

A)

```
1 0x00200513 -> addi a0, zero, 2
2 0x00400593 -> addi a1, zero, 4
3 0x00b50663 -> beq a0, a1, fim
4 0x00a50533 -> add a0, a0, a0
5 0xff9ff06f -> j loop
6 0x00b58633 -> add a2, a1, a1
7 0x00000013 -> nop
8
```

B)

Editor Simulator

```
1 main:
2
3     addi a0, zero, 2    # variavel = 2
4     addi a1, zero, 4    # variavel = 4
5
6 loop:
7
8     beq a0, a1, fim     # condição de saída caso a0 for igual a a1
9     add a0, a0, a0      # dobra o valor de a0
10    j loop              # faz o loop
11
12 fim:
13
14    add a2, a1, a1       # soma a1 com ele mesmo e salva em a2
15    nop                 # saída
```

C)

Registadores gerais: a0, a1, a2.  
Registrador zero: zero.

D) a0 = 4, a1 = 4, a2 = 8.

Editor Simulator

Run Step Prev Reset Dump

Machine Code	Basic Code	Original Code
0x00200513	addi x10 x0 2	addi a0, zero, 2 # variavel = 2
0x00400593	addi x11 x0 4	addi a1, zero, 4 # variavel = 4
0x00b50663	beq x10 x11 12	beq a0, a1, fim # condição de saída caso a0 for igual a a1
0x00a50533	add x10 x10 x10	add a0, a0, a0 # dobra o valor de a0
0xff9ff06f	jal x0 -8	j loop # faz o loop
0x00b58633	add x12 x11 x11	add a2, a1, a1 # soma a1 com ele mesmo e salva em a2
0x00000013	addi x0 x0 0	nop # saída

console output

Registers Memory

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)	0x00000004
a1 (x11)	0x00000004
a2 (x12)	0x00000008
a3 (x13)	0x00000000
a4 (x14)	0x00000000
a5 (x15)	0x00000000

E)

Primeira Variável

Editor

Simulator

RunStepPrevResetDump

Machine Code	Basic Code	Original Code
0x00200513	addi x10 x0 2	addi a0, zero, 2 # variavel = 2
0x00400593	addi x11 x0 4	addi a1, zero, 4 # variavel = 4
0x00b50663	beq x10 x11 12	beq a0, a1, fim # condição de saída caso a0 for igual a a1
0x00a50533	add x10 x10 x10	add a0, a0, a0 # dobra o valor de a0
0xff9ff06f	jal x0 -8	j loop # faz o loop
0x00b50633	add x12 x11 x11	add a2, a1, a1 # soma a1 com ele mesmo e salva em a2
0x00000013	addi x0 x0 0	nop # saída

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)

0x00000000

a2 (x12)

0x00000000

a3 (x13)

0x00000000

a4 (x14)

0x00000000

a5 (x15)

0x00000000

Display SettingsFlex

Segunda Variável

Editor

Simulator

RunStepPrevResetDump

Machine Code	Basic Code	Original Code
0x00200513	addi x10 x0 2	addi a0, zero, 2 # variavel = 2
0x00400593	addi x11 x0 4	addi a1, zero, 4 # variavel = 4
0x00b50663	beq x10 x11 12	beq a0, a1, fim # condição de saída caso a0 for igual a a1
0x00a50533	add x10 x10 x10	add a0, a0, a0 # dobra o valor de a0
0xff9ff06f	jal x0 -8	j loop # faz o loop
0x00b50633	add x12 x11 x11	add a2, a1, a1 # soma a1 com ele mesmo e salva em a2
0x00000013	addi x0 x0 0	nop # saída

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)

0x00000000

a3 (x13)

0x00000000

a4 (x14)

0x00000000

a5 (x15)

0x00000000

Display SettingsFlex

Iteração (loop):

Editor

Simulator

RunStepPrevResetDump

Machine Code	Basic Code	Original Code
0x00200513	addi x10 x0 2	addi a0, zero, 2 # variavel = 2
0x00400593	addi x11 x0 4	addi a1, zero, 4 # variavel = 4
0x00b50663	beq x10 x11 12	beq a0, a1, fim # condição de saída caso a0 for igual a a1
0x00a50533	add x10 x10 x10	add a0, a0, a0 # dobra o valor de a0
0xff9ff06f	jal x0 -8	j loop # faz o loop
0x00b50633	add x12 x11 x11	add a2, a1, a1 # soma a1 com ele mesmo e salva em a2
0x00000013	addi x0 x0 0	nop # saída

console output

RegistersMemory

zero

ra (x1)

sp (x2)

gp (x3)

tp (x4)

t0 (x5)

t1 (x6)

t2 (x7)

s0 (x8)

s1 (x9)

a0 (x10)

a1 (x11)

a2 (x12)

a3 (x13)

a4 (x14)

a5 (x15)

Display SettingsHex

Terceira variável:

Editor

Simulator

RunStepPrevResetDump

Machine Code	Basic Code	Original Code
0x00200513	addi x10 x0 2	addi a0, zero, 2 # variavel = 2
0x00400593	addi x11 x0 4	addi a1, zero, 4 # variavel = 4
0x00b50663	beq x10 x11 12	beq a0, a1, fim # condição de saída caso a0 for igual a a1
0x00a50533	add x10 x10 x10	add a0, a0, a0 # dobra o valor de a0
0xff9ff06f	jal x0 -8	j loop # faz o loop
0x00b50633	add x12 x11 x11	add a2, a1, a1 # soma a1 com ele mesmo e salva em a2
0x00000013	addi x0 x0 0	nop # saída

console output

RegistersMemory

zero

ra (x1)

sp (x2)

gp (x3)

tp (x4)

t0 (x5)

t1 (x6)

t2 (x7)

s0 (x8)

s1 (x9)

a0 (x10)

a1 (x11)

a2 (x12)

a3 (x13)

a4 (x14)

a5 (x15)

Display SettingsHex

Fim:

Editor

Simulator

RunStepPrevResetDump

Machine Code	Basic Code	Original Code
0x00200513	addi x10 x0 2	addi a0, zero, 2 # variavel = 2
0x00400593	addi x11 x0 4	addi a1, zero, 4 # variavel = 4
0x00b50663	beq x10 x11 12	beq a0, a1, fim # condição de saída caso a0 for igual a a1
0x00a50533	add x10 x10 x10	add a0, a0, a0 # dobra o valor de a0
0xff9ff06f	jal x0 -8	j loop # faz o loop
0x00b50633	add x12 x11 x11	add a2, a1, a1 # soma a1 com ele mesmo e salva em a2
0x00000013	addi x0 x0 0	nop # saída

console output

RegistersMemory

zero

ra (x1)

sp (x2)

gp (x3)

tp (x4)

t0 (x5)

t1 (x6)

t2 (x7)

s0 (x8)

s1 (x9)

a0 (x10)

a1 (x11)

a2 (x12)

a3 (x13)

a4 (x14)

a5 (x15)

Display SettingsHex