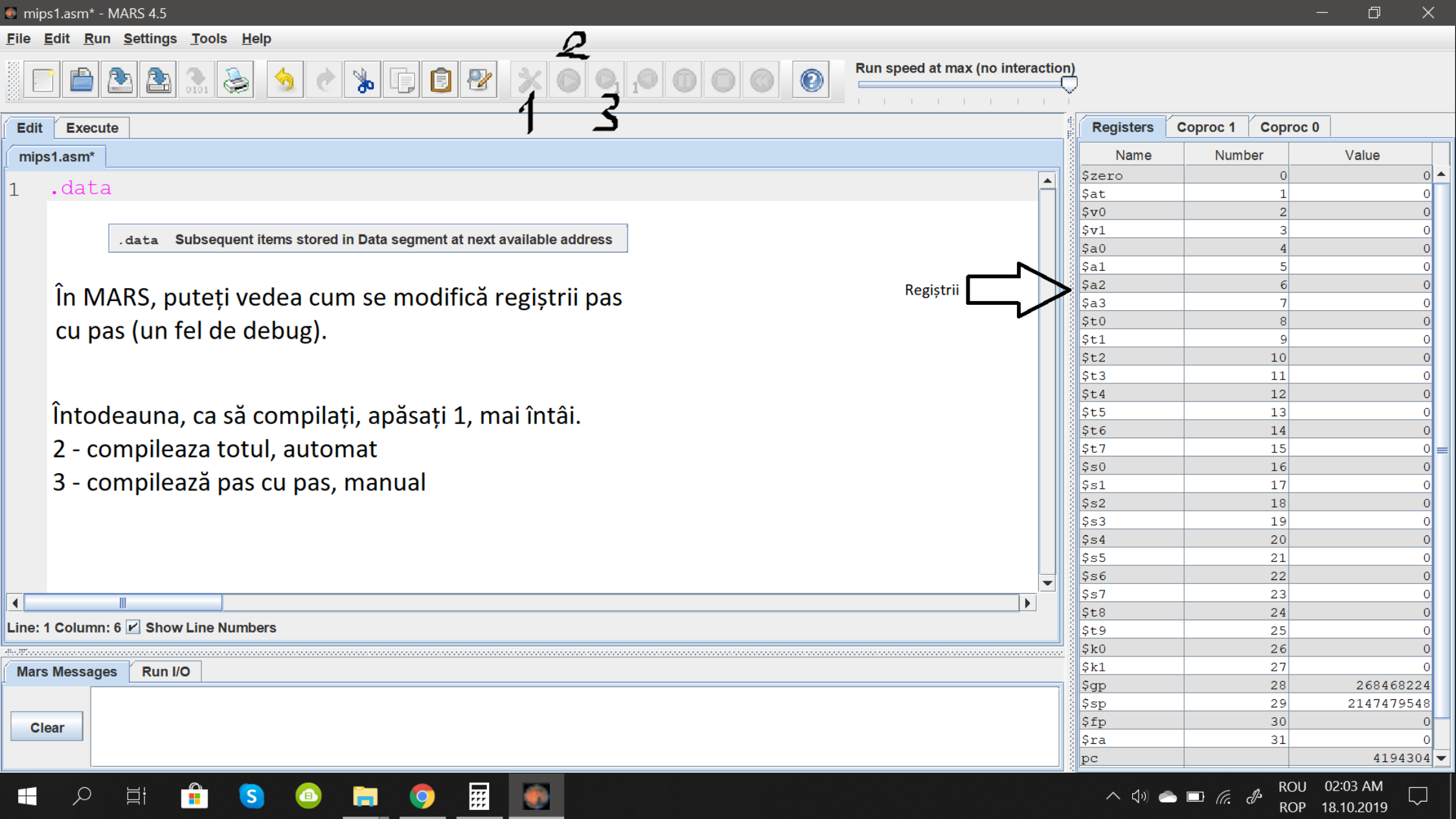

mips1.asm* - MARS 4.5
File Edit Run Settings Tools Help
2
1
3
Run speed at max (no interaction)
Edit Execute
mips1.asm*
1 .data
.data Subsequent items stored in Data segment at next available address
În MARS, puteți vedea cum se modifică regiștrii pas cu pas (un fel de debug).
Întodeauna, ca să compilați, apăsați 1, mai întâi.
2 - compileaza totul, automat
3 - compilează pas cu pas, manual
Regiștrii
Line: 1 Column: 6 Show Line Numbers
Mars Messages Run I/O
Clear
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 0
$t2 10 0
$t3 11 0
$t4 12 0
$t5 13 0
$t6 14 0
$t7 15 0
$s0 16 0
$s1 17 0
$s2 18 0
$s3 19 0
$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 4194304
ROU 02:03 AM
ROP 18.10.2019

mips1.asm* - MARS 4.5

File Edit Run Settings Tools Help

Run speed at max (no interaction)

Edit Execute

mips1.asm*

```
1  #Suma 1+2+3+...+n
2  .data #zona pentru declararea datelor
3          n: .word 11 #(word=)int n=11
4          s: .word 0 #int s=0 --- unde vom pune suma
5  .te
```

.text Subsequent items (instructions) stored in Text segment at next available address

Line: 5 Column: 4 Show Line Numbers

Mars Messages Run I/O

Clear

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	0
$t2	10	0
$t3	11	0
$t4	12	0
$t5	13	0
$t6	14	0
$t7	15	0
$s0	16	0
$s1	17	0
$s2	18	0
$s3	19	0
$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		4194304

ROU 02:31 AM
ROP 18.10.2019

mips1.asm* - MARS 4.5

File Edit Run Settings Tools Help

Run speed at max (no interaction)

Edit Execute

mips1.asm*

```
1  Suma 1+2+3+...+n
2  data #zona pentru declararea datelor
3        n: .word 11 #(word=)int n=11
4        s: .word 0 #int s=0 --- unde vom pune suma
5  text #codul programului
6    main:
7        #Nu lucram cu variabile, ci cu registrii (direct pe CPU),
8        #ceea ce mareste timpul de executare
9        lw
10
```

lw	Load word
lwc1	Load word into Coprocessor 1 (FPU)
lwl	Load word left
lwr	Load word right

Unele dintre motivele pt care recomand MARS, este faptul că vă lasă să vedeți denumirile comenzilor, dar și ce argumente trebuie să folosiți (uitați-vă în poza următoare).

Line: 9 Column: 7 Show Line Numbers

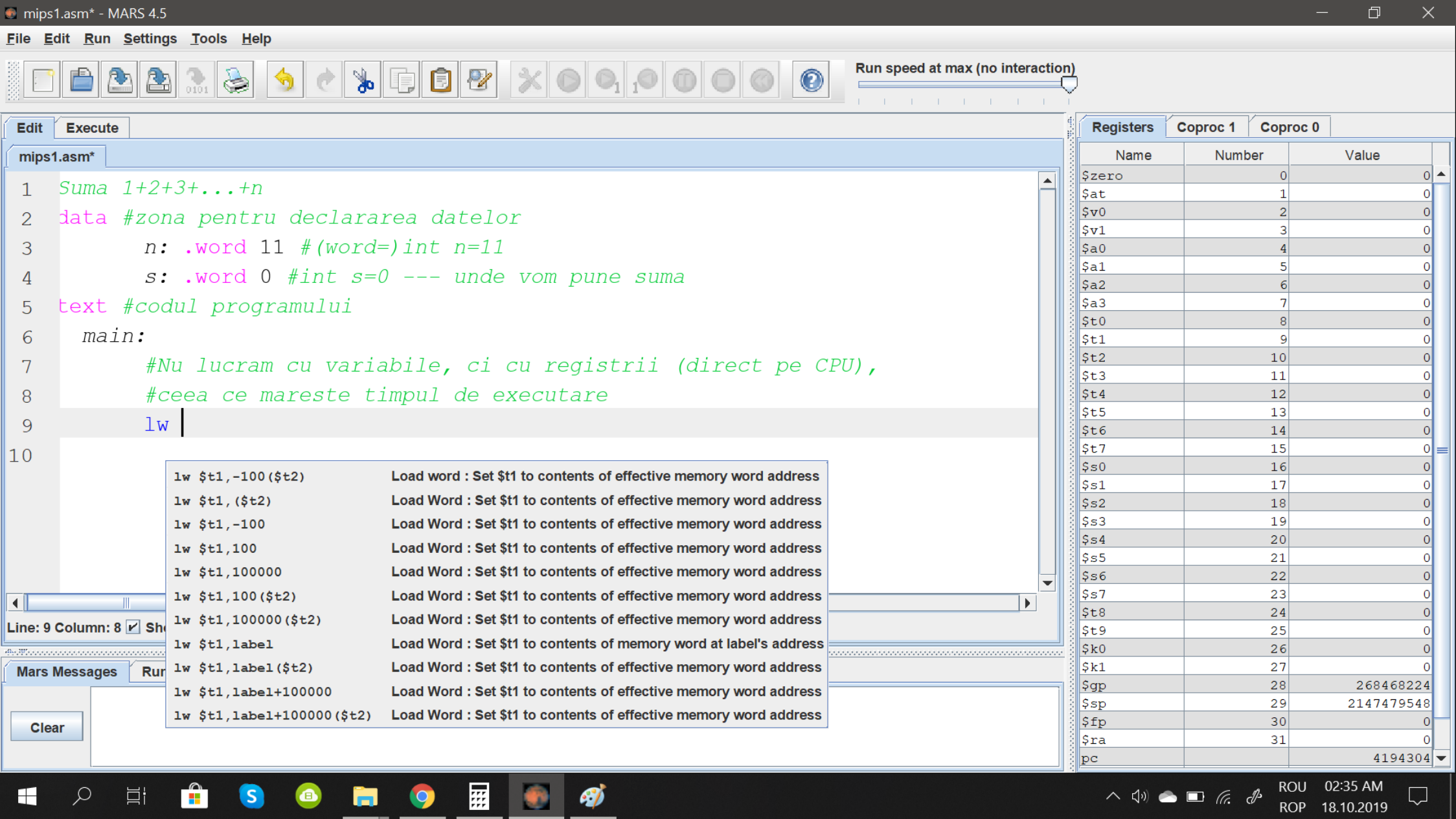
Mars Messages Run I/O

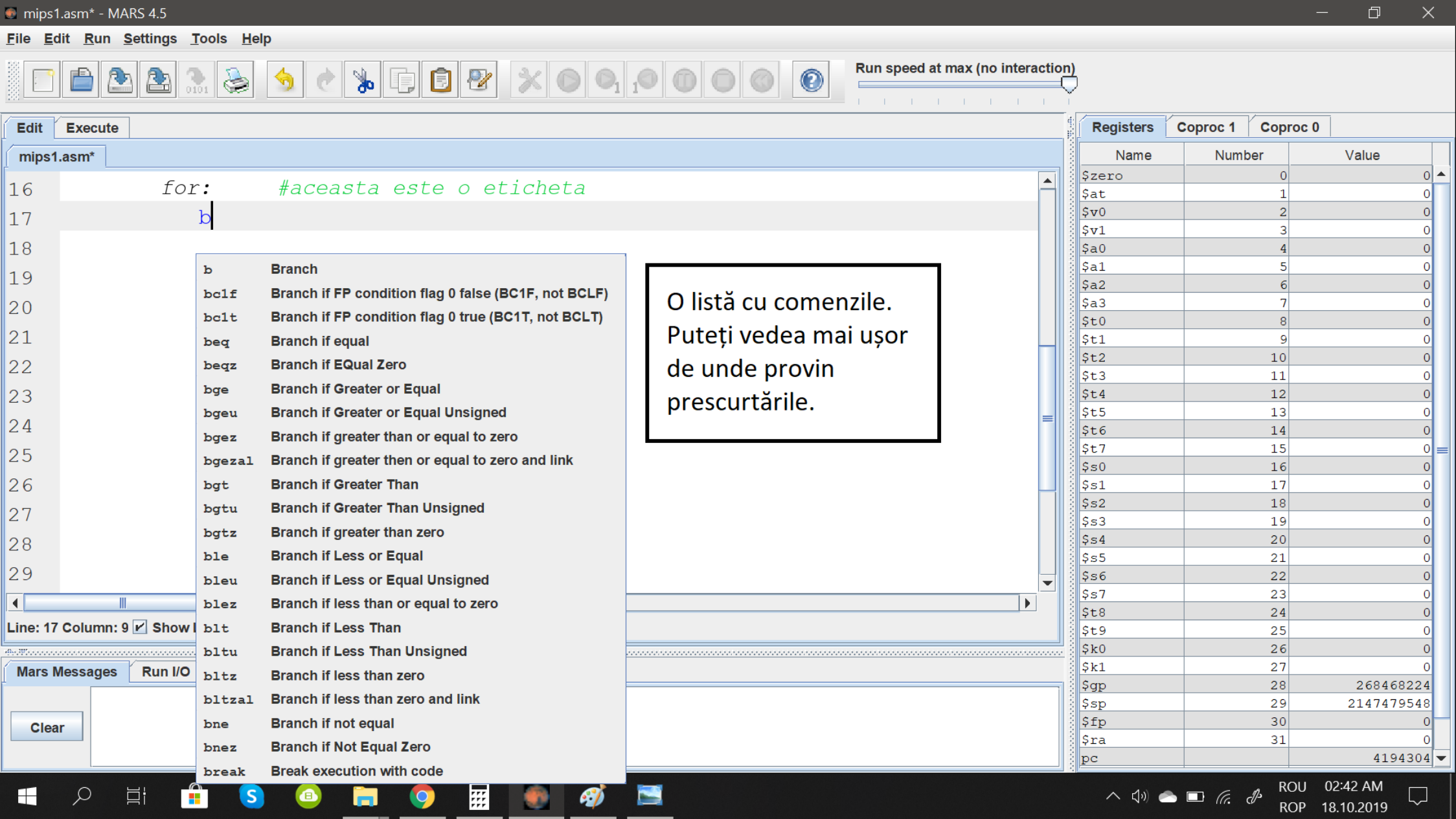
Clear

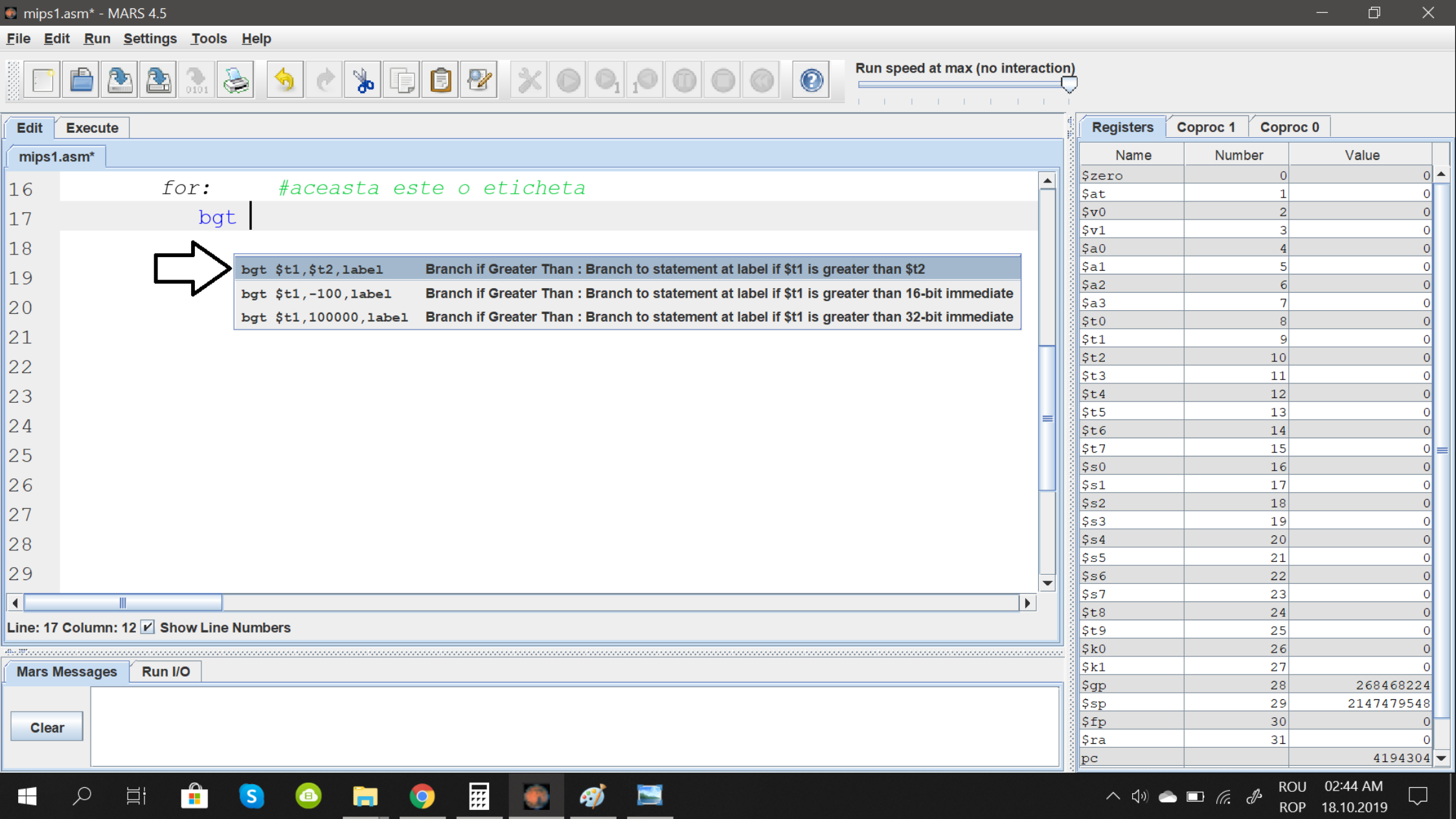
Registers Coproc 1 Coproc 0

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	0
$t2	10	0
$t3	11	0
$t4	12	0
$t5	13	0
$t6	14	0
$t7	15	0
$s0	16	0
$s1	17	0
$s2	18	0
$s3	19	0
$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		4194304

ROU 02:33 AM
ROP 18.10.2019


mips1.asm* - MARS 4.5
File Edit Run Settings Tools Help
Run speed at max (no interaction)
Edit Execute
mips1.asm*
1 Suma 1+2+3+...+n
2 data #zona pentru declararea datelor
3 n: .word 11 #(word=)int n=11
4 s: .word 0 #int s=0 --- unde vom pune suma
5 text #codul programului
6 main:
7 #Nu lucram cu variabile, ci cu registrii (direct pe CPU),
8 #ceea ce mareste timpul de executare
9 lw
10
lw $t1,-100($t2) Load word : Set $t1 to contents of effective memory word address
lw $t1,($t2) Load Word : Set $t1 to contents of effective memory word address
lw $t1,-100 Load Word : Set $t1 to contents of effective memory word address
lw $t1,100 Load Word : Set $t1 to contents of effective memory word address
lw $t1,100000 Load Word : Set $t1 to contents of effective memory word address
lw $t1,100($t2) Load Word : Set $t1 to contents of effective memory word address
lw $t1,100000($t2) Load Word : Set $t1 to contents of effective memory word address
lw $t1,label Load Word : Set $t1 to contents of memory word at label's address
lw $t1,label($t2) Load Word : Set $t1 to contents of effective memory word address
lw $t1,label+100000 Load Word : Set $t1 to contents of effective memory word address
lw $t1,label+100000($t2) Load Word : Set $t1 to contents of effective memory word address
Line: 9 Column: 8
Mars Messages
Clear
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 0
$t2 10 0
$t3 11 0
$t4 12 0
$t5 13 0
$t6 14 0
$t7 15 0
$s0 16 0
$s1 17 0
$s2 18 0
$s3 19 0
$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 4194304
ROU 02:35 AM
ROP 18.10.2019


mips1.asm* - MARS 4.5
File Edit Run Settings Tools Help
Run speed at max (no interaction)
Edit Execute
mips1.asm*
16 for: #aceasta este o eticheta
17 b
b Branch
bc1f Branch if FP condition flag 0 false (BC1F, not BCLF)
bc1t Branch if FP condition flag 0 true (BC1T, not BCLT)
beq Branch if equal
beqz Branch if EQual Zero
bge Branch if Greater or Equal
bgeu Branch if Greater or Equal Unsigned
bgez Branch if greater than or equal to zero
bgezal Branch if greater then or equal to zero and link
bgt Branch if Greater Than
bgtu Branch if Greater Than Unsigned
bgtz Branch if greater than zero
ble Branch if Less or Equal
bleu Branch if Less or Equal Unsigned
blez Branch if less than or equal to zero
blt Branch if Less Than
bltu Branch if Less Than Unsigned
bltz Branch if less than zero
bltzal Branch if less than zero and link
bne Branch if not equal
bnez Branch if Not Equal Zero
break Break execution with code
O listă cu comenzile. Puteți vedea mai ușor de unde provin prescurtările.
Line: 17 Column: 9 Show
Mars Messages Run I/O
Clear
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 0
$t2 10 0
$t3 11 0
$t4 12 0
$t5 13 0
$t6 14 0
$t7 15 0
$s0 16 0
$s1 17 0
$s2 18 0
$s3 19 0
$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 4194304
ROU 02:42 AM
ROP 18.10.2019


mips1.asm* - MARS 4.5
File Edit Run Settings Tools Help
Run speed at max (no interaction)
Edit Execute
mips1.asm*
16 for: #aceasta este o eticheta
17 bgt
bgt $t1,$t2,label Branch if Greater Than : Branch to statement at label if $t1 is greater than $t2
bgt $t1,-100,label Branch if Greater Than : Branch to statement at label if $t1 is greater than 16-bit immediate
bgt $t1,100000,label Branch if Greater Than : Branch to statement at label if $t1 is greater than 32-bit immediate
Line: 17 Column: 12 Show Line Numbers
Mars Messages Run I/O
Clear
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 0
$t2 10 0
$t3 11 0
$t4 12 0
$t5 13 0
$t6 14 0
$t7 15 0
$s0 16 0
$s1 17 0
$s2 18 0
$s3 19 0
$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 4194304
ROU 02:44 AM
ROP 18.10.2019

MIPS syscall functions available

Nesecurizat | courses.missouristate.edu/k...

Aplicații Magazinul Google Facebook YouTube Mail IMDb

```
add $a0, $t0, $zero  # load desired value into argument register $a0, using pseudo
syscall
```

Table of Available Services

Service	Code in $v0	Arguments	Result
print integer	1	$a0 = integer to print	
print float	2	$f12 = float to print	
print double	3	$f12 = double to print	
print string	4	$a0 = address of null-terminated string to print	
read integer	5		$v0 contains integer read
read float	6		$f0 contains float read
read double	7		$f0 contains double read
read string	8	$a0 = address of input buffer $a1 = maximum number of characters to read	*See note below table*
sbrk (allocate heap memory)	9	$a0 = number of bytes to allocate	$v0 contains address of allocated memory
exit (terminate			

Găsiți site-ul în "Link-uri Utile" din folderul MIPS-Materiale

mips1.asm* - MARS 4.5

File Edit Run Settings Tools Help

Edit Execute

mips1.asm*

```
22
23          final:
24          #punem suma pe care am gasit-o in "s" (ca
25          sw $t0, s    #s=t0
26
27          #afisam suma pe ecran
28          li $v0, 1 #pt print de int, tb sa avem v0
29          #cand se face print int, procesorul se ui
30          #registrul a0 si o afiseaza
31          lw $a0, s #a0=s
32          syscall
33
34
35
```

Line: 31 Column: 14 Show Line Numbers

Mars Messages Run I/O

Clear

ROU 02:58 AM
ROP 18.10.2019

C:\Users\Larisa\Desktop\Suma_Gauss.asm* - MARS 4.5

File Edit Run Settings Tools Help

Run speed at max (no interaction)

Edit Execute

Suma_Gauss.asm*

```
1  #Suma 1+2+3+...+n=?
2  #Presupunem acum ca citim n de la tastatura
3  .data #zona pentru declararea datelor
4          s: .word 0     #int s=0 --- unde vom pune suma
5  .text #codul programului
6     main:
7          li $v0, 5 #cin>>int  --- int se gaseste in v0
8          syscall
9          move $t2, $v0
10
11         #Initia                    roblemei
12         li $t0, 0    #t0=s=0
13         li $t1, 1    #t1=i=1
14
```

move $t1,$t2 MOVE : Set $t1 to contents of $t2

Line: 9 Column: 13 Show Line Numbers

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	0
$t2	10	0
$t3	11	0
$t4	12	0
$t5	13	0
$t6	14	0
$t7	15	0
$s0	16	0
$s1	17	0
$s2	18	0
$s3	19	0
$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		4194304

Mars Messages Run I/O

Clear

```
66
-- program is finished running --
```

ROU 03:07 AM
ROP 18.10.2019