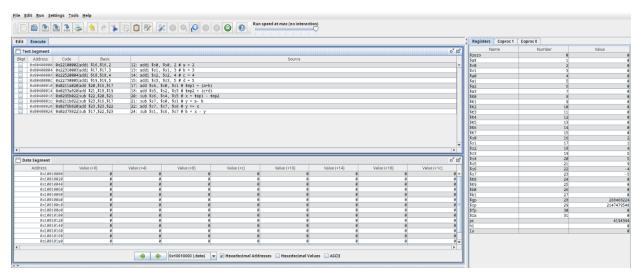
FELIPE AUGUSTO MORAIS SILVA

- 1- A
- 2- B
- 3- A
- 4- C
- 5- D
- 6- C
- 7- D
- 8- D
- 9- B
- 10- A
- 11- B
- 12- D
- __ _
- 13- B
- 14- C
- 15- B
- 16- B
- 17- A
- 18- C

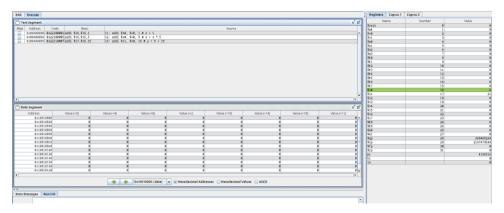
Programa 1 -

```
mips1.asm
             mips2.asm
 1 # programa 1 (add, addi, sub, lógicas)
 2 # a =2; b =3; c =4; d =5;
 3 \# x = (a+b) - (c+d); = -4
 4 # y = a - b + x; -5
 5 \# b = x - y; 1
 6
 7
   # inicio
 8
9
   .text
10 .globl main
11 main:
12 addi $s0, $s0, 2 # a = 2
13 addi $s1, $s1, 3 # b = 3
14 addi $s2, $s2, 4 # c = 4
   addi $s3, $s3, 5 # d = 5
15
16
   add $s4, $s0, $s1 # tmp1 = (a+b)
17
18
   add $s5, $s2, $s3 # tmp2 = (c+d)
19
20 sub $s6, $s4, $s5 # x = tmp1 - tmp2
   sub $s7, $s0, $s1 # y = a - b
21
   add $s7, $s7, $s6 # y += x
22
23
24 sub $s1, $s6, $s7 # b = x - y
25
26 #fim
```



Programa 2 -

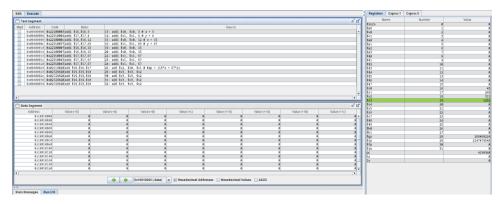
```
mips2.asm
 mips1.asm
   #programa 2
 2
 3
   \# x = 1;
   # y = 5*x + 15;
 4
 5
 6
   # inicio
 7
8
   .text
 9
   .globl main
10 main:
11 addi $s0, $s0, 1 # x = 1
12 addi $s0, $s0, 5 # x = x * 5
13 addi $s1, $s0, 15 # y = 5 + 15
14
15 #fim
```



Programa 3 -

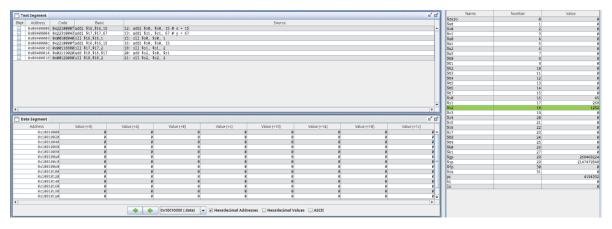
Eu instanciei as variaveis igual no programa de alto nível, mas não tem necessidade de fazer isso

```
mips3.asm
 mips1.asm
             mips2.asm
 1 #programa 3
 2
 3 \# x = 3;
 4  # y = 4; 
5  # z = (15*x + 67*y)*4
 6
 7 # inicio
8
9 .text
10 .globl main
11 main:
12 #initial values
13 addi $s0, $s0, 3 # x = 3
14 addi $s1, $s1, 4 # y = 4
15 addi $s0, $s0, 12 # x = 15
16 addi $s1, $s1, 63 # y = 67
17
18 # x = 15*x
19 addi $s0, $s0, 15
20 addi $s0, $s0, 15
21 # y = 67 * y
22 addi $s1, $s1, 67
23 addi $s1, $s1, 67
24 addi $s1, $s1, 67
25
26 add $s2, $s0, $s1 # tmp = (15*x + 67*y)
27
28 #tmp * 4
29 add $s3, $s3, $s2
30 add $s3, $s3, $s2
31 add $s3, $s3, $s2
32 add $s3, $s3, $s2
33
34 #fim
```



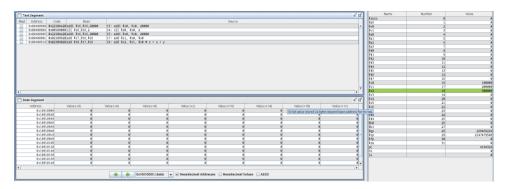
Programa 4 -

```
#programa 3
 2
 3 \# x = 3;
 4 # y = 4;
 5 \# z = (15*x + 67*y)*4
 6
 7 # inicio
 8
   .text
 9 .globl main
10 main:
11 #initial values
12 addi $s0, $s0, 15 # x = 15
13 addi $s1, $s1, 67 # y = 67
14 # 15 * x
15 sll $s0, $s0, 1
16 addi $s0, $s0, 15
17 #67 * x
18 sll $s1, $s1, 2
19 # (15*x + 67*y)*4
20 add $s2, $s0, $s1
21 sll $s2, $s2, 2
22 #fim
```



Programa 5 -

```
#programa 5
 2
 3 \# x = 1e5;
 4 \# y = 2e5;
 5 \# z = x + y
 6
7
   # inicio
8 .text
9 .globl main
10 main:
11 #initial values
12 # x = 1e5
13 addi $s0, $s0, 20000
14 sll $s0, $s0, 2
15 addi $s0, $s0, 20000
16 # y = 2e5
17
   add $s1, $s0, $s0
18
   add $s2, $s1, $s0 # z = x + y
19
20
21 #fim
22
```



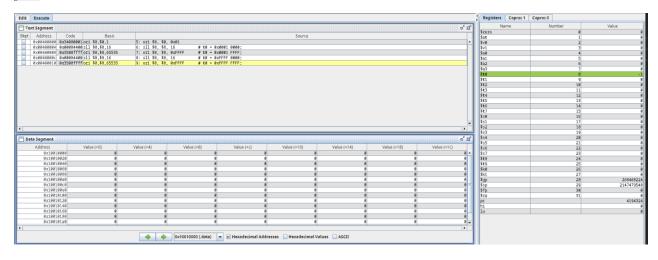
Programa 6

```
mips1.asm mips2.asm
                        mips3.asm
                                    mips4.asm
                                               mips5.asm
                                                            mips6.asm
 1 #programa 5
 2
 3 \# x = MAX\_INT;
 4 # y = 3e5;
 5 \# z = x - 4y
 7 # inicio
 8 .text
9 .globl main
10 main:
11 #initial values
12 \# x = MAX\_INT
13 ori $s0, $zero, 0x7fff # x = 0000 7fff
14 sll $s0, $s0, 16 # x = 7fff 0000
15 ori $s0, $s0, 0xffff # 7fff ffff
16
17 addi $s1, $s1, 25000 # y = 25000
18 add $s1, $s1, $s1 # y = 50000
19 add $t0, $s1, $s1 # tmp = 100000
20 add $s1, $s1, $s1 # y = 100000
21 add $s1, $s1, $s1 # y = 200000
22 add $s1, $s1, $t0 # y += tmp
23 sll $s1, $s1, 2 # 4y
24
25 sub $s2, $s0, $s1 # z = x - 4y
26
27 #fim
28
```

Text Segment										o* c	a'	Name	Number	Value
kpt Address Co	de Basic				Source							\$zero \$at	9	
	97fffori \$16.\$0.32767	13: ori \$50, \$zero, 0x7fff # x = 0000 7fff										508	1	
	98400 sll \$16.\$16.16	14: S11 \$50. \$50. 16 # x = 7fff 8000										\$v1	2	
	0.000 0.000											\$20	4	
0x0040000c 0x223161a8 addi \$17, \$17, 25000 17: addi \$s1, \$s1, 25000 # y = 25000												\$a1		
	x90400010 8x82318820 add \$17,\$17,\$17 18: add \$51, \$51, \$51 # y = 50000											\$a2	6	
8x88488814 8x82314828 add \$8,\$17,\$17 19: add \$t0, \$s1, \$s1 # tmp = 188888												\$a3	7	
0x88488818 8x823	8x98480818 8x82318828add \$17,\$17,\$17 28: add \$s1, \$s1, \$s1 # y - 188888												8	1986
3 8x9848081c 8x82318828add \$17,\$17,\$17 21: add \$s1, \$s1 # y = 280808												\$t1	9	
8x90400020 0x92288820add \$17,\$17,\$8 22: add \$s1, \$s1, \$s0 # y += tmp												\$t2	18	
0x00400024 0x00110880[sll \$17,\$17,2 23: sll \$51, \$51, 2 # 4y												\$t3	11	
0x00400028 0x021	19822 sub \$18,\$16,\$17	25: sub \$s2, \$s0	, \$s1 # z = x - 4y									St4	12	
												\$t5	13	
												\$ t 6	14	
												St7	15	
												\$50	16	21474836
											₹IB	\$s1	17	12000
)		\$52	18	21462836
												\$53	19	
Data Segment										o* [a' i	\$54	28	
Address	Value (+0)	Value (+4)	Value (+8)	Value (+c)	Value (+10)		Value (+14)	Value {+18	0	Value (+1c)		\$s5 \$s6	21	
8x18818888	VBIDE (YO)	VBIUC (14)	Value (YD)	value (+c)	A ABIDE (+10)	0	value (+ i →)	ABIOG (A10	,	value (+1c)		\$57	22	
8x18818928					9	9				0		5t8	23	
8x18818848	0				0	9			0	0		5t9	25	
8x18818868	0		9		9	9			0	0		\$k8	26	
8x18818888					9	9			9	9		\$k1	27	
8x188189a8	ě	ě	9		a	9	9		8	8		\$qp	28	2684682
8:c188188c8	P	8	9		8	9	9		B	P		\$sp	29	21474795
8x188188e8	P	8	9		8	9	9		8	8		\$fp	38	2177772
8:c18818189	9	9	8		8	9	9		9	9		Sia	31	
8:c18818128	9		8		9	9	0		8	8		pc		41943
8:c18818148	9	8	8		8	9	9		8	9		hi		
8:c18818168	9	8	8		8	9	9		8	9		10		
8x18818188	9	8	8		9	8	9		8	9				
8::188181a8	9	9	9		8	9	9		9	8	₹ 8			
)				
		A .												
		→ → → → → → → → → → → → → → → → → → →	0x10010000 (.data)	Mexadecimal A	ddresses 🔲 Hexadecin	nai Values	ASCII							

Programa 7

```
mips1.asm mips2.asm mips3.asm
                                  mips4.asm mips5.asm mips6.asm mips7.asm
 1 # inicio
2 .text
3 .globl main
4 main:
5 ori $8, $0, 0x01
6 sll $8, $8, 16
                         # t0 = 0x0001 0000;
7 ori $8, $8, 0xFFFF
                        # t0 = 0x0001 FFFF;
8 sll $8, $8, 16
                        # t0 = 0xFFFF 0000;
9 ori $8, $8, 0xFFFF
                        # t0 = 0xFFFF FFFF;
10 #fim
11
```



Programa 8

```
mips1.asm mips2.asm mips3.asm mips4.asm mips5.asm mips6.asm mips7.asm mips8.asm*
 1 # inicio
2 .text
3 .globl main
4 main:
 5 ori $8, $zero, 0x1234 #8 = 0000 1234
6 sll $8, $8, 16 #8 = 1234 0000
7 ori $8, $8, 0x5678 #8 = 1234 5678
                             #8 = 1234 5678
8 srl $9, $8, 24
                             #9 = 0000 0034
9 sll $10, $8, 8
                             #10 = 3456 7800
                             #10 = 0000 0034
10 srl $10, $10, 24
                            #10 - 0000 0034
#11 = 5678 0000
#11 = 0000 0078
#12 = 7800 0000
#12 = 0000 0078
11 sll $11, $8, 16
12 srl $11, $11, 24
13 sll $12, $8, 24
14 srl $12, $12, 24
15
16 #fim
17
```

