

## Exercício 07b

### Objetivo:

Consolidar o aprendizado da interface do RARS e aprender como executam as instruções de suporte a procedimentos do RISC-V.

### Instruções:

1. Inicie o RARS.
2. No editor de texto do RARS, transcreva o código abaixo e salve o arquivo com o nome **exercicio\_07b**.

```
#####
# Exercício 07b - Patterson pags. 72/73
# Mostra a compilação de um procedimento que não chama outro
#####
# Trecho em C:
# int leaf_example (int g, int h, int i, int j) {
#     int f;
#     f = (g + h) - (i + j);
#     return f;

        .text    # segmento de código (programa)
        jal, zero, main

leaf_example:
    addi sp, sp, -12    # ajusta a pilha para 3 valores
    sw   t1, 8(sp)      # salva o conteúdo de t1
    sw   t0, 4(sp)      # salva o conteúdo de t0
    sw   s0, 0(sp)      # salva o conteúdo de s0

    add t0, a0, a1      # t0 = g + h
    add t1, a2, a3      # t1 = i + j
    sub s0, t0, t1      # f = t0 - t1
    add a4, s0, zero    # return f

    lw   s0, 0(sp)      # restaura s0 da pilha
    lw   t0, 4(sp)      # restaura t0 da pilha
    lw   t1, 8(sp)      # restaura t1 da pilha
    addi sp, sp, 12     # ajusta a pilha
    jr   ra              # retorna do procedimento

main:
    addi t1, zero, 1    # inicia reg. com valor diferente de 0
    addi t0, zero, 2    # inicia reg. com valor diferente de 0
    addi s0, zero, 3    # inicia reg. com valor diferente de 0

    addi a0, zero, 4    # inicializa 1º parâmetro (g)
    addi a1, zero, 3    # inicializa 2º parâmetro (h)
    addi a2, zero, 2    # inicializa 3º parâmetro (i)
    addi a3, zero, 1    # inicializa 4º parâmetro (j)

    jal zero, leaf_example # chama o procedimento

    nop                  # não faz nada. v0 tem o resultado do procedimento
```

3. Para iniciar a montagem do código vá ao menu **Run** e selecione a opção **Assemble** ou pressione **F3**.
4. Faça a execução passo-a-passo do programa e, a cada instrução, preencha a tabela da página a seguir cada vez que o valor de um registrador ou posição da memória de dados for modificado. Note que o registrador **sp** aponta para um endereço do

segmento de dados identificado como current sp na janela Data Segment. Observe cuidadosamente a variação do valor sp e a atualização dos endereços dessa região do segmento de dados. Ao final da execução preencha a tabela final (na última página) que representa o conteúdo do segmento de dados de pilha usado pelo procedimento.

Antes da execução da instrução		Depois da execução da instrução											
PC	Instrução	f		g	h	i	j			Segmento de Dados (Pilha – current \$sp)			
		R2	R3	R4	R5	R6	R7	R29	R31				
		(\$v0)	(\$v1)	(\$a0)	(\$a1)	(\$a2)	(\$a3)	(\$sp)	(\$ra)	7FFFEFF0	7FFFEFF4	7FFFEFF8	7FFFEFFC
		00000000	00000000	00000000	00000000	00000000	00000000	7FFFEFFC	00000000	00000000	00000000	00000000	00000000

OBS: acrescente as linhas adicionais que forem necessárias.

Endereço da linha	Deslocamento			
	+0	+4	+8	+C
0x7FFFEFF0				

Posições inferiores ao topo original da pilha a serem ocupadas pelos valores inseridos na pilha.

Posição apontada originalmente pelo \$sp (0x7FFFEFFC)

