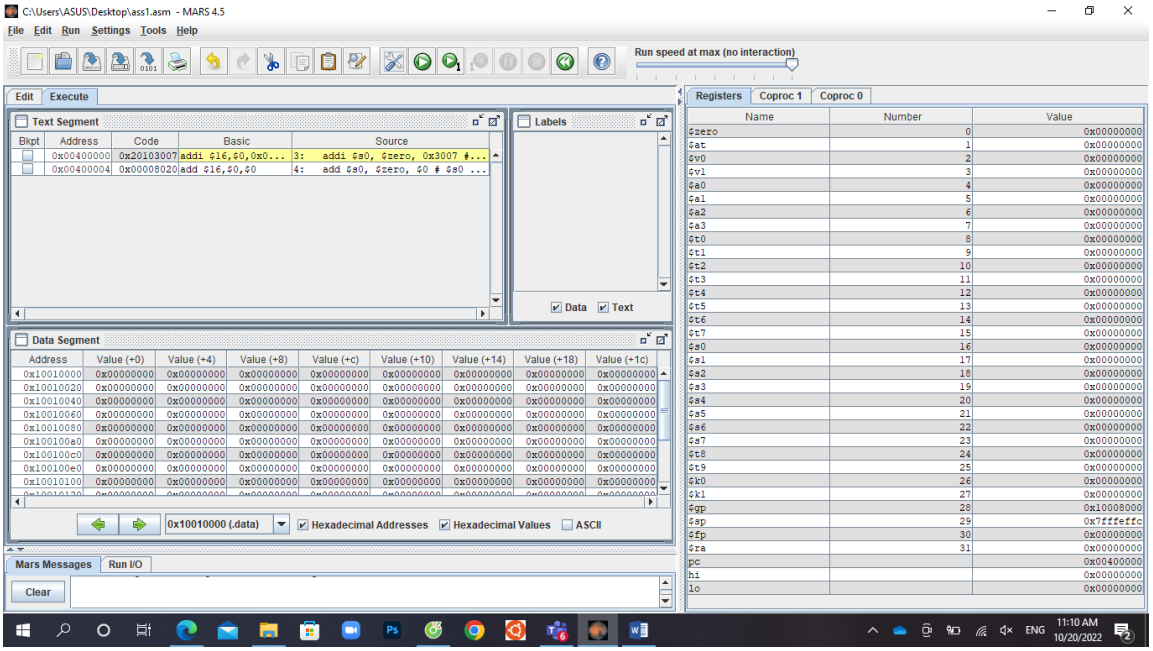
Assignment 1:

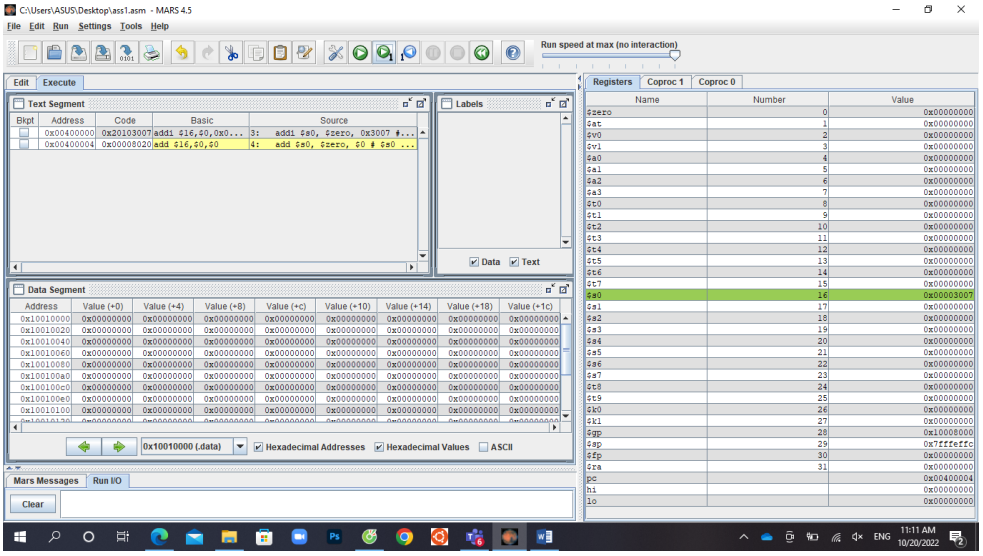
Start:



$s0	16	0x00000000

pc		0x00400000

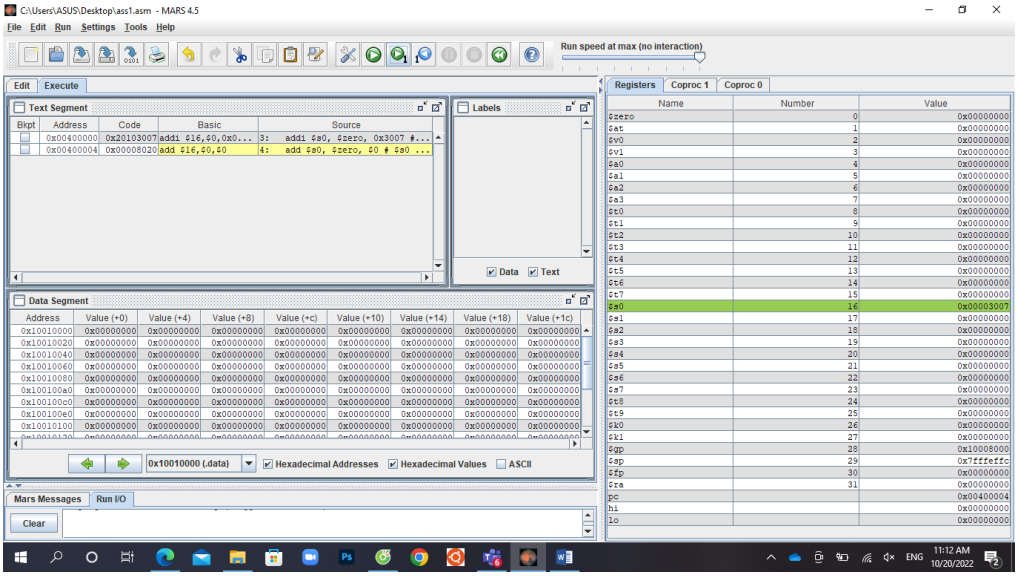
Step 1:



$s0	16	0x00003007

pc		0x00400004

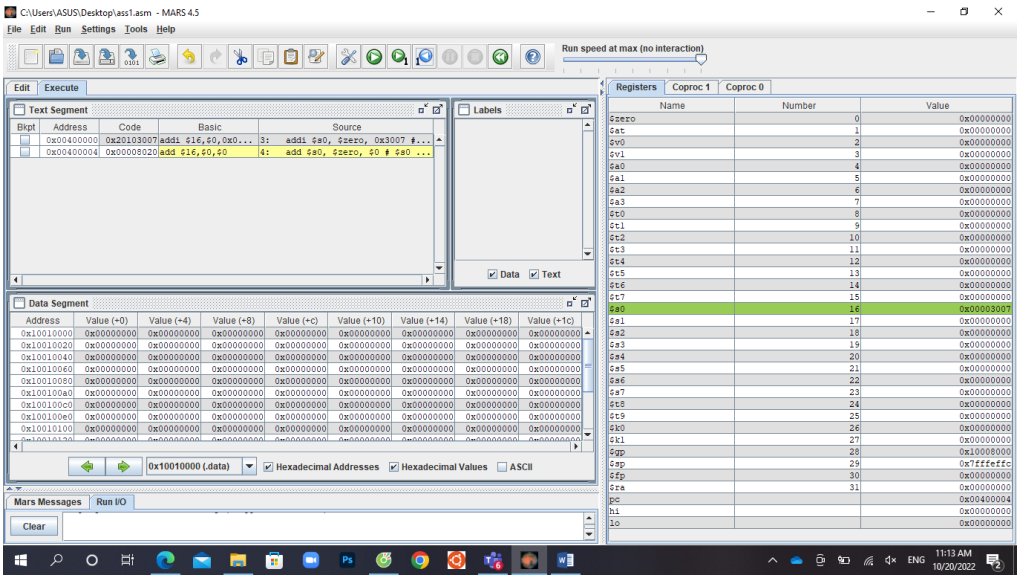
Step 2:



$s0	16	0x00000000

pc	0x00400008

Step 3:



$s0	16	0x00000000

pc	0x00400008

- Ở cửa sổ Text Segment:

Lệnh đầu tiên là 0x20103007: 0010 0000 0001 0000 0011 0000 0000 0111

001000 00000 10000 0011000000000111

Op: 8 (I)(addi)

Rs: 0 ($zero)

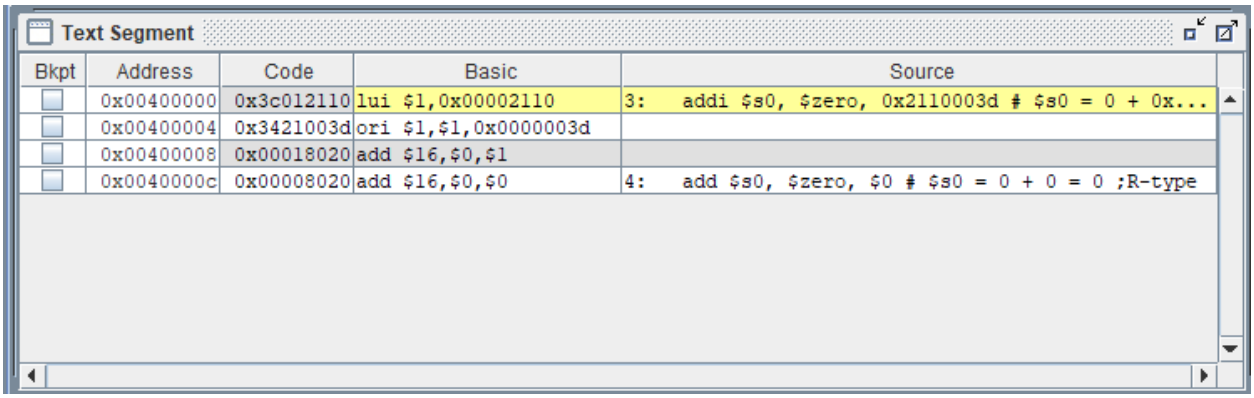
Rt: 16 ($s0)

Imm: 12295

Đúng với: addi $s0, $zero, 0x3007

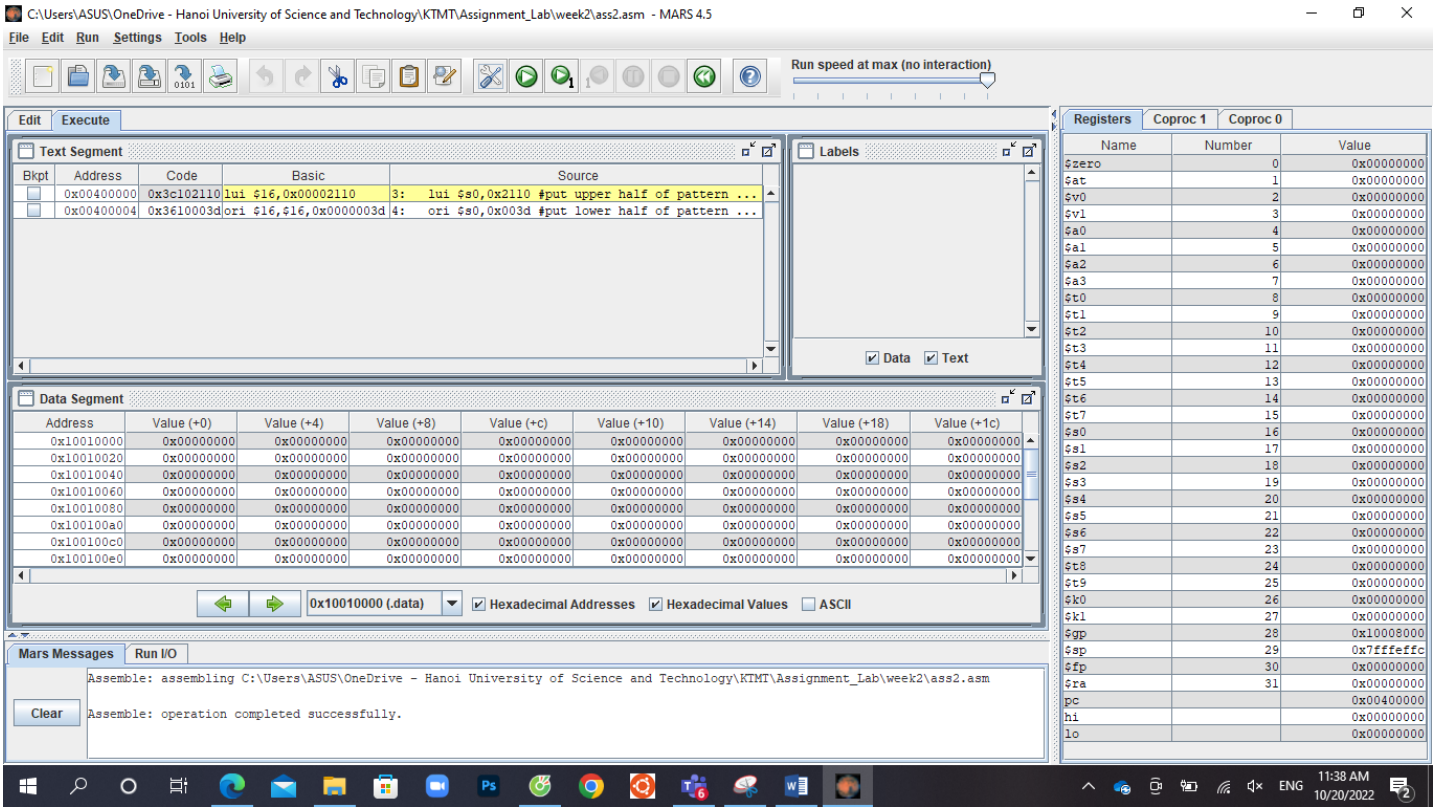
- Khi sửa lại lệnh lui:

Sau khi sửa lệnh addi $s0, $zero, 0x3007 thành addi $s0,$zero, 0x2110003d có nghĩa là chúng ta đang đòi hỏi việc cộng s0 với số 32bit. Tuy nhiên, lệnh addi chỉ phục vụ cộng số 16bit do đó sau khi chạy chương trình hiện lên như sau:



Assignment 2:

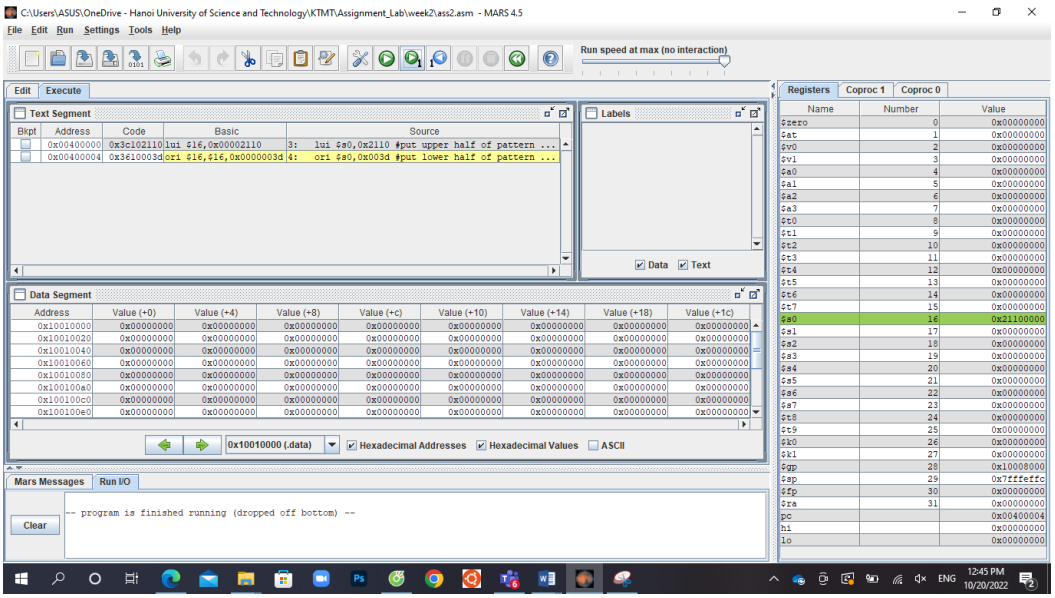
Start

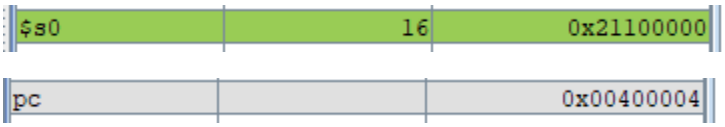


$s0	16	0x00000000

pc		0x00400000

Step 1:



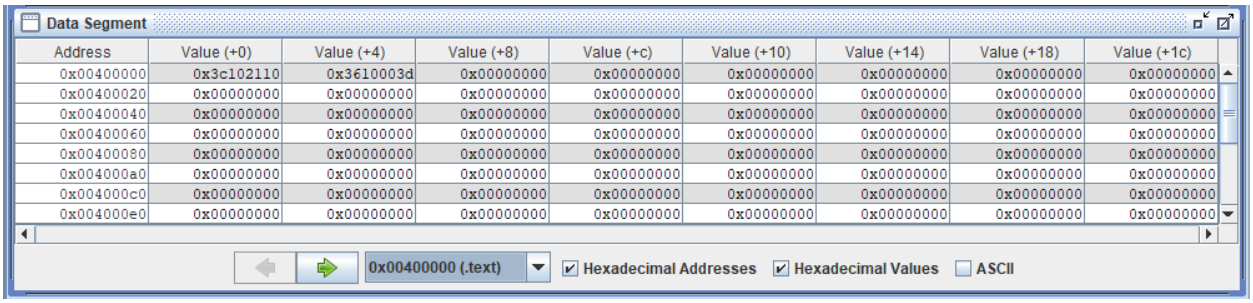


Step 2:

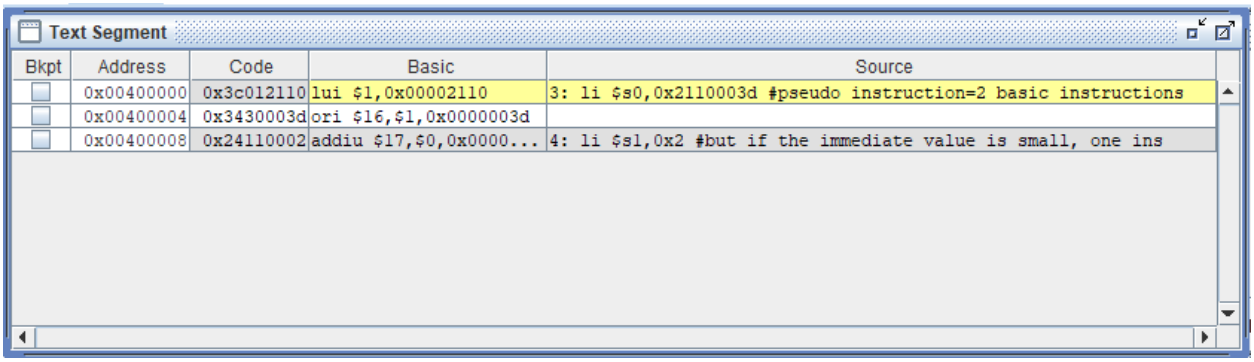
Step 3:

2 lệnh lui và ori sẽ giúp cộng một số 32 bit dần dần vào biến $s0 bằng cách cộng 16bit đầu rồi cộng tiếp 16 bit cuối.

- Ở cửa sổ Data Segment, sự thay đổi của các byte trong cung lệnh .text giống với cột Code trong cửa sổ Text Segment.



Assignment 3:

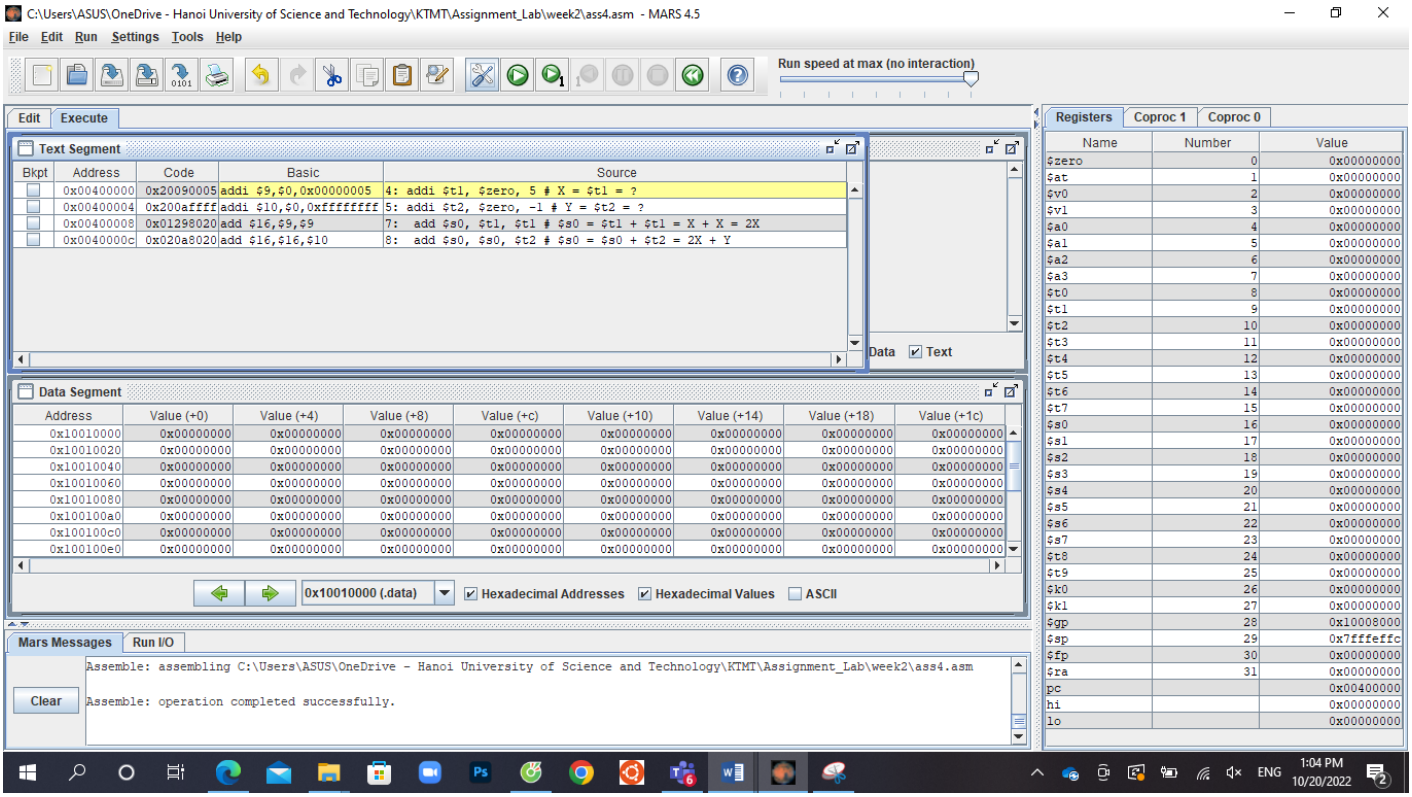


Li là lệnh gán ngay lập tức 1 số 16bit hoặc 32 bit vào biến .

Ở câu lệnh li đầu tiên, lệnh li muốn gán số 32bit 0x2110003d vào biến $s0 nên đã tách ra làm 2 lệnh là lui và ori.

Còn lệnh thứ 2 là li $s1,0x2 là chỉ gán 1 số <32bit nên nó sẽ gán luôn và tương đương với addiu $s1, $s0,0x2

Assignment 4:



Khi debug, gán $t1 = 5, $t2 = -1, thì kết quả là

- Ở lần add đầu tiên, tương ứng với 10 và lần add thứ hai là tương ứng với 9
- => Kết quả đúng: 5 * 2 – 1 = 9

Addi $t2, $zero, -1

0x200affff : 0010 0000 0000 1010 1111 1111 1111 1111

001000 00000 01010 1111111111111111

Op: addi

Rs: $0

Rt: 10 ($t2)

Imm: -1

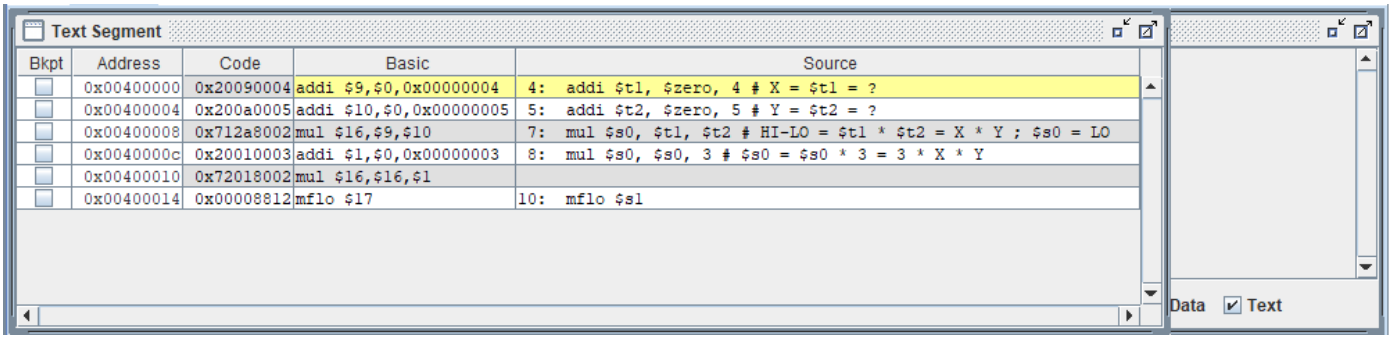
Trùng với khuôn mẫu kiểu lệnh I

Add $s0, $t1, $t1 : 0x01298020

0000 0001 0010 1001 1000 0000 0010 0000
000000 01001 01001 10000 00000 100000
Op: R
Rs: 9 ($t1)
Rt: 9 ($t1)
Rd: 16 ($s0)
Shift amount: 0
Function: 32 (add)

Trùng với khuôn mẫu kiểu lệnh R

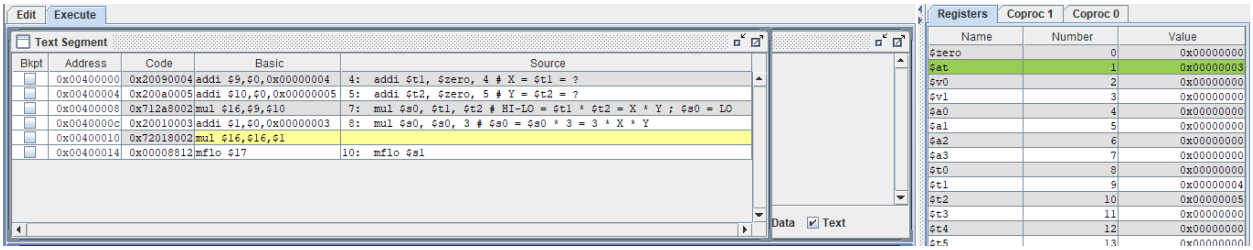
Assignment 5:



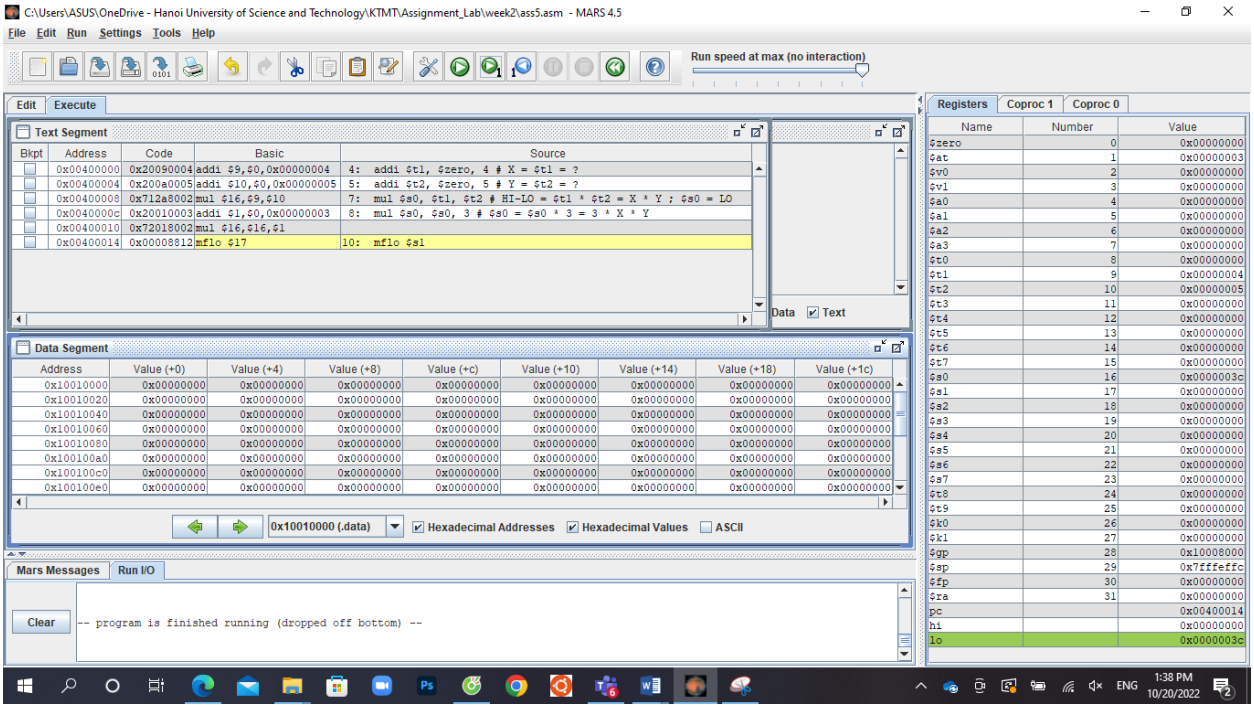
2 lệnh addi để gán giá trị cho 2 biến t1 và t2

Thao tác nhân có kết quả chứa hai thanh ghi HI <lưu 32 bit sau> và LO<lưu 32 bit đầu>

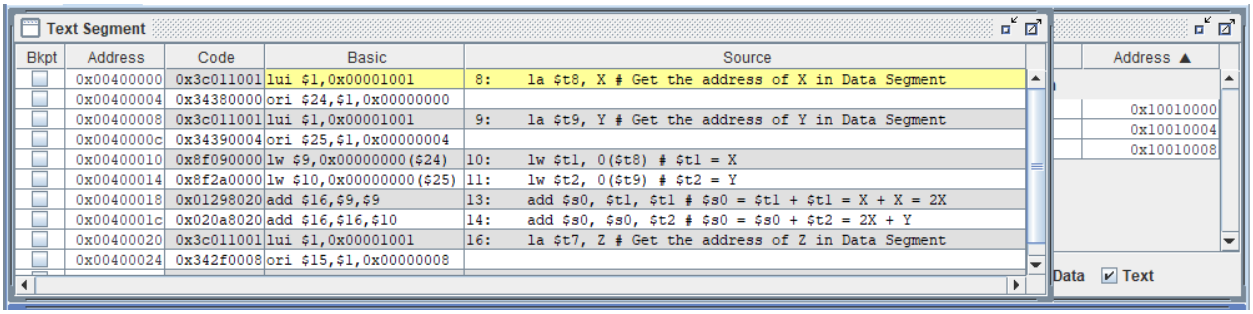
Điều bất thường là trong lệnh mul, 3 không phải là 1 biến nên khi debug xuất hiện biến at lưu giá trị của 3



Lệnh mflo giúp sao lưu giá trị của biến s0 sang biến s1.



Assignment 6:



- Lệnh la được biên dịch thành 2 lệnh là lui và ori nhằm lưu địa chỉ của các biến X,Y vào biến t8,t9.

- Giá trị của X và Y tương ứng vs giá trị khởi tạo, được hiện thị trong Data Segment lần lượt là 0x00000005 và 0xffffffff

- Ở cửa sổ Register,

Câu lệnh lw lưu giá trị của biến X vào biến t1 bằng việc trỏ tới địa chỉ của X qua 0($t8).

Câu lệnh sw lưu giá trị của thanh ghi s0 vào biến có địa chỉ t7 (là Z).