

Exercício 04

Objetivo:

Consolidar o aprendizado da interface do RARS e aprender como executam as instruções de desvio do RISC-V executando o quarto exemplo de programação na linguagem de montagem do RISC-V no livro texto.

Instruções:

1. Inicie o MARS.
2. No editor de texto do RARS, transcreva o código abaixo e salve o arquivo com o nome **exercicio_04**.

```
#####  
# Exercício 04 - Patterson pags. 65/66  
# Mostra a compilação de um comando de desvio condicional (if-then-else) com else  
#####  
# Trecho em C:  
# if (i==j)  
#     f = g + h;  
# else  
#     f = g - h;  
  
    .text    # segmento de código (programa)  
main:  
    addi    s1, zero, 10    # inicializa s1 em 10  
    addi    s2, zero, 20    # inicializa s2 em 20  
    addi    s3, zero, 1     # inicializa s3 em 1  
    addi    s4, zero, 2     # inicializa s4 em 2  
    bne     s3, s4, Else    # se i!=j goto Else  
    add     s0, s1, s2      # f = g + h  
    j       Exit           # goto Exit  
Else: sub   s0, s1, s2      # f = g - h  
Exit: nop
```

3. Para iniciar a montagem do código vá ao menu **Run** e selecione a opção **Assemble** ou pressione **F3**.
4. Observe que fazendo uso da instrução **addi** é possível atribuir valores aos registradores s1, s2, s3 e s4, conforme segue:

Register name	Address		Value
s1	17	R17	10 ou 0xA
s2	18	R18	20 ou 0x14
s3	19	R19	1
s4	20	R20	2

Obs:

- O endereço 1 não se refere ao registrador s1, mas sim ao registrador at.
 - O endereço do registrador s1 é 17 (0x11). Ele pode ser referenciado digitando-se r17 ou R17, bem como s1 ou S1 (insensível à caixa)
5. Faça a execução passo-a-passo do programa e, a cada instrução, preencha a tabela abaixo cada vez que o valor de um registrador ou posição da memória de dados for modificado.

Antes da execução da instrução		Depois da execução da instrução				
PC	Instrução	R8	R9	R18	R19	R20
		(s0)	(s1)	(s2)	(s3)	(s4)
00400000	addi s1, zero, 10		0000000A			
00400004	addi s2, zero, 20			00000014		
00400008	addi s3, zero, 1				00000001	
0040000c	addi s4, zero, 2					00000002
00400010	bne s3, s4, Else					
00400014	add s0, s1, s2					
00400018	j Exit					
0040001c	sub s0, s1, s2 # f = g - h	ffffff6				
00400020	nop					

6. Altere as instruções addi para atribuir os seguintes valores aos registradores s0, s1, s2, s3 e s4 (note que agora s3 e s4 são iguais a 1):

Register name	Address		Value
s1	17	R17	10 ou 0xA
s2	18	R18	20 ou 0x14
s3	19	R19	1
s4	20	R20	1

7. Recarregue o programa com a opção: Run > Reset
8. Faça a execução passo-a-passo do programa e, a cada instrução, preencha a tabela abaixo cada vez que o valor de um registrador ou posição da memória de dados for modificado.

Antes da execução da instrução		Depois da execução da instrução				
PC	Instrução	R16	R17	R18	R19	R20
		(s0)	(s1)	(s2)	(s3)	(s4)
00400000	addi s1, zero, 10		0000000A			
00400004	addi s2, zero, 20			00000014		
00400008	addi s3, zero, 1				00000001	
0040000c	addi s4, zero, 2					00000001
00400010	bne s3, s4, Else					
00400014	add s0, s1, s2	0000001e				
00400018	j Exit					
0040001c	sub s0, s1, s2 # f = g - h					
00400020	nop					

9. Compare as duas tabelas e analise a diferença entre o fluxo de instruções executadas (veja a seqüência de valores do PC) e o valor final s0.