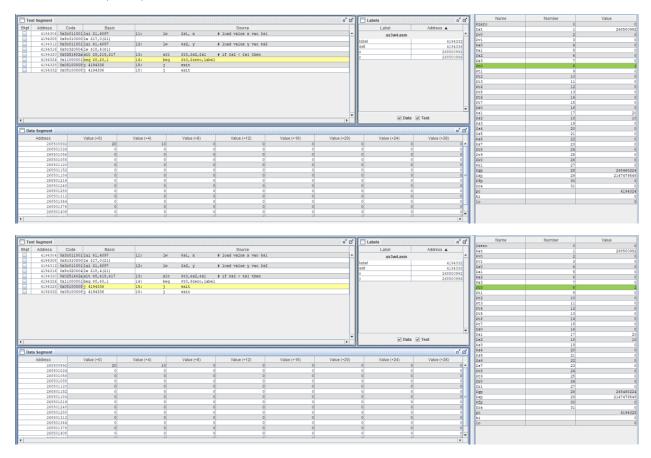


- + TH2: \$s1 = \$s2





+ TH3: \$s1 > \$s2

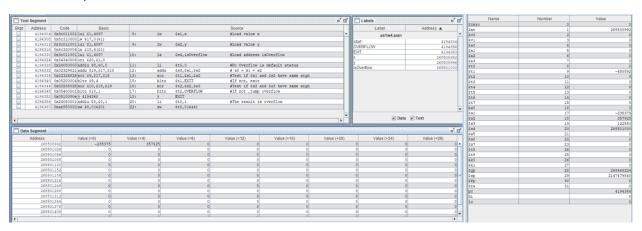


Bài 4:

```
#Laboratory Exercise 4, Home Assignment 1
.data
             .word -235375
x:
y:
             .word 357925
isOverflow: .word 0
.text
start:
                                  #Load value x
      lw $s1,x
                                  #Load value y
      lw
            $s2,y
            $s4,isOverflow
                                 #Load address isOverflow
       la
      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 same sign
      bltz $t1,EXIT
                                  #If not, exit
       xor $t2,$s2,$s3
                                  #Test if $s2 and $s3 have same sign
                                  #If not ,jump overflow
      bltz $t2,0VERFLOW
       j
            EXIT
OVERFLOW:
                                  #The result is overflow
      li
            $t0,1
EXIT:
      sw $t0,0($s4)
```

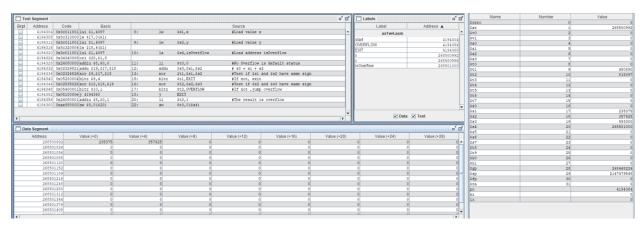
### TH1: x,y trái dấu

```
#Laboratory Exercise 4, Home Assignment 1
.data
                       -235375
x:
               .word
у:
                .word
                       357925
isOverflow:
               .word
.text
start:
               $s1,x
                                        #Load value x
        lw
        lw
               $s2,y
                                       #Load value y
               $s4,isOverflow
                                        #Load address isOverflow
        la
        li
               $t0,0
                                       #No Overflow is default status
               $s3,$s1,$s2
                                        # s3 = s1 + s2
        addu
               $t1,$s1,$s2
                                        #Test if $s1 and $s2 have same sign
       xor
                                       #If not, exit
       bltz
               $t1,EXIT
                                        #Test if $s2 and $s3 have same sign
               $t2,$s2,$s3
        xor
       bltz
               $t2,OVERFLOW
                                        #If not ,jump overflow
               EXIT
        j
OVERFLOW:
                                        #The result is overflow
                $t0,1
        li.
EXIT:
                $t0,0($s4)
```



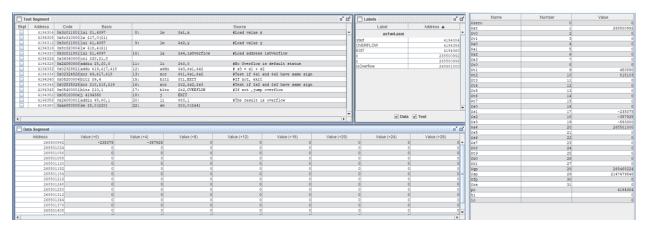
TH2: x,y cùng dương, không tràn

```
#Laboratory Exercise 4, Home Assignment 1
.data
x:
                .word
                        235375
у:
                .word
                        357925
isOverflow:
                .word
.text
start:
        lw
                $s1,x
                                         #Load value x
        lw
                $s2,y
                                         #Load value y
                $s4,isOverflow
                                         #Load address isOverflow
        la
                $t0,0
                                         #No Overflow is default status
        li
                $s3,$s1,$s2
                                         # s3 = s1 + s2
        addu
        xor
                $t1,$s1,$s2
                                         #Test if $s1 and $s2 have same sign
        bltz
                $t1,EXIT
                                         #If not, exit
                                         #Test if $s2 and $s3 have same sign
                $t2,$s2,$s3
        xor
                $t2, OVERFLOW
                                         #If not ,jump overflow
        bltz
        j
                EXIT
OVERFLOW:
                                         #The result is overflow
        li.
                $t0,1
EXIT:
        sw
                $t0,0($s4)
```



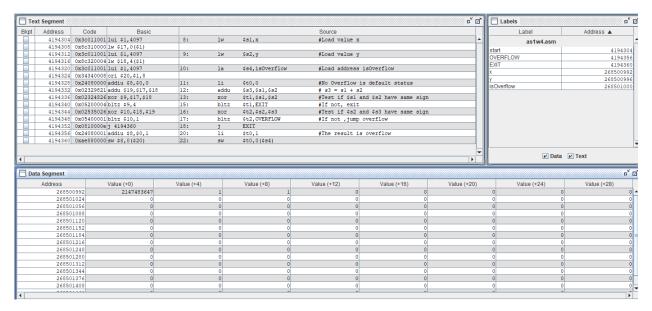
#### TH3: x,y cùng âm, không tràn

```
#Laboratory Exercise 4, Home Assignment 1
.data
                        -235375
x:
                .word
                        -357925
у:
                .word
isOverflow:
                .word
                        0
.text
start:
        lw
                $s1,x
                                        #Load value x
       lw
                $s2,y
                                        #Load value y
        1a
               $s4,isOverflow
                                        #Load address isOverflow
        li.
               $t0,0
                                        #No Overflow is default status
               $s3,$s1,$s2
                                        \# s3 = s1 + s2
        addu
               $t1,$s1,$s2
                                        #Test if $s1 and $s2 have same sign
        xor
       bltz
               $t1,EXIT
                                        #If not, exit
        xor
               $t2,$s2,$s3
                                        #Test if $s2 and $s3 have same sign
               $t2,OVERFLOW
                                        #If not ,jump overflow
       bltz
                EXIT
OVERFLOW:
                                        #The result is overflow
       1i
                $t0,1
EXIT:
                $t0,0($s4)
        sw
```



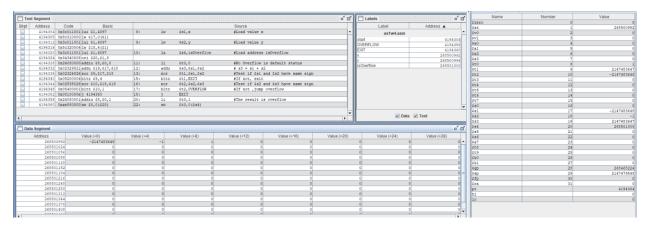
## TH4: x,y cùng dương, tràn

```
#Laboratory Exercise 4, Home Assignment 1
.data
х:
                .word
                         0x7fffffff
у:
                .word
isOverflow:
                .word
                         0
.text
start:
        lw
                $s1,x
                                         #Load value x
        lw
                $s2,y
                                         #Load value y
                $s4,isOverflow
                                         #Load address isOverflow
        la
        li.
                $t0,0
                                         #No Overflow is default status
        addu
                $s3,$s1,$s2
                                         # s3 = s1 + s2
                $t1,$s1,$s2
                                         #Test if $s1 and $s2 have same sign
        xor
        bltz
                $t1,EXIT
                                         #If not, exit
        xor
                $t2,$s2,$s3
                                         #Test if $s2 and $s3 have same sign
        bltz
                $t2,OVERFLOW
                                         #If not , jump overflow
        i
                EXIT
OVERFLOW:
                                         #The result is overflow
                $t0,1
EXIT:
                $t0,0($s4)
```



#### TH5: x,y cùng âm, tràn

```
#Laboratory Exercise 4, Home Assignment 1
.data
x:
                .word
                        0x80000000
                        -1
y:
                .word
                        0
isOverflow:
                .word
.text
start:
                                        #Load value x
        lw
                $s1,x
                                        #Load value y
        lw
                $s2,y
        1a
                $s4,isOverflow
                                        #Load address isOverflow
                                        #No Overflow is default status
        li.
                $t0,0
                                        # s3 = s1 + s2
        addu
                $s3,$s1,$s2
                                        #Test if $s1 and $s2 have same sign
        xor
                $t1,$s1,$s2
        bltz
                $t1,EXIT
                                        #If not, exit
                $t2,$s2,$s3
                                        #Test if $s2 and $s3 have same sign
        xor
                $t2, OVERFLOW
                                        #If not ,jump overflow
        bltz
                EXIT
OVERFLOW:
                $t0,1
                                       #The result is overflow
        1i
EXIT:
                $t0,0($s4)
        sw
```



### Bài 5:

```
#Laboratory Exercise 5
.data
\mathbf{x}:
       .word
              12
                   #x=12
       .word
              4
                     #v = 4
у:
       .word
i:
                     # bien chay i=0
             0
                     # tich sum=0
sum:
       .word 0
.text
       lw
             $s0,x # load value x
       lw
              $s1,y # load value y
       lw
              $s2,i # load value i
       la
             $s3,sum # load address sum
       li 
            $s4,1 # load 1 to $s4
100p:
             $s1,$s4,endloop # if y=1 then endloop
       beq
             $s1,$s1,1
       srl
                        # y=y:2
       addi $s2,$s2,1 # i=i+1
       j.
              loop
endloop:
       sllv
             $s5,$s0,$s2 # sum=x*y
              $s5,0($s3) # return sum
       sw
```

