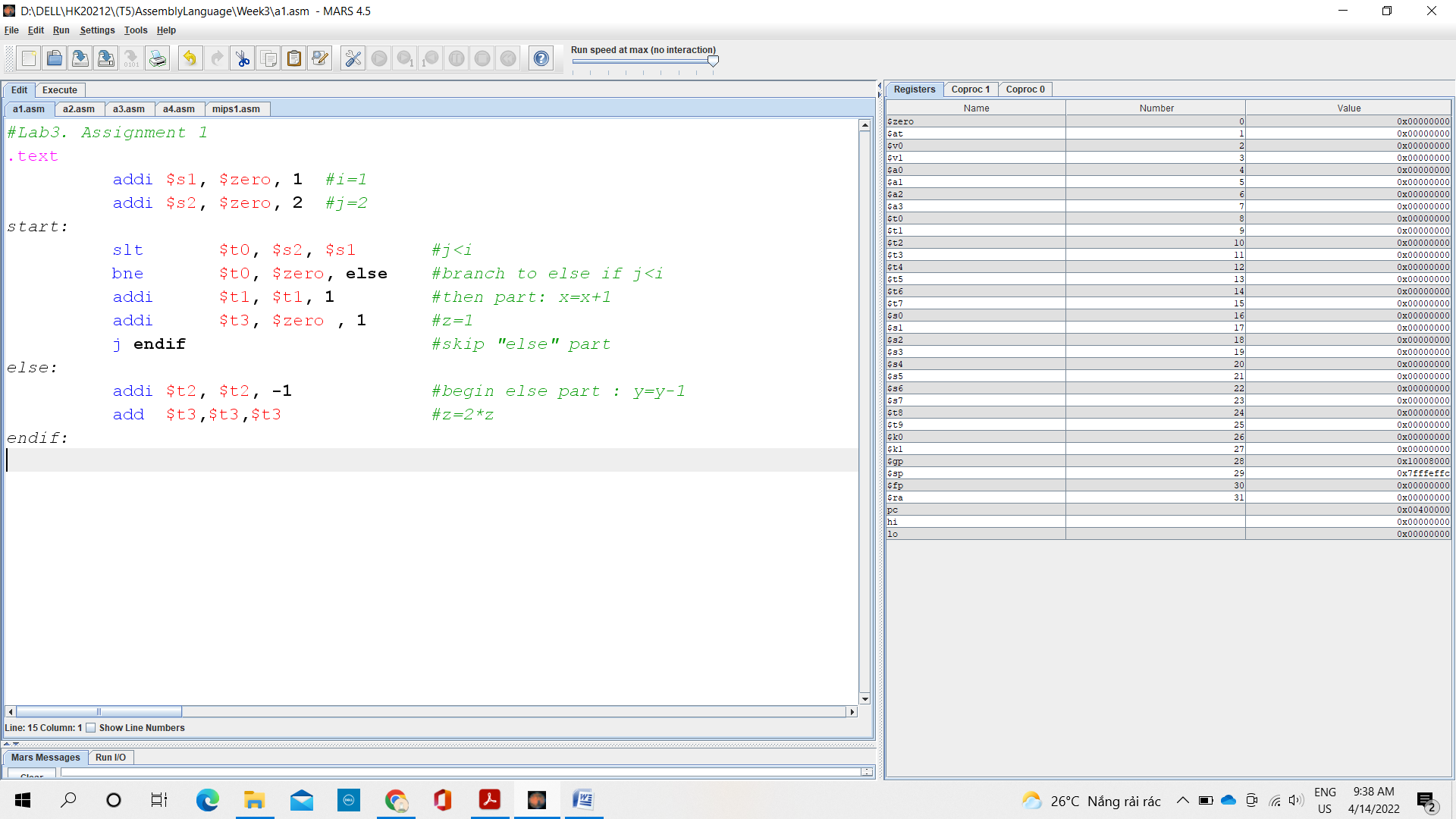
Assignment 1:



The change of registers:

* $s1 : 0x00000000 => 0x00000001 ( i=1)

pc : 0x00400000 => 0x00400004 (addi line 4)

* $s2: 0x00000000 => 0x00000002 (j=2)

pc: 0x00400000 (addi line 4) => 0x00400008(slt line 6)

* $t0: 0x00000000 doesn’t change because $s2=2 >$s1=1

pc: 0x00400008 (slt line 6) => 0x0040000c(bne line 7)

* $pc: 0x0040000c (bne line 7) => 0x00400010 (addi line 8) (because $t0 =0)
* $t1: 0x00000000 => 0x00000001

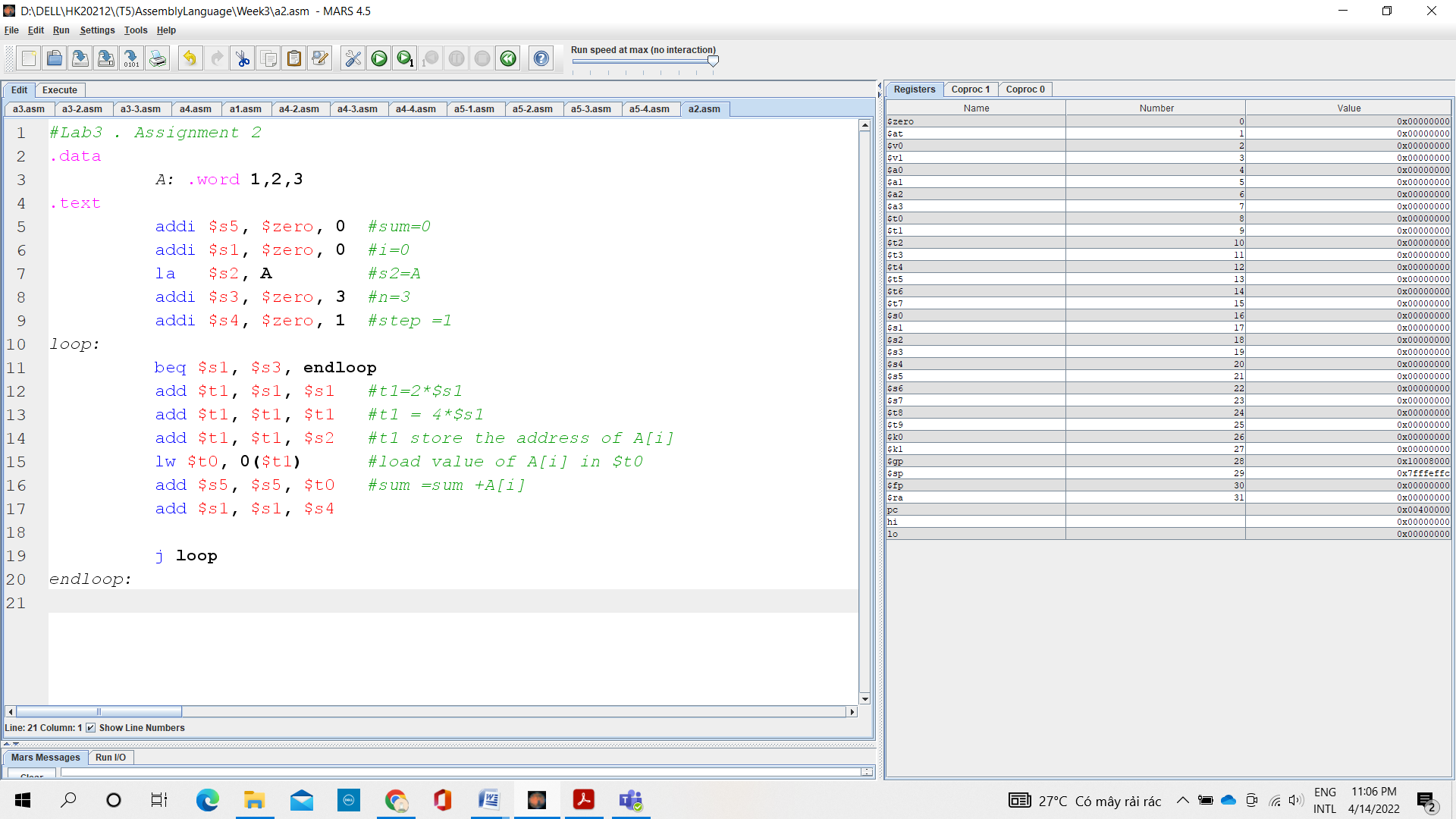
pc: 0x00400010 (addi line 8) => 0x00400014(addi line 9)

* $t3: 0x00000000 => 0x00000001

pc: 0x00400014 (addi line 9) => 0x00400018(j line 10)

* pc : 0x00400018 (j line 10)=> 0x00400024 (endif(14))

Assignment 2:



The change of registers:

* $s5 : 0x00000000 doesn’t change (=0)

pc : 0x00400000(addi(5)) => 0x00400004 (addi(6))

* $s1: 0x00000000 doesn’t change( =0)

pc: 0x00400004 (addi (6)) => 0x00400008(la (7))

* $at: 0x00000000 => 0x10010000 ( because “lui” of “la(7))

pc: 0x00400008 (lui) => 0x0040000c(ori)

* $s2: 0x00000000 => 0x10010000 (address of A )

pc: 0x0040000c(ori) => 0x00400010 (addi(8))

* $s3: 0x00000000 => 0x00000003

pc: 0x00400010 (addi 8) => 0x00400014(addi line 9)

* $s4: 0x00000000 => 0x00000001 (since $1<$3)

pc: 0x00400014 (addi line 9) => 0x00400018(loop: slt(11))

* pc: 0x00400018 (slt 11)=> 0x0040001c (beq(12))
* $t1:0x00000000

pc: 0x0040001c (beq 12)=> 0x00400020 (add(13))

* pc: 0x00400020 (add 13)=> 0x00400024 (add(14))
* $t1: 0x10010000

pc: 0x00400024 (add 14)=> 0x00400028 (add(15))

* $t0:0x00000001

pc: 0x00400028 (add 15)=> 0x0040002c (lw(16))

* $s5: 0x00000001

pc: 0x0040002c => 0x00400030

* $s1:0x00000001

pc: 0x00400030 => 0x00400034

* pc:0x00400034 => 0x00400018
* pc: 0x00400018 => 0x0040001c
* $t1: 0x00000002

pc: 0x0040001c => 0x00400020

* $t1:0x00000004

pc: 0x00400020 => 0x00400024

* $t1: 0x10010004

pc: 0x00400024 => 0x00400028

* $t0:0x00000002

pc: 0x00400028 => 0x0040002c

* $s5:0x00000003

pc: 0x0040002c => 0x00400030

* $s1:0x00000002

pc :0x00400030 =>0x00400034

* pc:0x00400034 => 0x00400018
* pc:0x00400018 => 0x0040001c
* $t1:0x00000004

pc: 0x0040001c=> 0x00400020

* $t1: 0x00000008

pc: 0x00400020=> 0x00400024

* $t1: 0x10010008

pc: 0x00400024 => 0x00400028

* $t0: 0x00000003

pc: 0x00400028 => 0x0040002c

* $s5: 0x00000006

pc: 0x0040002c => 0x00400030

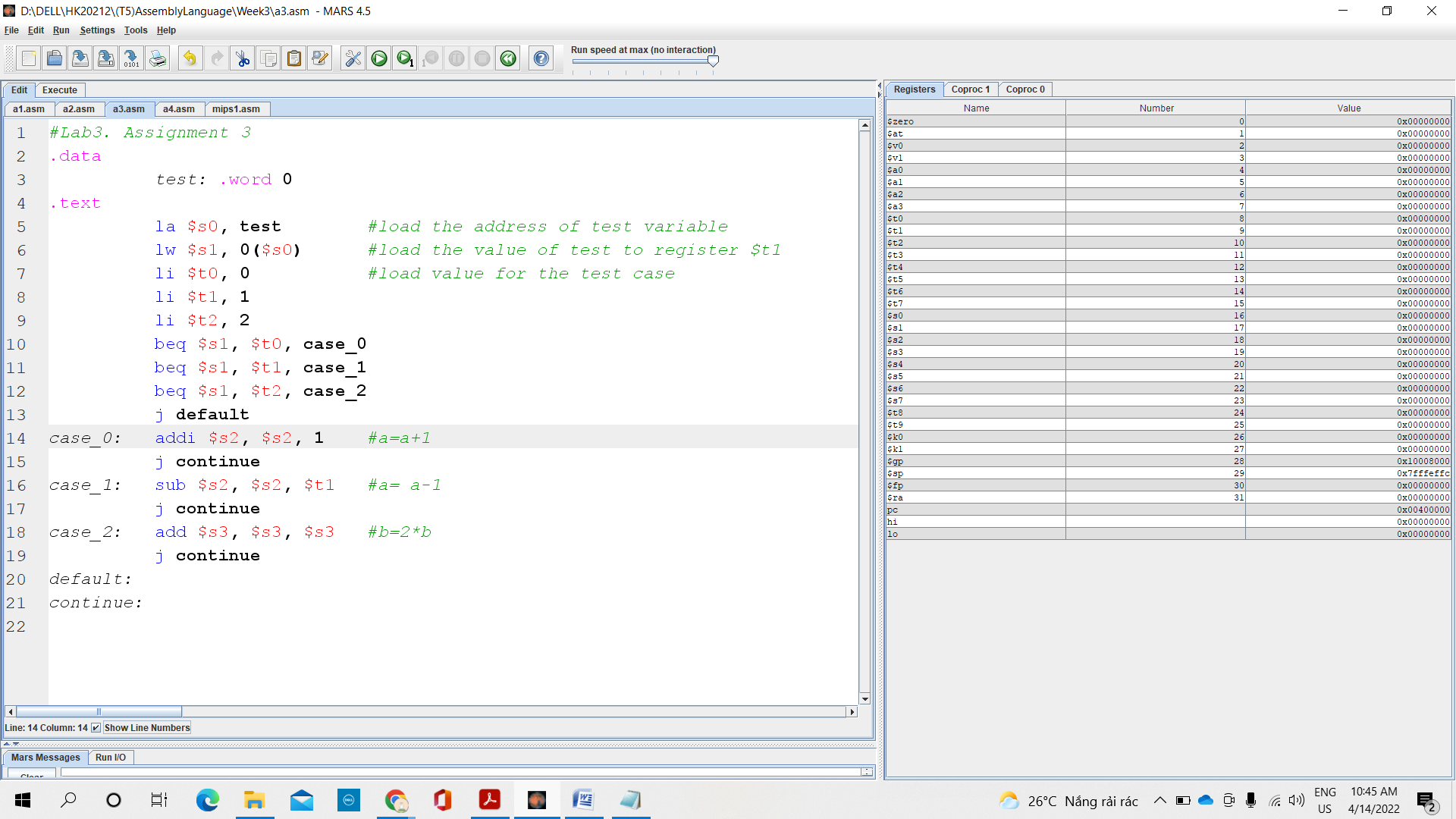
* $s1:0x00000003

pc: 0x00400030 => 0x00400034

* pc:0x00400034 =>0x00400018
* pc:0x00400018 => 0x00400038
* The final result : sum =$s5 =6

Assignment 3:

Case : test =0



* The change of registers:
  + “la” in line 5 is replaced by “lui” and “ori”
  + $at: 0x00000000 => 0x10010000

pc: 0x00400000 (lui (5)) =>0x00400004 (ori (5))

* + $s0: 0x00000000=> 0x10010000

pc: 0x00400004 (ori (5)) =>0x00400008 (lw (6))

* + $s1: 0x00000000 (= test = 0)

pc: 0x00400008 =>0x0040000c (li (7))

* + $t0: 0x00000000

pc: 0x0040000c (li (7)) => 0x00400010 (li (8))

* + $t1: 0x00000000=> 0x00000001

pc: 0x00400010=>0x00400014 (li (9))

* + $t2: 0x00000000=>0x00000002

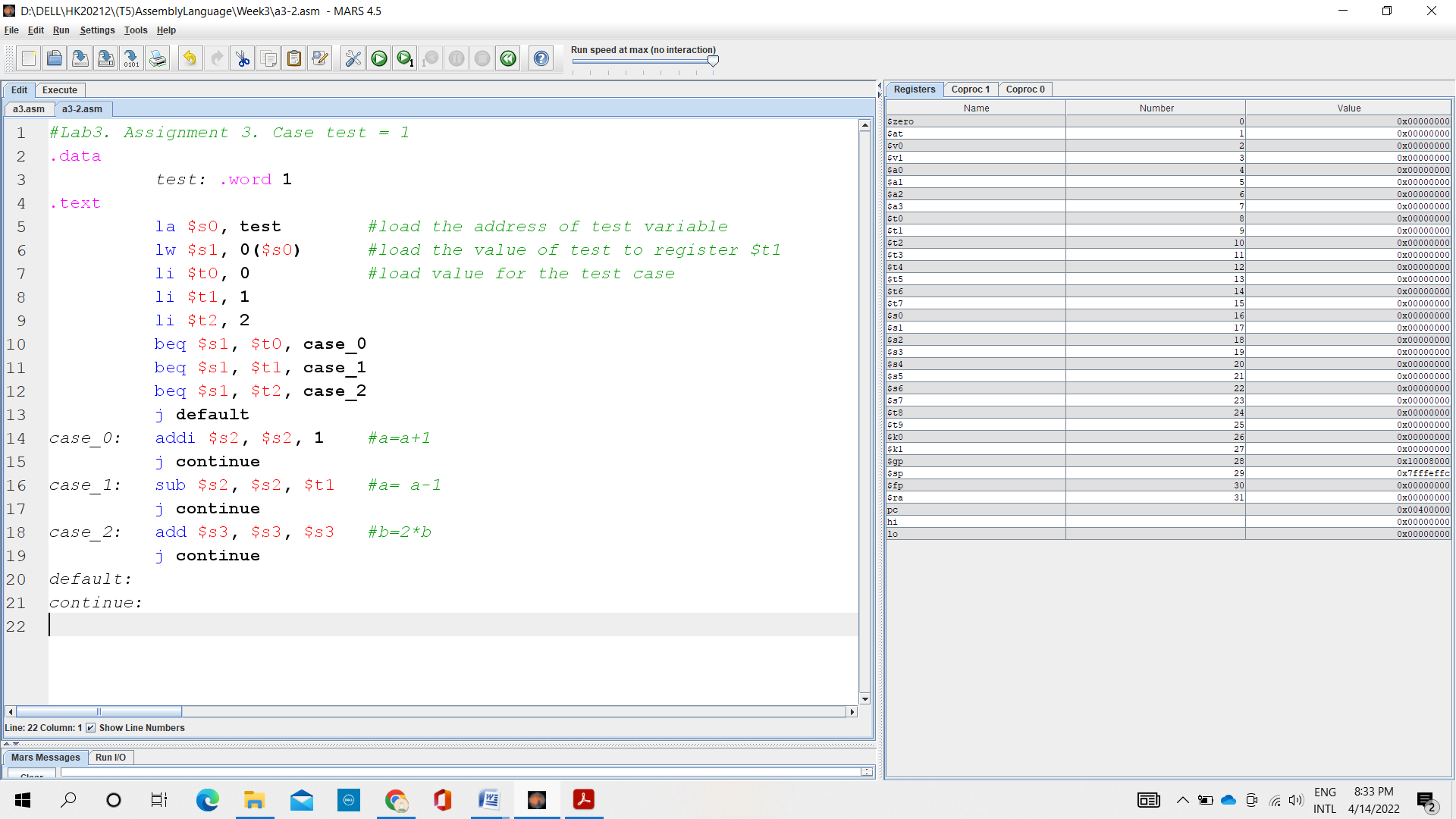
pc: 0x00400014 (li (9))=> 0x00400018 (beq (10))

* + pc: 0x00400018=> 0x00400028 (case\_0: addi (14)) (do test = $s1 = $t0 = 0)
  + $s2: 0x00000000=>0x00000001 (a = a + 1)

pc: 0x00400028=> 0x0040002c (j (15))

* + pc: 0x0040002c=>0x00400040 (continue (21))

Test =1:



The change of registers:

* “la” in line 5 is replaced by “lui” and “ori”
* $at: 0x00000000=> 0x10010000

pc: 0x00400000 (lui (5))=> 0x00400004 (ori (5))

* $s0: 0x00000000=> 0x10010000

pc: 0x00400004 (ori (5))=>0x00400008 (lw (6))

* $s1: 0x00000000=>0x00000001 (= test = 1)

pc: 0x00400008=> 0x0040000c (li (7))

* $t0: 0x00000000

pc: 0x0040000c (li (7))=> 0x00400010 (li (8))

* $t1: 0x00000000=> 0x00000001

pc: 0x00400010=>0x00400014 (li (9))

* $t2: 0x00000000=>0x00000002

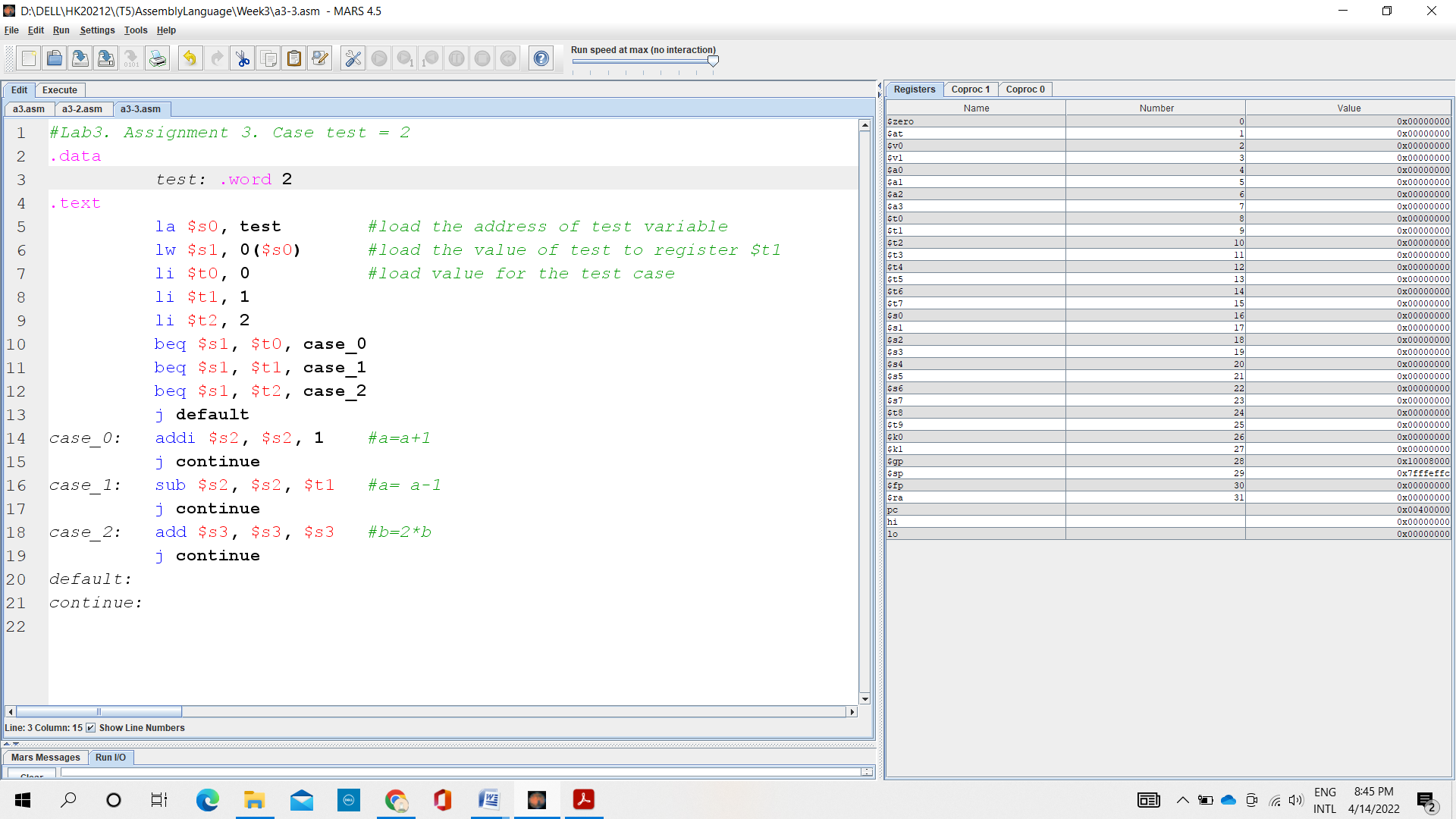
pc: 0x00400014 (li (9))=> 0x00400018 (beq (10))

* pc: 0x00400018=> 0x0040001c(beq(11)) (test= $s1 = 1≠ $t0=0)
* pc:0x0040001c=> 0x00400030 (case\_1: sub (16)) ( test = $s1 = $t1 = 1)
* $s2: 0x00000000=> 0xffffffff (a = a - 1)

pc: 0x00400030=> 0x00400034 (j (17))

* pc: 0x00400034=>0x00400040 (continue (21))

Test =2:



The change of registers:

* “la” in line 5 is replaced by “lui” and “ori”
* $at: 0x00000000=> 0x10010000

pc: 0x00400000 (lui (5))=>0x00400004 (ori (5))

* $s0: 0x00000000=>0x10010000

pc: 0x00400004 (ori (5))=> 0x00400008 (lw (6))

* $s1: 0x00000000=> 0x00000002 (= test = 2)

pc: 0x00400008=> 0x0040000c (li (7))

* $t0: 0x00000000

pc: 0x0040000c (li (7))=> 0x00400010 (li (8))

* $t1: 0x00000000=> 0x00000001

pc: 0x00400010=> 0x00400014 (li (9))

* $t2: 0x00000000=> 0x00000002

pc: 0x00400014 (li (9))=> 0x00400018 (beq (10))

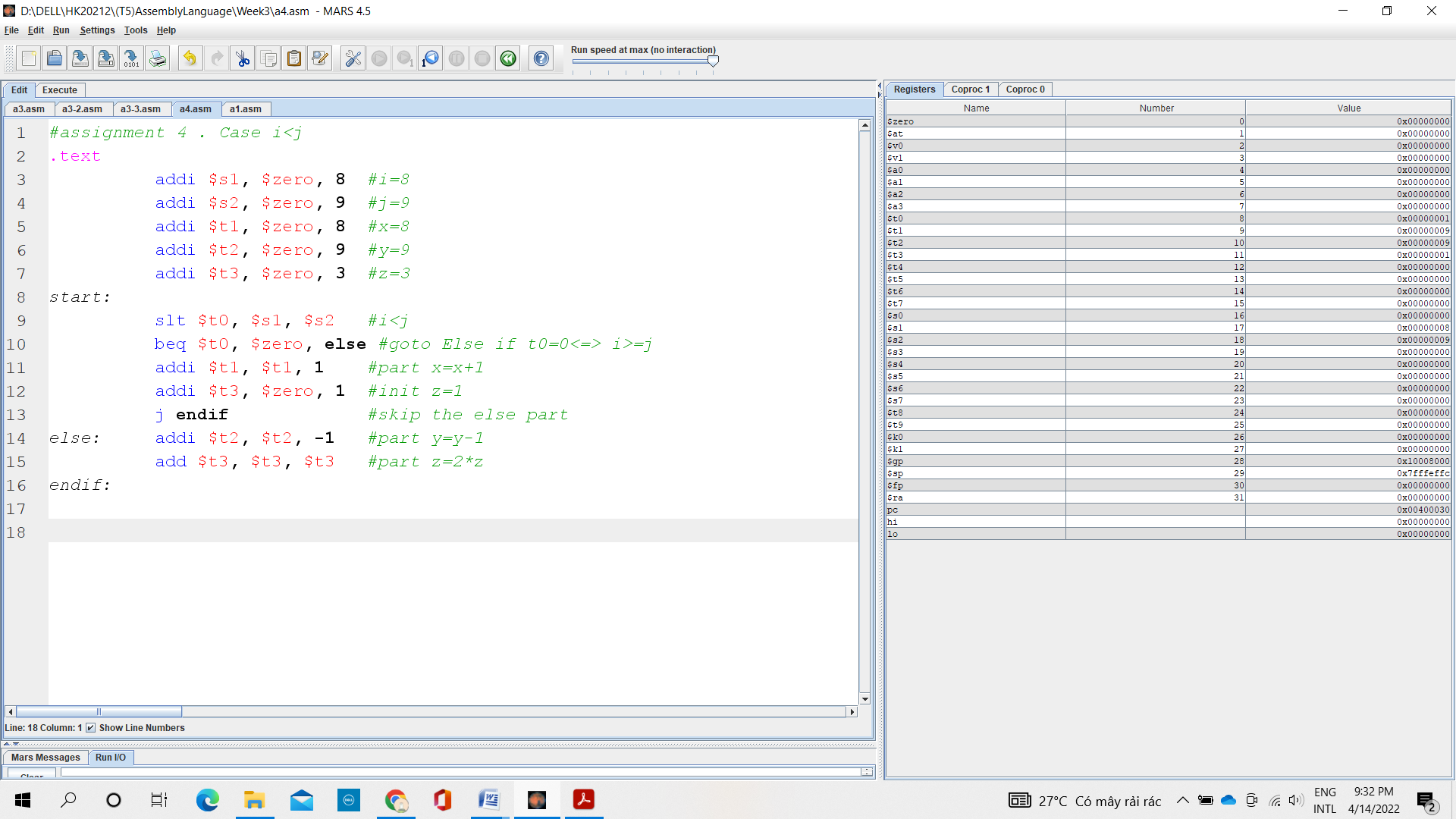
* pc: 0x00400018=> 0x0040001c (beq (11)) ( test = $s1 = 2 ≠ $t0 = 0)
* pc: 0x0040001c=> 0x00400020 (beq (12)) ( test = $s1 = 2 ≠ $t1 = 1)
* pc: 0x00400020=>0x00400038 (case\_2: add (18)) ( test = $s1 = $t2 = 2)
* $s3: 0x00000000 (b = 2 \* b)

pc: 0x00400038=>0x0040003c (j (19))

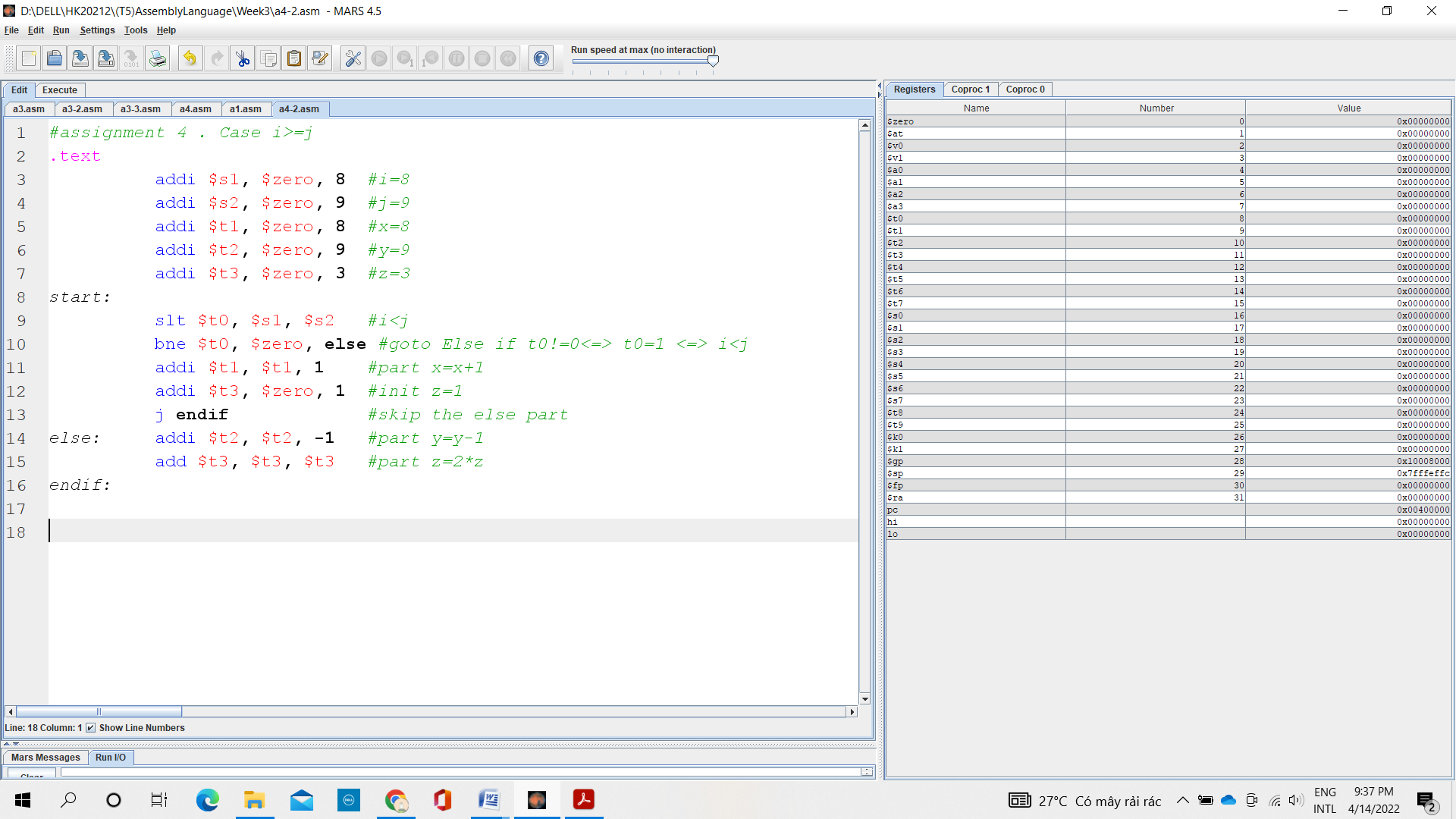
* pc: 0x00400034=>0x00400040 (continue (21))

Assignment 4:

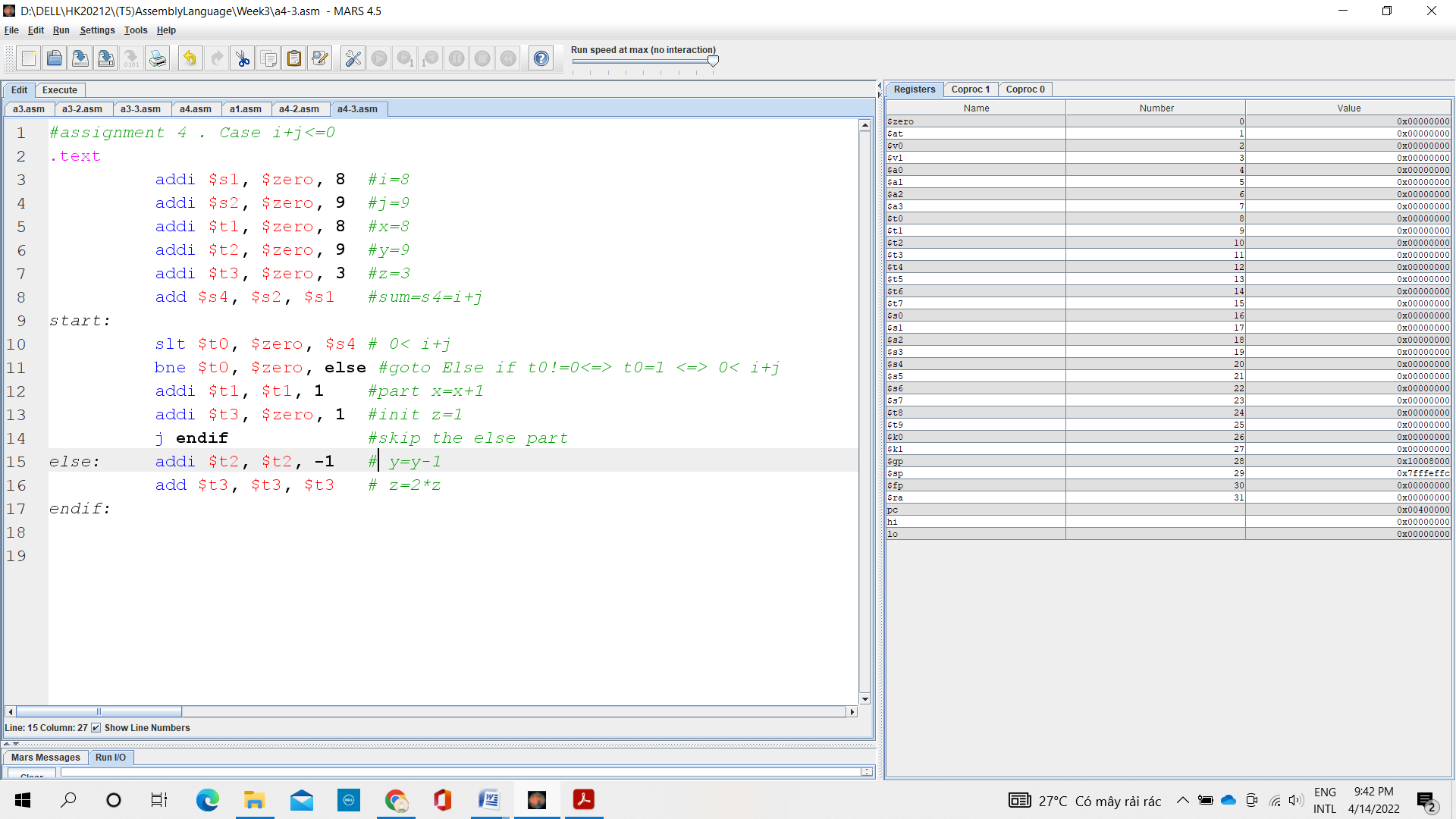
1. i<j



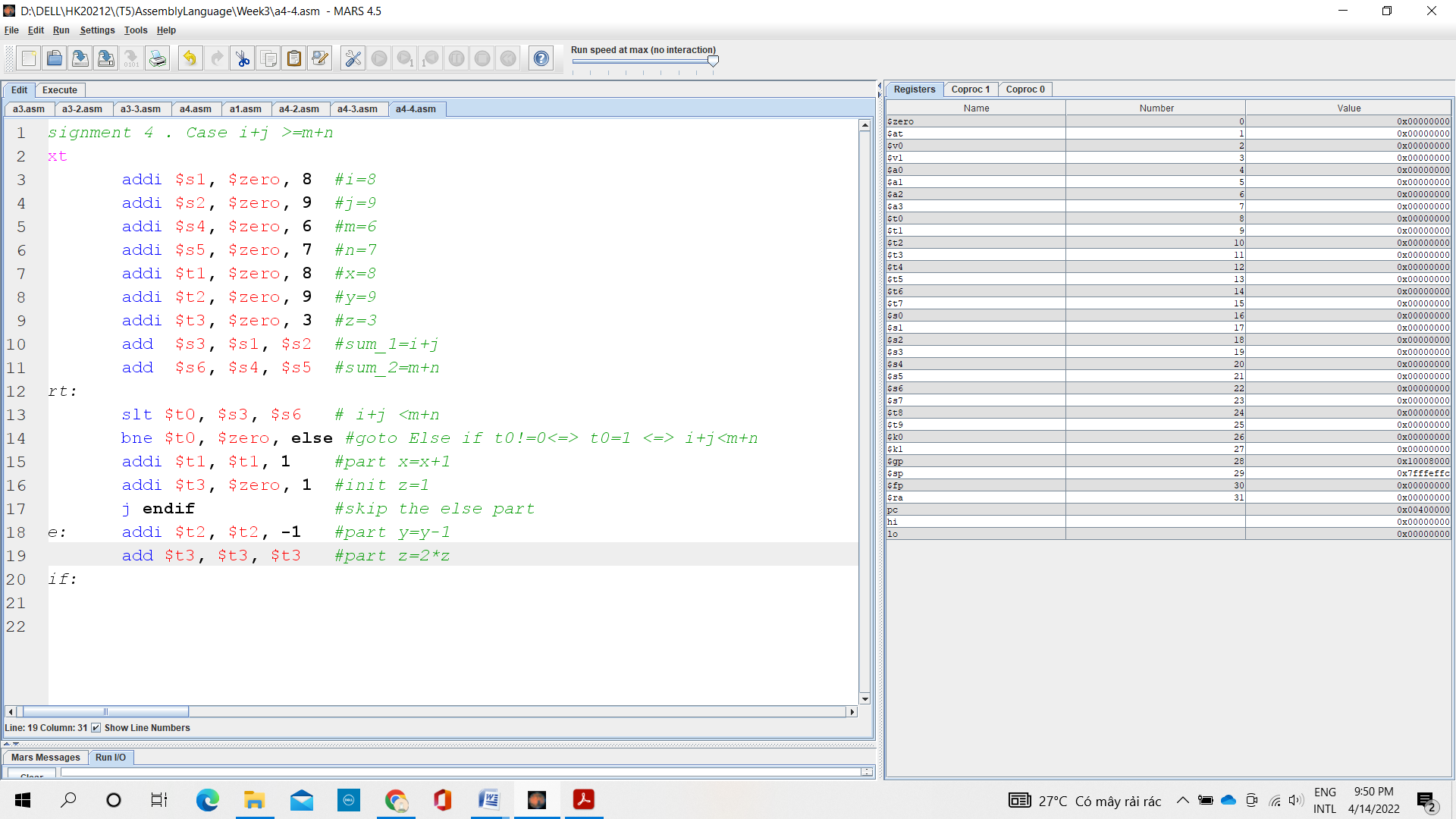
1. i>=j



1. i+j<=0

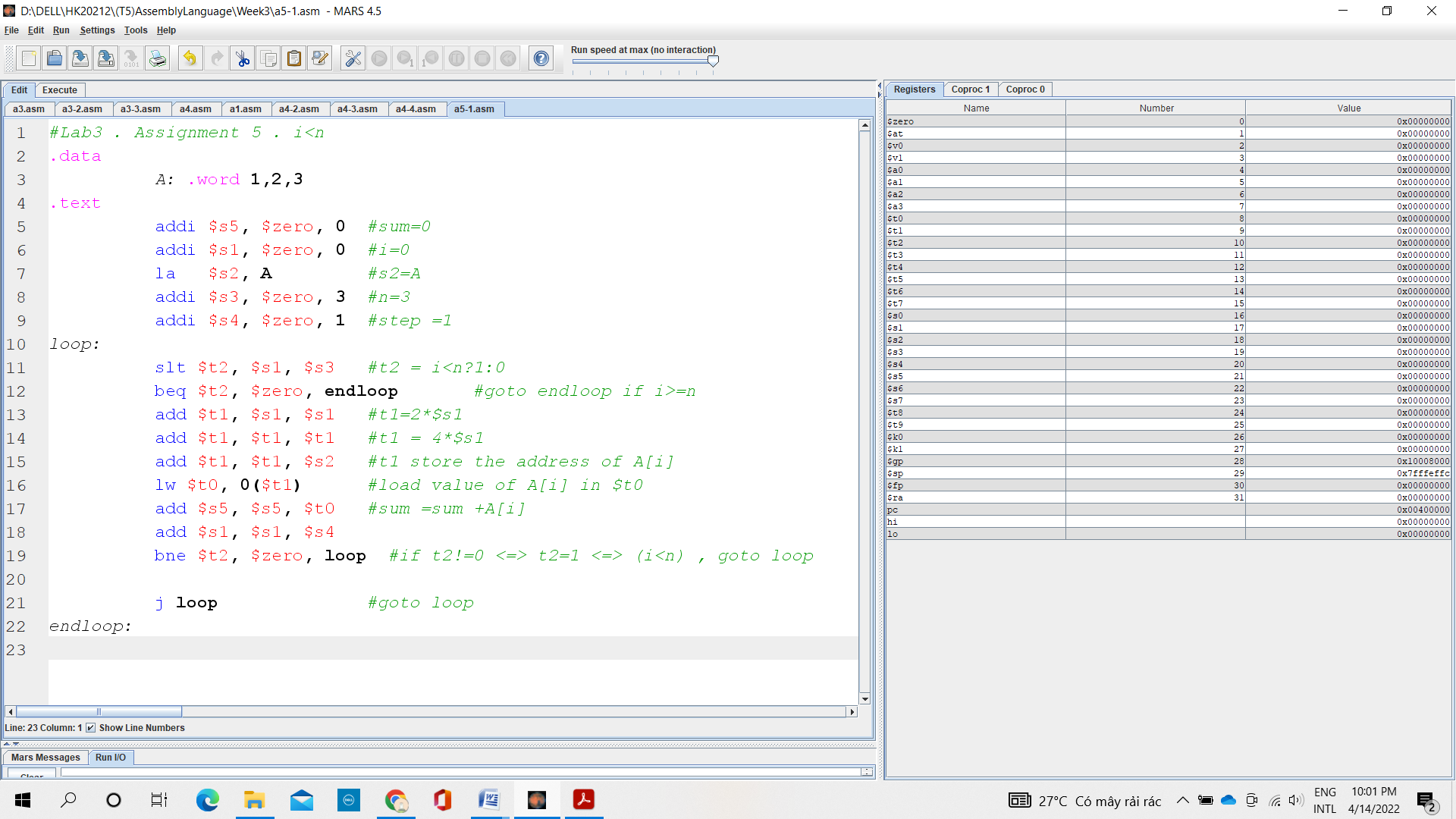


1. i+j >m+n

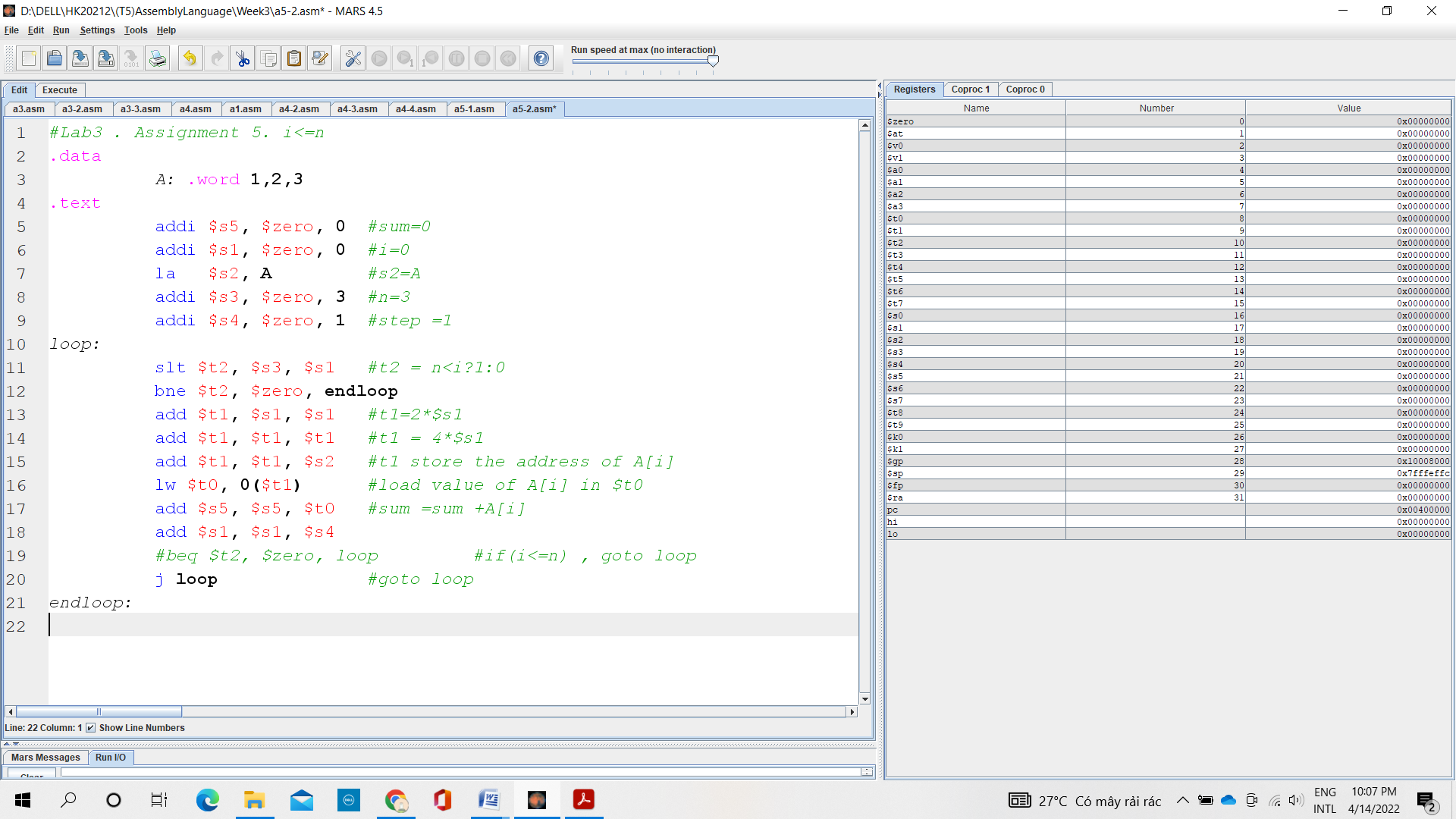


Assignment 5:

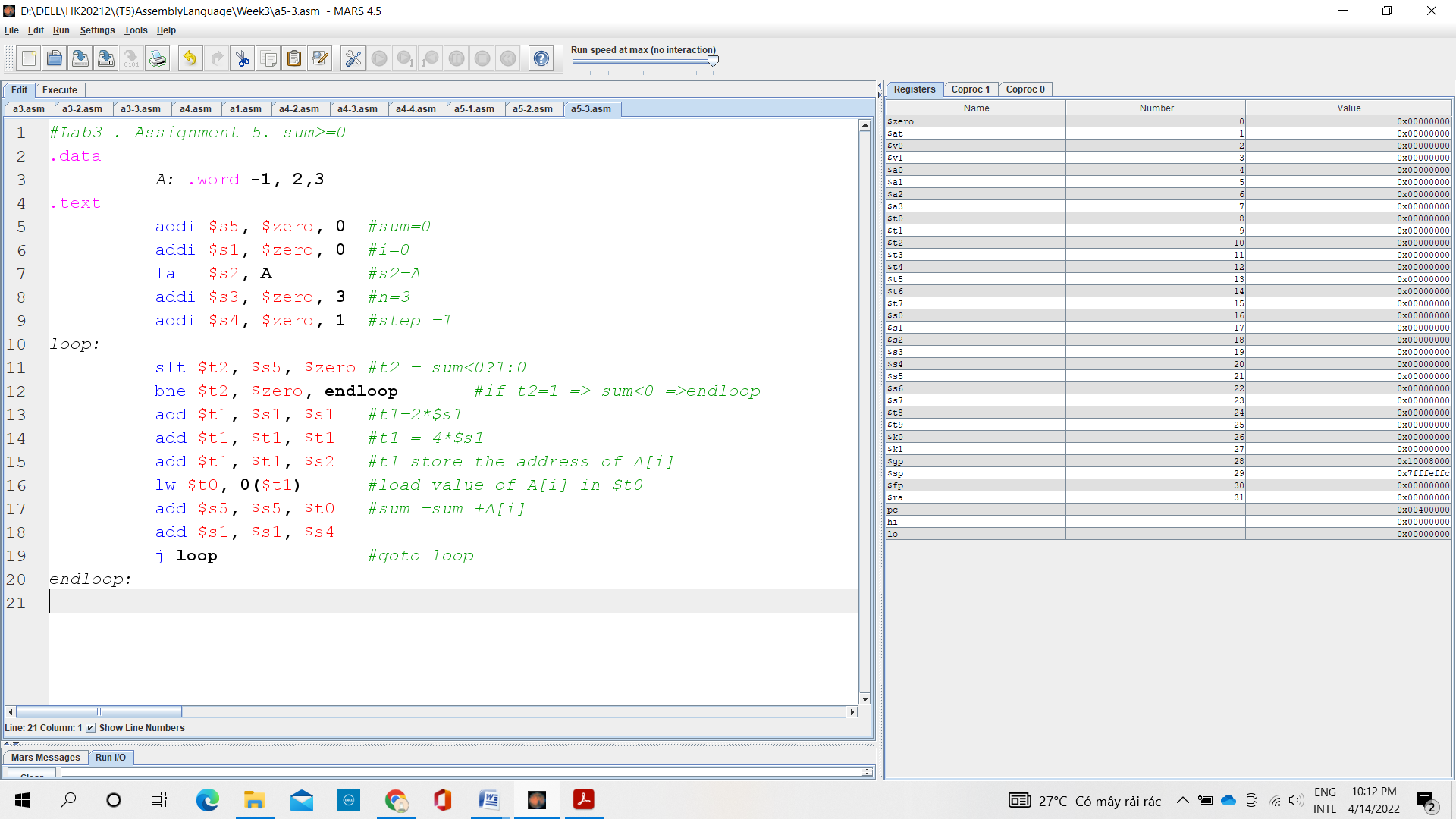
1. i<n



1. i<=n



1. sum >= 0



1. A[i] !=0

