MONTCLAIR STATE
UNIVERSITY

Assessment Module-5

Rana, Karan

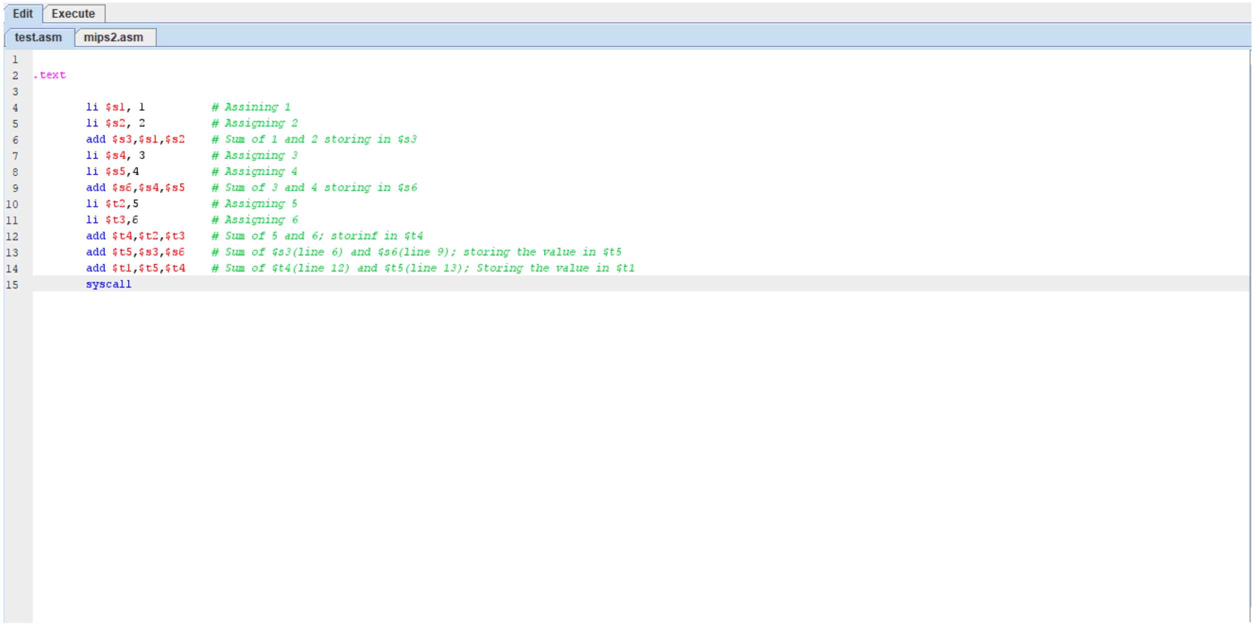
CSIT_230SP21

Instructor: Dr. G.E. Antoniou
TA: Dev Dave
4,4,2021

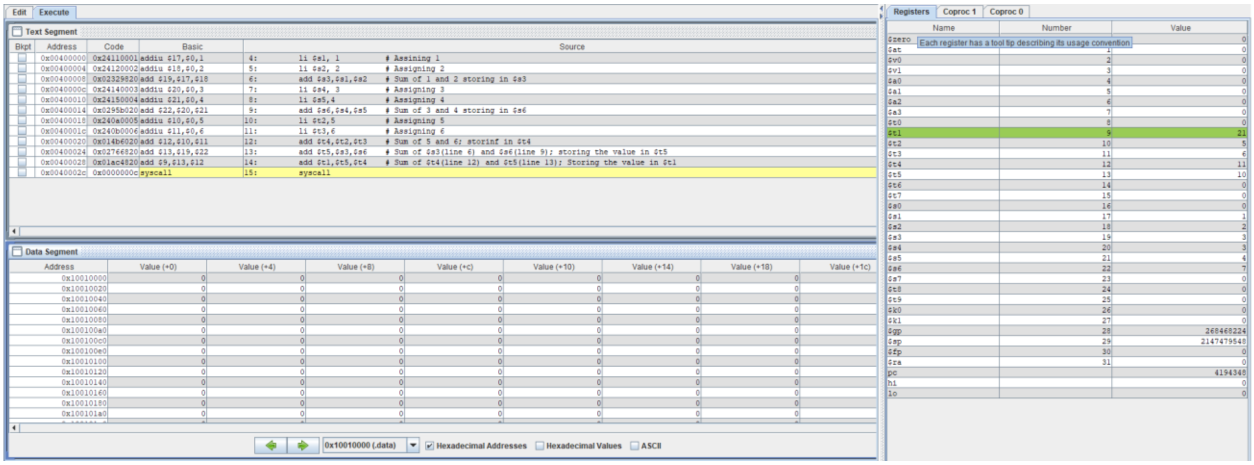
Problem 1:

Write a MIPS Assembly program to ADD the numbers: 1, 2, 3, 4, 5, 6 (as in ex–1)

- Use ONLY the instructions: add and li
- The result of the addition (in decimal) should be in register: $t1



```
1
2  .text
3
4          li $s1, 1           # Assining 1
5          li $s2, 2           # Assigning 2
6          add $s3,$s1,$s2     # Sum of 1 and 2 storing in $s3
7          li $s4, 3           # Assigning 3
8          li $s5,4            # Assigning 4
9          add $s6,$s4,$s5     # Sum of 3 and 4 storing in $s6
10         li $t2,5            # Assigning 5
11         li $t3,6            # Assigning 6
12         add $t4,$t2,$t3     # Sum of 5 and 6; storinf in $t4
13         add $t5,$s3,$s6     # Sum of $s3(line 6) and $s6(line 9); storing the value in $t5
14         add $t1,$t5,$t4     # Sum of $t4(line 12) and $t5(line 13); Storing the value in $t1
15         syscall
```


Edit
Execute
Registers
Coproc 1
Coproc 0
Each register has a tool tip describing its usage convention

Text Segment

Bkpt	Address	Code	Basic	Source
☐	0x00400000	0x24110001	addiu $17,$0,1	4: li $s1, 1 # Assining 1
☐	0x00400004	0x24120002	addiu $18,$0,2	5: li $s2, 2 # Assigning 2
☐	0x00400008	0x02329820	add $19,$17,$18	6: add $s3,$s1,$s2 # Sum of 1 and 2 storing in $s3
☐	0x0040000c	0x24140003	addiu $20,$0,3	7: li $s4, 3 # Assigning 3
☐	0x00400010	0x24150004	addiu $21,$0,4	8: li $s5,4 # Assigning 4
☐	0x00400014	0x0295b020	add $22,$20,$21	9: add $s6,$s4,$s5 # Sum of 3 and 4 storing in $s6
☐	0x00400018	0x240a0005	addiu $10,$0,5	10: li $t2,5 # Assigning 5
☐	0x0040001c	0x240b0006	addiu $11,$0,6	11: li $t3,6 # Assigning 6
☐	0x00400020	0x014b6020	add $12,$10,$11	12: add $t4,$t2,$t3 # Sum of 5 and 6; storinf in $t4
☐	0x00400024	0x02766820	add $13,$19,$22	13: add $t5,$s3,$s6 # Sum of $s3(line 6) and $s6(line 9); storing the value in $t5
☐	0x00400028	0x01ac4820	add $9,$13,$12	14: add $t1,$t5,$t4 # Sum of $t4(line 12) and $t5(line 13); Storing the value in $t1
☐	0x0040002c	0x0000000c	syscall	15: syscall

Data Segment

Address	Value (+0)	Value (+4)	Value (+8)	Value (+c)	Value (+10)	Value (+14)	Value (+18)	Value (+1c)
0x10010000	0	0	0	0	0	0	0	
0x10010020	0	0	0	0	0	0	0	
0x10010040	0	0	0	0	0	0	0	
0x10010060	0	0	0	0	0	0	0	
0x10010080	0	0	0	0	0	0	0	
0x100100a0	0	0	0	0	0	0	0	
0x100100c0	0	0	0	0	0	0	0	
0x100100e0	0	0	0	0	0	0	0	
0x10010100	0	0	0	0	0	0	0	
0x10010120	0	0	0	0	0	0	0	
0x10010140	0	0	0	0	0	0	0	
0x10010160	0	0	0	0	0	0	0	
0x10010180	0	0	0	0	0	0	0	
0x100101a0	0	0	0	0	0	0	0	

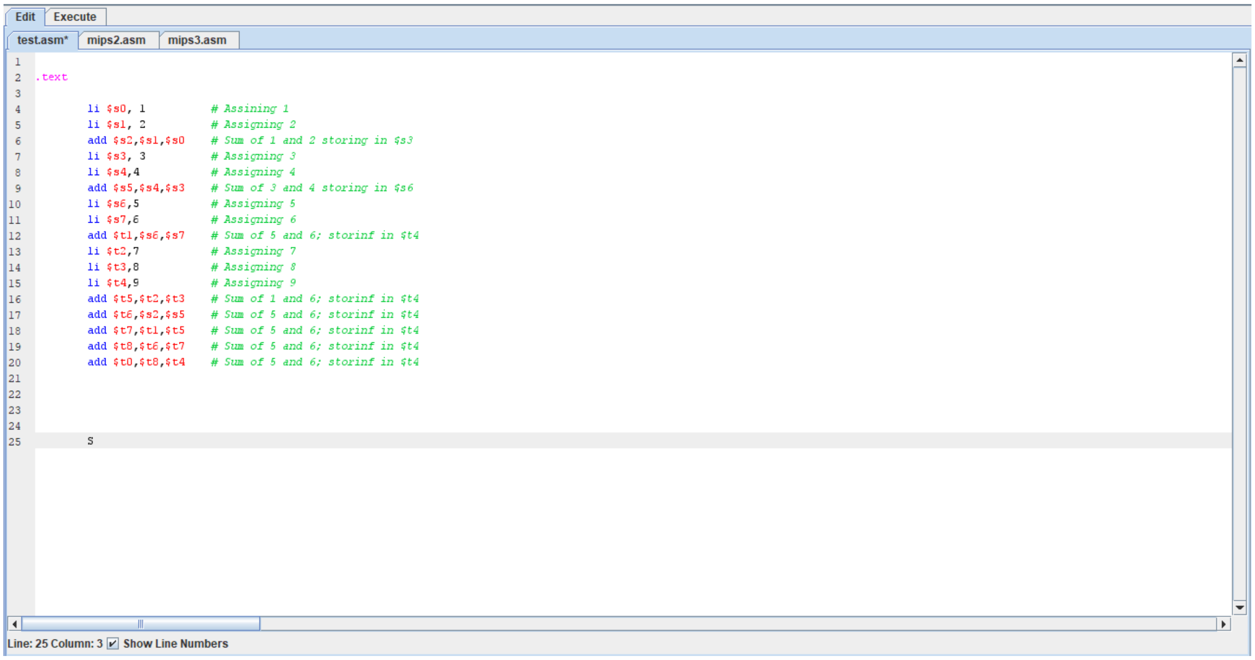
0x10010000 (.data) ☑ Hexadecimal Addresses ☐ Hexadecimal Values ☐ ASCII

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	21
$t2	10	5
$t3	11	6
$t4	12	11
$t5	13	10
$t6	14	0
$t7	15	0
$s0	16	0
$s1	17	1
$s2	18	2
$s3	19	3
$s4	20	3
$s5	21	4
$s6	22	7
$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		4194348
hi		0
lo		0

Problem 2:

Write a MIPS Assembly program to ADD the numbers: 1, 2, 3, 4, 5, 6, 7, 8, 9.

- Use ONLY the instructions: add and li
- Use any $t, $s registers
- The result of the addition (in decimal) should be in register: $t0



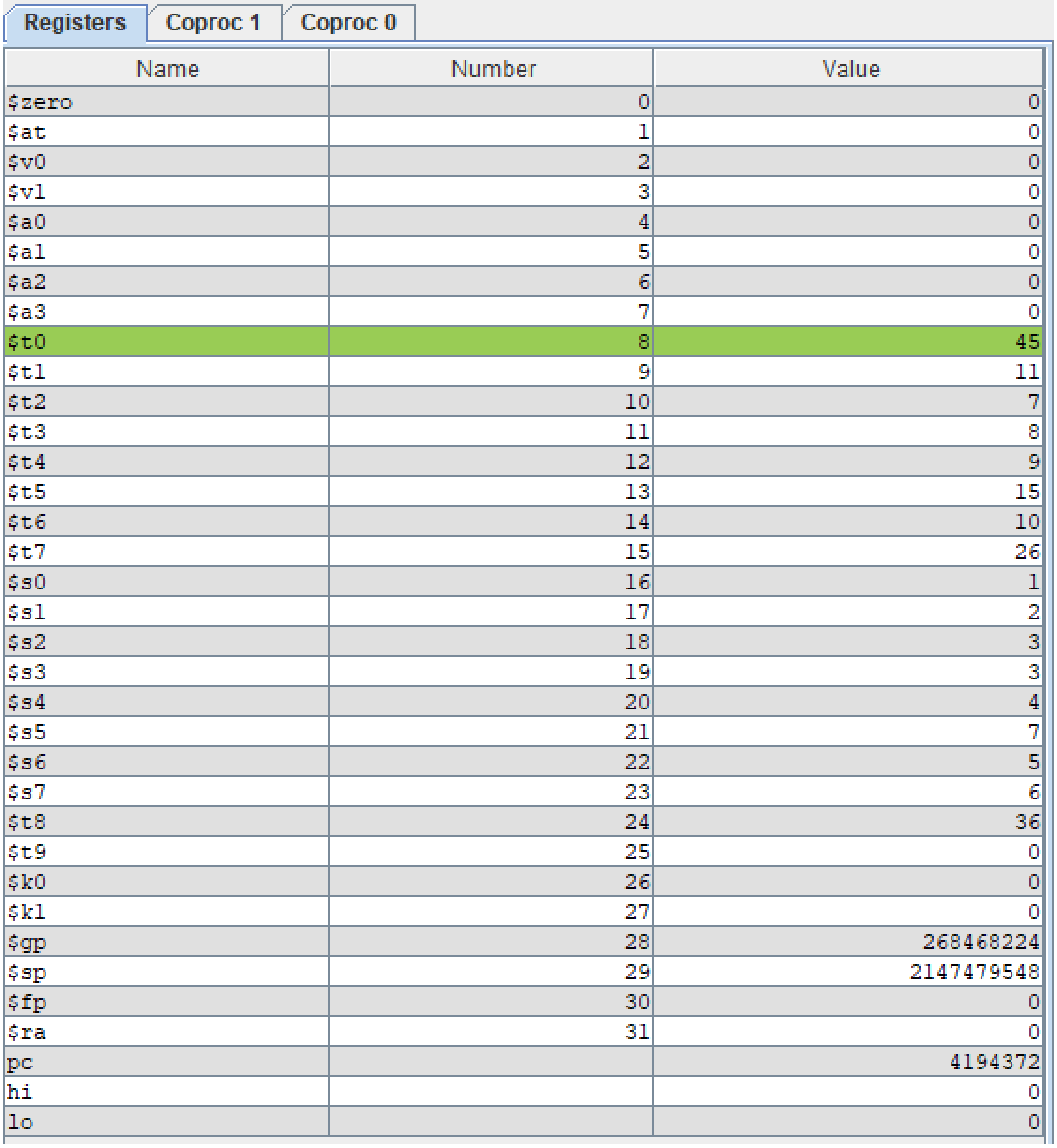
```

.text

        li $s0, 1          # Assining 1
        li $s1, 2          # Assigning 2
        add $s2,$s1,$s0    # Sum of 1 and 2 storing in $s3
        li $s3, 3          # Assigning 3
        li $s4,4           # Assigning 4
        add $s5,$s4,$s3    # Sum of 3 and 4 storing in $s6
        li $s6,5           # Assigning 5
        li $s7,6           # Assigning 6
        add $t1,$s6,$s7    # Sum of 5 and 6; storinf in $t4
        li $t2,7           # Assigning 7
        li $t3,8           # Assigning 8
        li $t4,9           # Assigning 9
        add $t5,$t2,$t3    # Sum of 1 and 6; storinf in $t4
        add $t6,$s2,$s5    # Sum of 5 and 6; storinf in $t4
        add $t7,$t1,$t5    # Sum of 5 and 6; storinf in $t4
        add $t8,$t6,$t7    # Sum of 5 and 6; storinf in $t4
        add $t0,$t8,$t4    # Sum of 5 and 6; storinf in $t4




        s
```

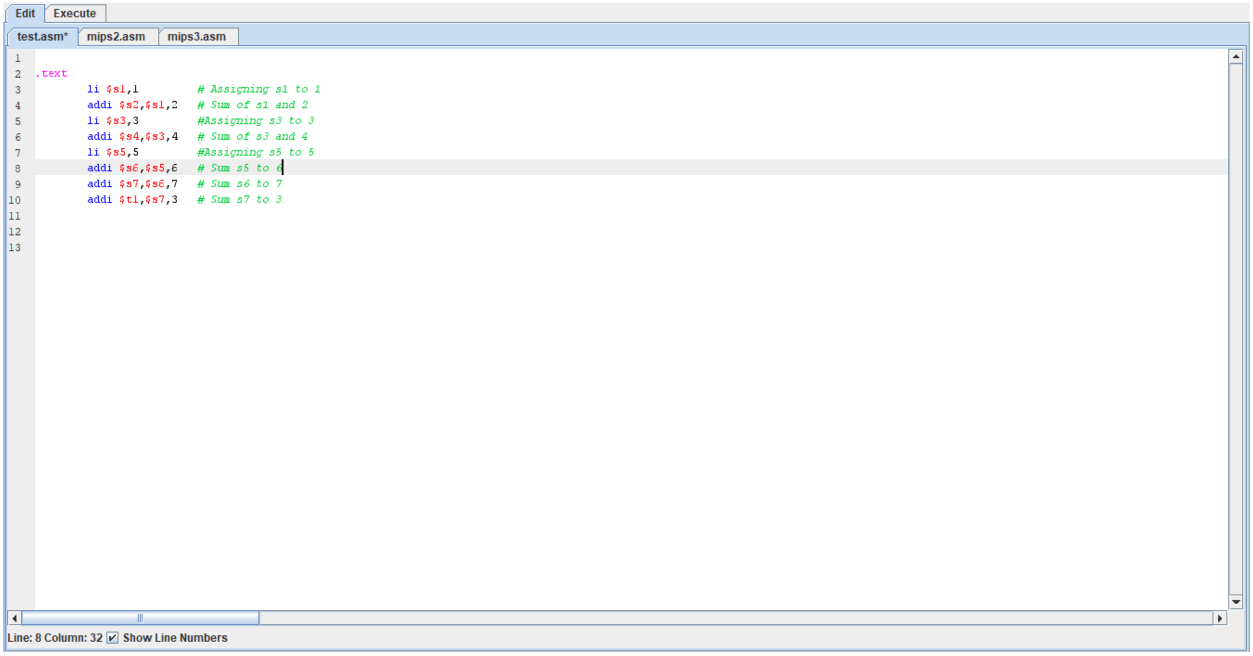


Name	Number	Value
$zero	0	0
$at	1	0
$v0	2	0
$v1	3	0
$a0	4	0
$a1	5	0
$a2	6	0
$a3	7	0
$t0	8	45
$t1	9	11
$t2	10	7
$t3	11	8
$t4	12	9
$t5	13	15
$t6	14	10
$t7	15	26
$s0	16	1
$s1	17	2
$s2	18	3
$s3	19	3
$s4	20	4
$s5	21	7
$s6	22	5
$s7	23	6
$t8	24	36
$t9	25	0
$k0	26	0
$k1	27	0
$gp	28	268468224
$sp	29	2147479548
$fp	30	0
$ra	31	0
pc		4194372
hi		0
lo		0

Problem 3:

Write a MIPS Assembly program to ADD the numbers: 1, 2, 3, 4, 5, 6 (as in ex–2)

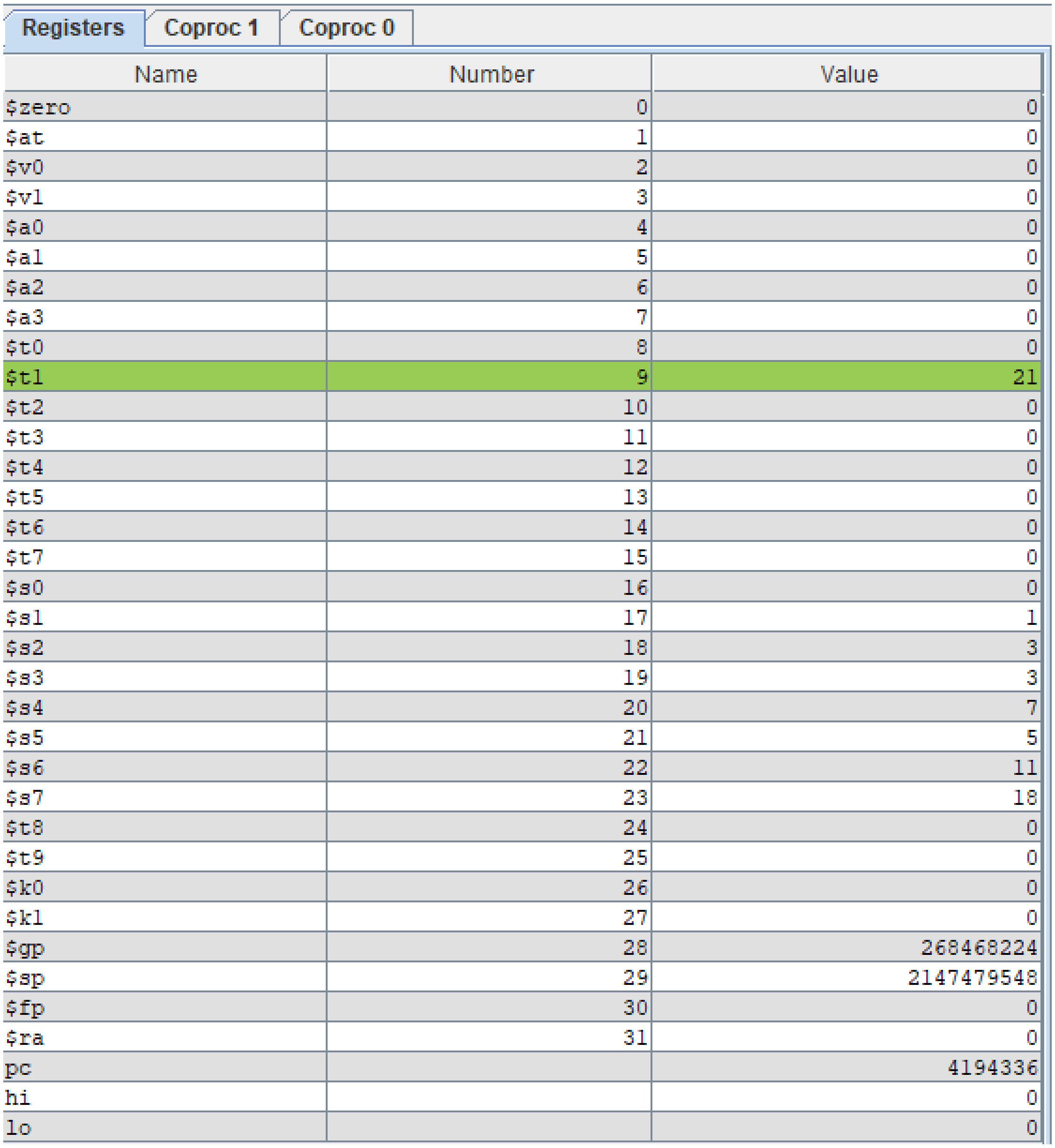
- Use ONLY the instructions: addi and li
- The result of the addition (in decimal) should be in register: $t1



```

.text
        li $s1,1        # Assigning s1 to 1
        addi $s2,$s1,2  # Sum of s1 and 2
        li $s3,3        #Assigning s3 to 3
        addi $s4,$s3,4  # Sum of s3 and 4
        li $s5,5        #Assigning s5 to 5
        addi $s6,$s5,6  # Sum s5 to 6
        addi $s7,$s6,7  # Sum s6 to 7
        addi $t1,$s7,3  # Sum s7 to 3
```

Line: 8 Column: 32 ☑ Show Line Numbers



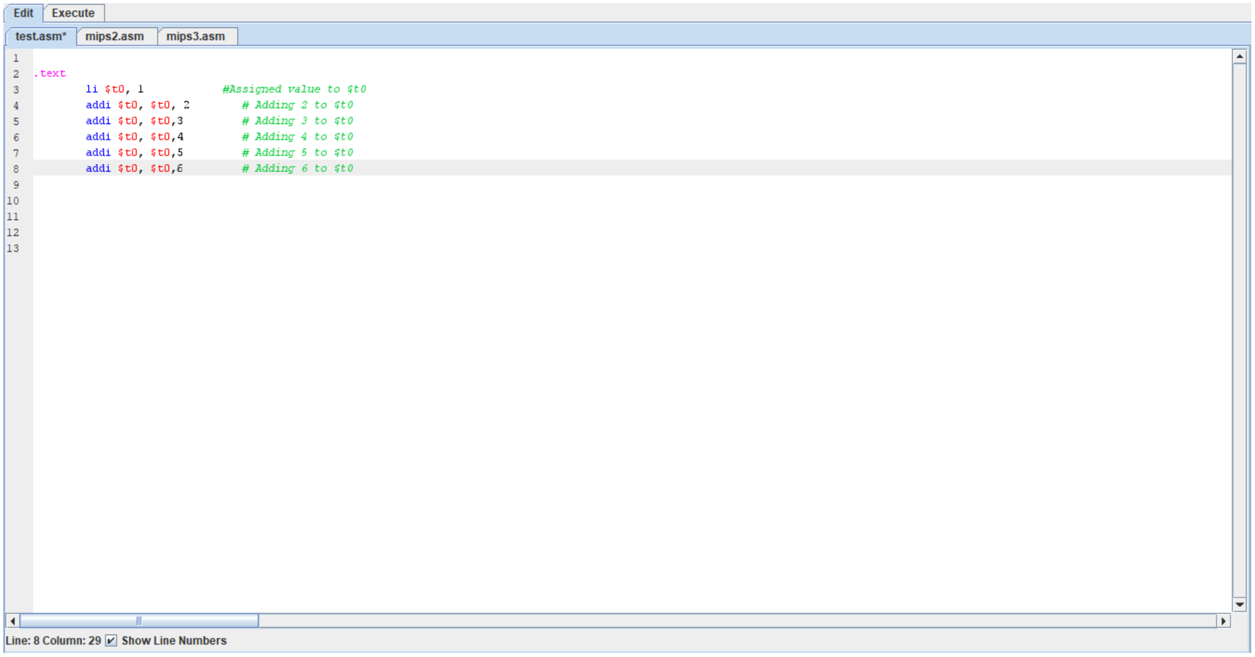
Name	Number	Value
$zero	0	0
$at	1	0
$v0	2	0
$v1	3	0
$a0	4	0
$a1	5	0
$a2	6	0
$a3	7	0
$t0	8	0
$t1	9	21
$t2	10	0
$t3	11	0
$t4	12	0
$t5	13	0
$t6	14	0
$t7	15	0
$s0	16	0
$s1	17	1
$s2	18	3
$s3	19	3
$s4	20	7
$s5	21	5
$s6	22	11
$s7	23	18
$t8	24	0
$t9	25	0
$k0	26	0
$k1	27	0
$gp	28	268468224
$sp	29	2147479548
$fp	30	0
$ra	31	0
pc		4194336
hi		0
lo		0

Problem 4:

Write a MIPS Assembly program to ADD the numbers: 1, 2, 3, 4, 5, 6 (as in ex–3)

- Use ONLY the instructions: addi and li ... and ONLY the register: $t0
- The result of the addition (in decimal) should be in register: $t0

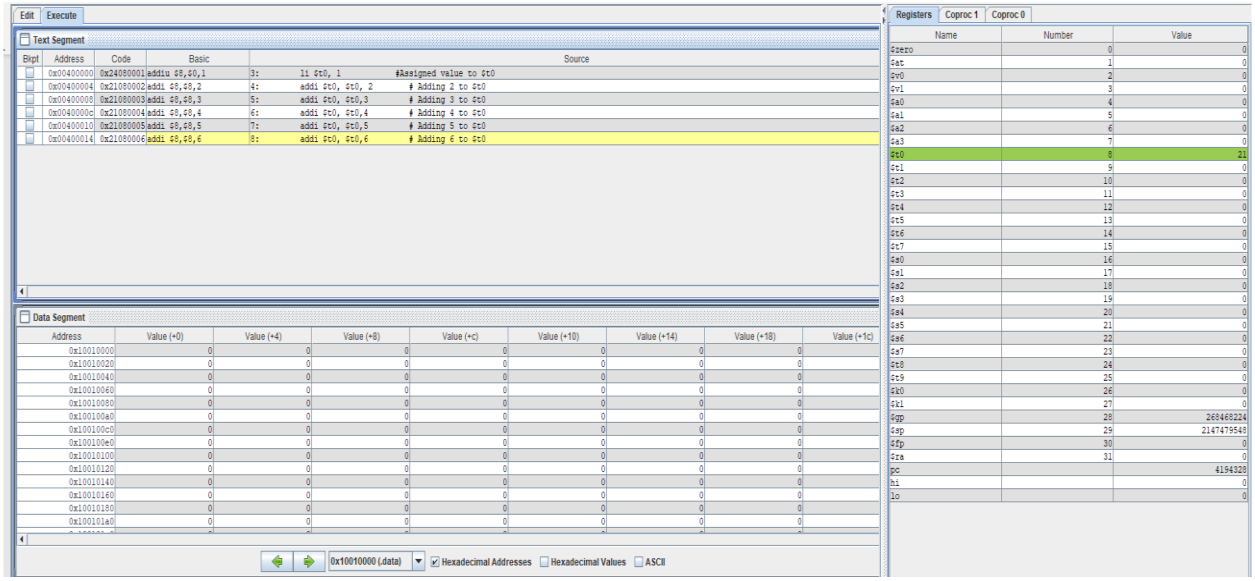
Rana, Karan



```

.text
        li $t0, 1            #Assigned value to $t0
        addi $t0, $t0, 2        # Adding 2 to $t0
        addi $t0, $t0,3         # Adding 3 to $t0
        addi $t0, $t0,4         # Adding 4 to $t0
        addi $t0, $t0,5         # Adding 5 to $t0
        addi $t0, $t0,6         # Adding 6 to $t0
```

Line: 8 Column: 29 ☑ Show Line Numbers



Text Segment

Bkpt	Address	Code	Basic	Source
□	0x00400000	0x24080001	addiu $8,$0,1	3: li $t0, 1 #Assigned value to $t0
□	0x00400004	0x21080002	addi $8,$8,2	4: addi $t0, $t0, 2 # Adding 2 to $t0
□	0x00400008	0x21080003	addi $8,$8,3	5: addi $t0, $t0,3 # Adding 3 to $t0
□	0x0040000c	0x21080004	addi $8,$8,4	6: addi $t0, $t0,4 # Adding 4 to $t0
□	0x00400010	0x21080005	addi $8,$8,5	7: addi $t0, $t0,5 # Adding 5 to $t0
□	0x00400014	0x21080006	addi $8,$8,6	8: addi $t0, $t0,6 # Adding 6 to $t0

Data Segment

Address	Value (+0)	Value (+4)	Value (+8)	Value (+c)	Value (+10)	Value (+14)	Value (+18)	Value (+1c)
0x10010000	0	0	0	0	0	0	0	
0x10010020	0	0	0	0	0	0	0	
0x10010040	0	0	0	0	0	0	0	
0x10010060	0	0	0	0	0	0	0	
0x10010080	0	0	0	0	0	0	0	
0x100100a0	0	0	0	0	0	0	0	
0x100100c0	0	0	0	0	0	0	0	
0x100100e0	0	0	0	0	0	0	0	
0x10010100	0	0	0	0	0	0	0	
0x10010120	0	0	0	0	0	0	0	
0x10010140	0	0	0	0	0	0	0	
0x10010160	0	0	0	0	0	0	0	
0x10010180	0	0	0	0	0	0	0	
0x100101a0	0	0	0	0	0	0	0	

0x10010000 (.data) ☑ Hexadecimal Addresses □ Hexadecimal Values □ ASCII

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	21
$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		4194328
hi		0
lo		0