

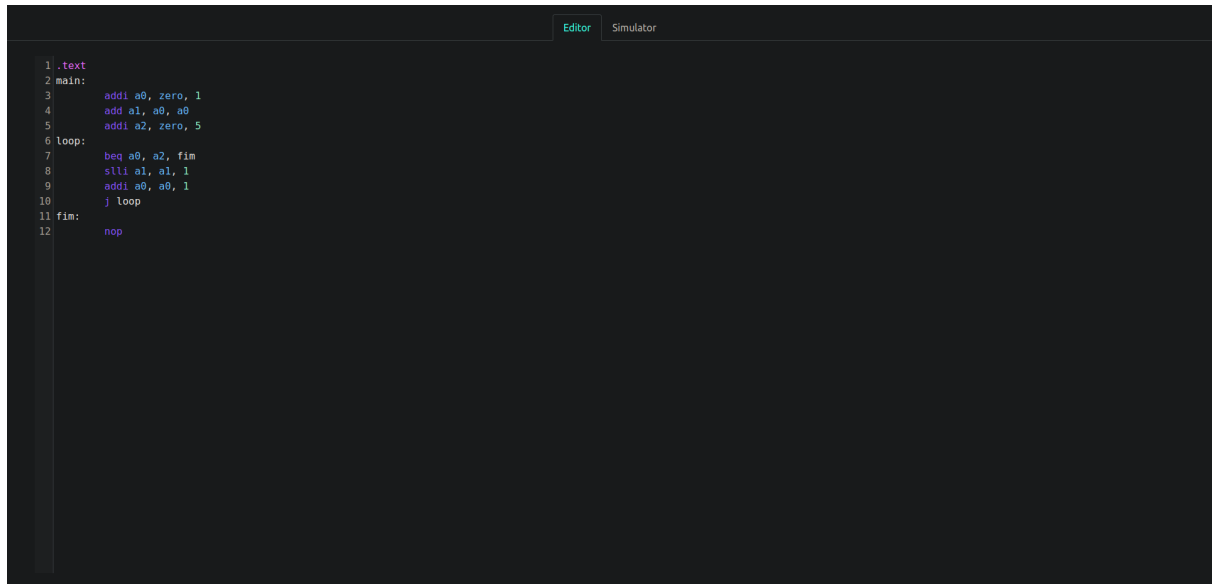
ROTEIRO 6

Problema 1:

1. Inicia a0 com valor 1, a1 com a0 + a0, 2, e a2 com 5. O código entra em um laço de repetição até que a0 e a2 sejam iguais, enquanto isso não for verdade, incrementa-se a0 e multiplica a1 por 2.
2. a0: 5
a1: 32
a2: 5
3. loop: 16
fim: -12

Esses valores são o resultado da multiplicação de 4 (tamanho de cada endereço) e o valor do número de endereços que deverão ser saltados para ir para o comando destino.

4. código em Assembly



```
1 .text
2 main:
3     addi a0, zero, 1
4     add a1, a0, a0
5     addi a2, zero, 5
6 loop:
7     beq a0, a2, fim
8     slli a1, a1, 1
9     addi a0, a0, 1
10    j loop
11 fim:
12    nop
```

código de máquina armazenado na memória

Editor

Simulator

Run

Step

Prev

Reset

Dump

Machine Code	Basic Code	Original Code
0x00100513	addi x10 x0 1	addi a0, zero, 1
0x00a505b3	add x11 x10 x10	add a1, a0, a0
0x00500613	addi x12 x0 5	addi a2, zero, 5
0x00c50863	beq x10 x12 16	beq a0, a2, fim
0x00159593	slli x11 x11 1	slli a1, a1, 1
0x00150513	addi x10 x10 1	addi a0, a0, 1
0xff5ff06f	jal x0 -12	j loop
0x00000013	addi x0 x0 0	nop

console output

Registers

Memory

Address	+0	+1	+2	+3
0x00000018	6f	f0	5f	ff
0x00000014	13	05	15	00
0x00000010	93	95	15	00
0x0000000c	63	08	c5	00
0x00000008	13	06	50	00
0x00000004	b3	05	a5	00
0x00000000	13	05	10	00
-----	--	--	--	--
-----	--	--	--	--
-----	--	--	--	--
-----	--	--	--	--
-----	--	--	--	--
-----	--	--	--	--
-----	--	--	--	--

Jump to -- choose -- Up Down

Display Settings Hex

registrador x10 (a0) recebeu valor 1 em decimal

Editor

Simulator

Run

Step

Prev

Reset

Dump

Machine Code	Basic Code	Original Code
0x00100513	addi x10 x0 1	addi a0, zero, 1
0x00a505b3	add x11 x10 x10	add a1, a0, a0
0x00500613	addi x12 x0 5	addi a2, zero, 5
0x00c50863	beq x10 x12 16	beq a0, a2, fim
0x00159593	slli x11 x11 1	slli a1, a1, 1
0x00150513	addi x10 x10 1	addi a0, a0, 1
0xff5ff06f	jal x0 -12	j loop
0x00000013	addi x0 x0 0	nop

console output

Registers

Memory

zero	0x00000000
ra (x1)	0x00000000
sp (x2)	0xffffffff
gp (x3)	0x10000000
tp (x4)	0x00000000
t0 (x5)	0x00000000
t1 (x6)	0x00000000
t2 (x7)	0x00000000
s0 (x8)	0x00000000
s1 (x9)	0x00000000
a0 (x10)	0x00000001
a1 (x11)	0x00000000
a2 (x12)	0x00000000
a3 (x13)	0x00000000
a4 (x14)	0x00000000
a5 (x15)	0x00000000

Display Settings Hex

registrador x11 (a1) recebeu valor 2 em decimal

Editor

Simulator

RunStepPrevResetDump

Machine Code	Basic Code	Original Code
0x00100513	addi x10 x0 1	addi a0, zero, 1
0x00a505b3	add x11 x10 x10	add a1, a0, a0
0x00500613	addi x12 x0 5	addi a2, zero, 5
0x00c50863	beq x10 x12 16	beq a0, a2, fim
0x00159593	slli x11 x11 1	slli a1, a1, 1
0x00150513	addi x10 x10 1	addi a0, a0, 1
0xff5ff06f	jal x0 -12	j loop
0x00000013	addi x0 x0 0	nop

console output

RegistersMemory

zero	0x00000000
ra (x1)	0x00000000
sp (x2)	0x7ffffff0
gp (x3)	0x10000000
tp (x4)	0x00000000
t0 (x5)	0x00000000
t1 (x6)	0x00000000
t2 (x7)	0x00000000
s0 (x8)	0x00000000
s1 (x9)	0x00000000
a0 (x10)	0x00000001
a1 (x11)	0x00000002
a2 (x12)	0x00000000
a3 (x13)	0x00000000
a4 (x14)	0x00000000
a5 (x15)	0x00000000

Display SettingsHex

registrador x12 (a2) recebeu valor 5 em decimal

Editor

Simulator

RunStepPrevResetDump

Machine Code	Basic Code	Original Code
0x00100513	addi x10 x0 1	addi a0, zero, 1
0x00a505b3	add x11 x10 x10	add a1, a0, a0
0x00500613	addi x12 x0 5	addi a2, zero, 5
0x00c50863	beq x10 x12 16	beq a0, a2, fim
0x00159593	slli x11 x11 1	slli a1, a1, 1
0x00150513	addi x10 x10 1	addi a0, a0, 1
0xff5ff06f	jal x0 -12	j loop
0x00000013	addi x0 x0 0	nop

console output

RegistersMemory

zero	0x00000000
ra (x1)	0x00000000
sp (x2)	0x7ffffff0
gp (x3)	0x10000000
tp (x4)	0x00000000
t0 (x5)	0x00000000
t1 (x6)	0x00000000
t2 (x7)	0x00000000
s0 (x8)	0x00000000
s1 (x9)	0x00000000
a0 (x10)	0x00000001
a1 (x11)	0x00000002
a2 (x12)	0x00000005
a3 (x13)	0x00000000
a4 (x14)	0x00000000
a5 (x15)	0x00000000

Display SettingsHex

como a condição do **beq** não era verdade, não houve o salto e seguiu o fluxo normal de execução

Editor

Simulator

RunStepPrevResetDump

Machine Code	Basic Code	Original Code
0x00100513	addi x10 x0 1	addi a0, zero, 1
0x00a505b3	add x11 x10 x10	add a1, a0, a0
0x00500613	addi x12 x0 5	addi a2, zero, 5
0x00c50863	beq x10 x12 16	beq a0, a2, fim
0x00150593	slli x11 x11 1	slli a1, a1, 1
0x00150513	addi x10 x10 1	addi a0, a0, 1
0xff5ff06f	jal x0 -12	j loop
0x00000013	addi x0 x0 0	nop

console output

RegistersMemory

zero	0x00000000
ra (x1)	0x00000000
sp (x2)	0x7fffffff0
gp (x3)	0x10000000
tp (x4)	0x00000000
t0 (x5)	0x00000000
t1 (x6)	0x00000000
t2 (x7)	0x00000000
s0 (x8)	0x00000000
s1 (x9)	0x00000000
a0 (x10)	0x00000001
a1 (x11)	0x00000002
a2 (x12)	0x00000005
a3 (x13)	0x00000000
a4 (x14)	0x00000000
a5 (x15)	0x00000000

Display SettingsHex

o valor do registrador a1 sofreu um shift para a esquerda de uma casa

Editor

Simulator

RunStepPrevResetDump

Machine Code	Basic Code	Original Code
0x00100513	addi x10 x0 1	addi a0, zero, 1
0x00a505b3	add x11 x10 x10	add a1, a0, a0
0x00500613	addi x12 x0 5	addi a2, zero, 5
0x00c50863	beq x10 x12 16	beq a0, a2, fim
0x00150593	slli x11 x11 1	slli a1, a1, 1
0x00150513	addi x10 x10 1	addi a0, a0, 1
0xff5ff06f	jal x0 -12	j loop
0x00000013	addi x0 x0 0	nop

console output

RegistersMemory

zero	0x00000000
ra (x1)	0x00000000
sp (x2)	0x7fffffff0
gp (x3)	0x10000000
tp (x4)	0x00000000
t0 (x5)	0x00000000
t1 (x6)	0x00000000
t2 (x7)	0x00000000
s0 (x8)	0x00000000
s1 (x9)	0x00000000
a0 (x10)	0x00000001
a1 (x11)	0x00000004
a2 (x12)	0x00000005
a3 (x13)	0x00000000
a4 (x14)	0x00000000
a5 (x15)	0x00000000

Display SettingsHex

o valor do registrador a0 sofreu um incremento

Editor

Simulator

RunStepPrevResetDump

Machine Code	Basic Code	Original Code
0x00100513	addi x10 x0 1	addi a0, zero, 1
0x00a505b3	add x11 x10 x10	add a1, a0, a0
0x00500613	addi x12 x0 5	addi a2, zero, 5
0x00c50863	beq x10 x12 16	beq a0, a2, fim
0x00159593	slli x11 x11 1	slli a1, a1, 1
0x00150513	addi x10 x10 1	addi a0, a0, 1
0xff5ff06f	jal x0 -12	j loop
0x00000013	addi x0 x0 0	nop

console output

RegistersMemory

zero	0x00000000
ra (x1)	0x00000000
sp (x2)	0x7fffffff0
gp (x3)	0x10000000
tp (x4)	0x00000000
t0 (x5)	0x00000000
t1 (x6)	0x00000000
t2 (x7)	0x00000000
s0 (x8)	0x00000000
s1 (x9)	0x00000000
a0 (x10)	0x00000002
a1 (x11)	0x00000004
a2 (x12)	0x00000005
a3 (x13)	0x00000000
a4 (x14)	0x00000000
a5 (x15)	0x00000000

Display SettingsHex

o comando faz o código voltar para o início do loop

Editor

Simulator

RunStepPrevResetDump

Machine Code	Basic Code	Original Code
0x00100513	addi x10 x0 1	addi a0, zero, 1
0x00a505b3	add x11 x10 x10	add a1, a0, a0
0x00500613	addi x12 x0 5	addi a2, zero, 5
0x00c50863	beq x10 x12 16	beq a0, a2, fim
0x00159593	slli x11 x11 1	slli a1, a1, 1
0x00150513	addi x10 x10 1	addi a0, a0, 1
0xff5ff06f	jal x0 -12	j loop
0x00000013	addi x0 x0 0	nop

console output

RegistersMemory

zero	0x00000000
ra (x1)	0x00000000
sp (x2)	0x7fffffff0
gp (x3)	0x10000000
tp (x4)	0x00000000
t0 (x5)	0x00000000
t1 (x6)	0x00000000
t2 (x7)	0x00000000
s0 (x8)	0x00000000
s1 (x9)	0x00000000
a0 (x10)	0x00000002
a1 (x11)	0x00000004
a2 (x12)	0x00000005
a3 (x13)	0x00000000
a4 (x14)	0x00000000
a5 (x15)	0x00000000

Display SettingsHex

o loop é executado seguindo os mesmo passos exibidos até que a condição do **beq** seja verdadeira.

quando isso finalmente acontece, o código dá um jump para fim, que é a execução de um nop

Editor

Simulator

Run

Step

Prev

Reset

Dump

Machine Code	Basic Code	Original Code
0x00100513	addi x10 x0 1	addi a0, zero, 1
0x00a505b3	add x11 x10 x10	add a1, a0, a0
0x00500613	addi x12 x0 5	addi a2, zero, 5
0x00c50863	beq x10 x12 16	beq a0, a2, f1m
0x00159593	slli x11 x11 1	slli a1, a1, 1
0x00150513	addi x10 x10 1	addi a0, a0, 1
0xff5ff06f	jal x0 -12	j loop
0x00000013	addi x0 x0 0	nop

console output

Registers

Memory

zero	0x00000000
ra (x1)	0x00000000
sp (x2)	0x7fffffff0
gp (x3)	0x10000000
tp (x4)	0x00000000
t0 (x5)	0x00000000
t1 (x6)	0x00000000
t2 (x7)	0x00000000
s0 (x8)	0x00000000
s1 (x9)	0x00000000
a0 (x10)	0x00000005
a1 (x11)	0x00000020
a2 (x12)	0x00000005
a3 (x13)	0x00000000
a4 (x14)	0x00000000
a5 (x15)	0x00000000

Display Settings

Hex

e o código é finalizado

Editor

Simulator

Run

Step

Prev

Reset

Dump

Machine Code	Basic Code	Original Code
0x00100513	addi x10 x0 1	addi a0, zero, 1
0x00a505b3	add x11 x10 x10	add a1, a0, a0
0x00500613	addi x12 x0 5	addi a2, zero, 5
0x00c50863	beq x10 x12 16	beq a0, a2, f1m
0x00159593	slli x11 x11 1	slli a1, a1, 1
0x00150513	addi x10 x10 1	addi a0, a0, 1
0xff5ff06f	jal x0 -12	j loop
0x00000013	addi x0 x0 0	nop

console output

Registers

Memory

zero	0x00000000
ra (x1)	0x00000000
sp (x2)	0x7fffffff0
gp (x3)	0x10000000
tp (x4)	0x00000000
t0 (x5)	0x00000000
t1 (x6)	0x00000000
t2 (x7)	0x00000000
s0 (x8)	0x00000000
s1 (x9)	0x00000000
a0 (x10)	0x00000005
a1 (x11)	0x00000020
a2 (x12)	0x00000005
a3 (x13)	0x00000000
a4 (x14)	0x00000000
a5 (x15)	0x00000000

Display Settings

Hex

Problema 2

Código em Assembly:

addi, x10, x0, 2

addi, x11, x0, 4

beq, x10, x11, b

add, x12, x10, x10

jal, x0, j

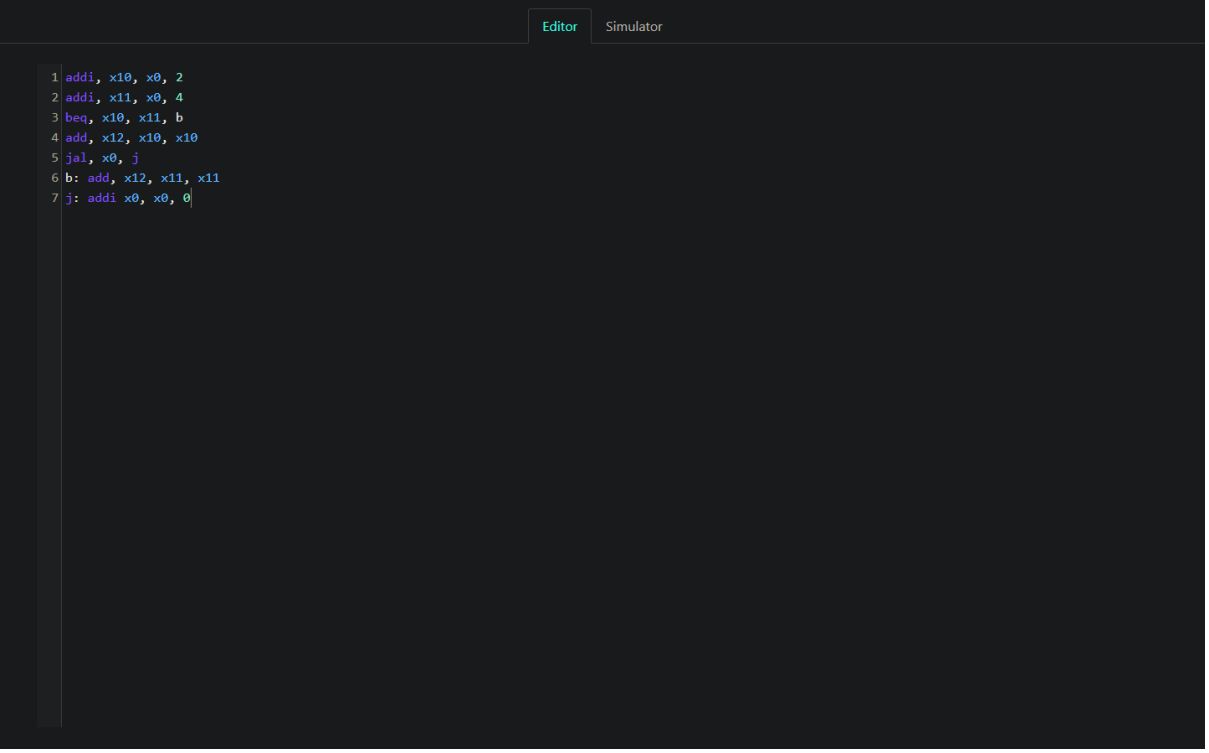
b: add, x12, x11, x11

j: addi x0, x0, 0

O código salva no registrador x10 o valor 2 e no x11 o valor 4, compara esses dois valores para, caso sejam iguais, salvar no registrador x12 o valor de duas vezes x11. Como não era verdade, ele foi para a próxima linha normalmente e salvou em x12 o valor de duas vezes o valor de x10 e pulou para a execução nop (fim do código).

Os registradores utilizados foram: x0, x10, x11 e x12

O conteúdo dos registradores foi, respectivamente: 0, 2, 4 e 4.

The image shows a screenshot of a web-based MIPS assembly simulator. At the top, there are two tabs: 'Editor' (which is active) and 'Simulator'. The 'Editor' tab contains a list of seven assembly instructions, each preceded by a line number from 1 to 7. The instructions are: 1. addi, x10, x0, 2; 2. addi, x11, x0, 4; 3. beq, x10, x11, b; 4. add, x12, x10, x10; 5. jal, x0, j; 6. b: add, x12, x11, x11; 7. j: addi x0, x0, 0. The code is displayed in a dark-themed editor with syntax highlighting. The 'Simulator' tab is currently empty.

```
1 addi, x10, x0, 2
2 addi, x11, x0, 4
3 beq, x10, x11, b
4 add, x12, x10, x10
5 jal, x0, j
6 b: add, x12, x11, x11
7 j: addi x0, x0, 0
```

Editor

Simulator

Run

Step

Prev

Reset

Dump

Machine Code	Basic Code	Original Code
0x00200513	addi x10 x0 2	addi, x10, x0, 2
0x00400593	addi x11 x0 4	addi, x11, x0, 4
0x00b50663	beq x10 x11 12	beq, x10, x11, b
0x00a50633	add x12 x10 x10	add, x12, x10, x10
0x0080006f	jal x0 8	jal, x0, j
0x00b58633	add x12 x11 x11	b: add, x12, x11, x11
0x00000013	addi x0 x0 0	j: addi x0, x0, 0

console output

Registers

Memory

Address	+0	+1	+2	+3
0x00000018	13	00	00	00
0x00000014	33	86	b5	00
0x00000010	6f	00	80	00
0x0000000c	33	06	a5	00
0x00000008	63	06	b5	00
0x00000004	93	05	40	00
0x00000000	13	05	20	00
-----	--	--	--	--
-----	--	--	--	--
-----	--	--	--	--
-----	--	--	--	--
-----	--	--	--	--
-----	--	--	--	--

Jump to -- choose -- Up Down

Display Settings Hex

Antes do primeiro comando:

Editor

Simulator

Run

Step

Prev

Reset

Dump

Machine Code	Basic Code	Original Code
0x00200513	addi x10 x0 2	addi, x10, x0, 2
0x00400593	addi x11 x0 4	addi, x11, x0, 4
0x00b50663	beq x10 x11 12	beq, x10, x11, b
0x00a50633	add x12 x10 x10	add, x12, x10, x10
0x0080006f	jal x0 8	jal, x0, j
0x00b58633	add x12 x11 x11	b: add, x12, x11, x11
0x00000013	addi x0 x0 0	j: addi x0, x0, 0

console output

Registers

Memory

zero	0x00000000
ra (x1)	0x00000000
sp (x2)	0x7fffffff0
gp (x3)	0x10000000
tp (x4)	0x00000000
t0 (x5)	0x00000000
t1 (x6)	0x00000000
t2 (x7)	0x00000000
s0 (x8)	0x00000000
s1 (x9)	0x00000000
a0 (x10)	0x00000000
a1 (x11)	0x00000000
a2 (x12)	0x00000000
a3 (x13)	0x00000000

Display Settings Hex

Após o primeiro comando: x10 recebe 2

Editor

Simulator

Run

Step

Prev

Reset

Dump

Machine Code	Basic Code	Original Code
0x00200513	addi x10 x0 2	addi, x10, x0, 2
0x00400593	addi x11 x0 4	addi, x11, x0, 4
0x00b50663	beq x10 x11 12	beq, x10, x11, b
0x00a50633	add x12 x10 x10	add, x12, x10, x10
0x0080006f	jal x0 8	jal, x0, j
0x00b58633	add x12 x11 x11	b: add, x12, x11, x11
0x00000013	addi x0 x0 0	j: addi x0, x0, 0

console output

Registers

Memory

zero	0x00000000
ra (x1)	0x00000000
sp (x2)	0x7fffffff0
gp (x3)	0x10000000
tp (x4)	0x00000000
t0 (x5)	0x00000000
t1 (x6)	0x00000000
t2 (x7)	0x00000000
s0 (x8)	0x00000000
s1 (x9)	0x00000000
a0 (x10)	0x00000002
a1 (x11)	0x00000000
a2 (x12)	0x00000000
a3 (x13)	0x00000000

Display Settings

Hex

Após o segundo comando: x11 recebe 4

Editor

Simulator

Run

Step

Prev

Reset

Dump

Machine Code	Basic Code	Original Code
0x00200513	addi x10 x0 2	addi, x10, x0, 2
0x00400593	addi x11 x0 4	addi, x11, x0, 4
0x00b50663	beq x10 x11 12	beq, x10, x11, b
0x00a50633	add x12 x10 x10	add, x12, x10, x10
0x0080006f	jal x0 8	jal, x0, j
0x00b58633	add x12 x11 x11	b: add, x12, x11, x11
0x00000013	addi x0 x0 0	j: addi x0, x0, 0

console output

Registers

Memory

zero	0x00000000
ra (x1)	0x00000000
sp (x2)	0x7fffffff0
gp (x3)	0x10000000
tp (x4)	0x00000000
t0 (x5)	0x00000000
t1 (x6)	0x00000000
t2 (x7)	0x00000000
s0 (x8)	0x00000000
s1 (x9)	0x00000000
a0 (x10)	0x00000002
a1 (x11)	0x00000004
a2 (x12)	0x00000000
a3 (x13)	0x00000000

Display Settings

Hex

Após o terceiro comando: pula 12 bits se x10 e x11 são iguais (não são e por isso nada é feito)

Editor

Simulator

Run

Step

Prev

Reset

Dump

Machine Code	Basic Code	Original Code
0x00200513	addi x10 x0 2	addi, x10, x0, 2
0x00400593	addi x11 x0 4	addi, x11, x0, 4
0x00b50663	beq x10 x11 12	beq, x10, x11, b
0x00a50633	add x12 x10 x10	add, x12, x10, x10
0x0080006f	jal x0 8	jal, x0, j
0x00b58633	add x12 x11 x11	b: add, x12, x11, x11
0x00000013	addi x0 x0 0	j: addi x0, x0, 0

console output

Registers

Memory

zero

0x00000000

ra (x1)

0x00000000

sp (x2)

0x7fffffff0

gp (x3)

0x10000000

tp (x4)

0x00000000

t0 (x5)

0x00000000

t1 (x6)

0x00000000

t2 (x7)

0x00000000

s0 (x8)

0x00000000

s1 (x9)

0x00000000

a0 (x10)

0x00000002

a1 (x11)

0x00000004

a2 (x12)

0x00000000

a3 (x13)

0x00000000

Display Settings

Hex

Após o quarto comando: x12 recebe duas vezes o valor de x10

Editor

Simulator

Run

Step

Prev

Reset

Dump

Machine Code	Basic Code	Original Code
0x00200513	addi x10 x0 2	addi, x10, x0, 2
0x00400593	addi x11 x0 4	addi, x11, x0, 4
0x00b50663	beq x10 x11 12	beq, x10, x11, b
0x00a50633	add x12 x10 x10	add, x12, x10, x10
0x0080006f	jal x0 8	jal, x0, j
0x00b58633	add x12 x11 x11	b: add, x12, x11, x11
0x00000013	addi x0 x0 0	j: addi x0, x0, 0

console output

Registers

Memory

zero

0x00000000

ra (x1)

0x00000000

sp (x2)

0x7fffffff0

gp (x3)

0x10000000

tp (x4)

0x00000000

t0 (x5)

0x00000000

t1 (x6)

0x00000000

t2 (x7)

0x00000000

s0 (x8)

0x00000000

s1 (x9)

0x00000000

a0 (x10)

0x00000002

a1 (x11)

0x00000004

a2 (x12)

0x00000004

a3 (x13)

0x00000000

Display Settings

Hex

Após o quinto comando: Pula 8 bits (pula para o segundo endereço abaixo)

Editor

Simulator

Run

Step

Prev

Reset

Dump

Machine Code	Basic Code	Original Code
0x00200513	addi x10 x0 2	addi, x10, x0, 2
0x00400593	addi x11 x0 4	addi, x11, x0, 4
0x00b50663	beq x10 x11 12	beq, x10, x11, b
0x00a50633	add x12 x10 x10	add, x12, x10, x10
0x0080006f	jal x0 8	jal, x0, j
0x00b58633	add x12 x11 x11	b: add, x12, x11, x11
0x00000013	addi x0 x0 0	j: addi x0, x0, 0

console output

Registers

Memory

zero

0x00000000

ra (x1)

0x00000000

sp (x2)

0x7fffffff0

gp (x3)

0x10000000

tp (x4)

0x00000000

t0 (x5)

0x00000000

t1 (x6)

0x00000000

t2 (x7)

0x00000000

s0 (x8)

0x00000000

s1 (x9)

0x00000000

a0 (x10)

0x00000002

a1 (x11)

0x00000004

a2 (x12)

0x00000004

a3 (x13)

0x00000000

Display Settings

Hex

Após o sexto comando: executa o comando nop (fim do código)

Editor

Simulator

Run

Step

Prev

Reset

Dump

Machine Code	Basic Code	Original Code
0x00200513	addi x10 x0 2	addi, x10, x0, 2
0x00400593	addi x11 x0 4	addi, x11, x0, 4
0x00b50663	beq x10 x11 12	beq, x10, x11, b
0x00a50633	add x12 x10 x10	add, x12, x10, x10
0x0080006f	jal x0 8	jal, x0, j
0x00b58633	add x12 x11 x11	b: add, x12, x11, x11
0x00000013	addi x0 x0 0	j: addi x0, x0, 0

console output

Registers

Memory

zero

0x00000000

ra (x1)

0x00000000

sp (x2)

0x7fffffff0

gp (x3)

0x10000000

tp (x4)

0x00000000

t0 (x5)

0x00000000

t1 (x6)

0x00000000

t2 (x7)

0x00000000

s0 (x8)

0x00000000

s1 (x9)

0x00000000

a0 (x10)

0x00000002

a1 (x11)

0x00000004

a2 (x12)

0x00000004

a3 (x13)

0x00000000

Display Settings

Hex