

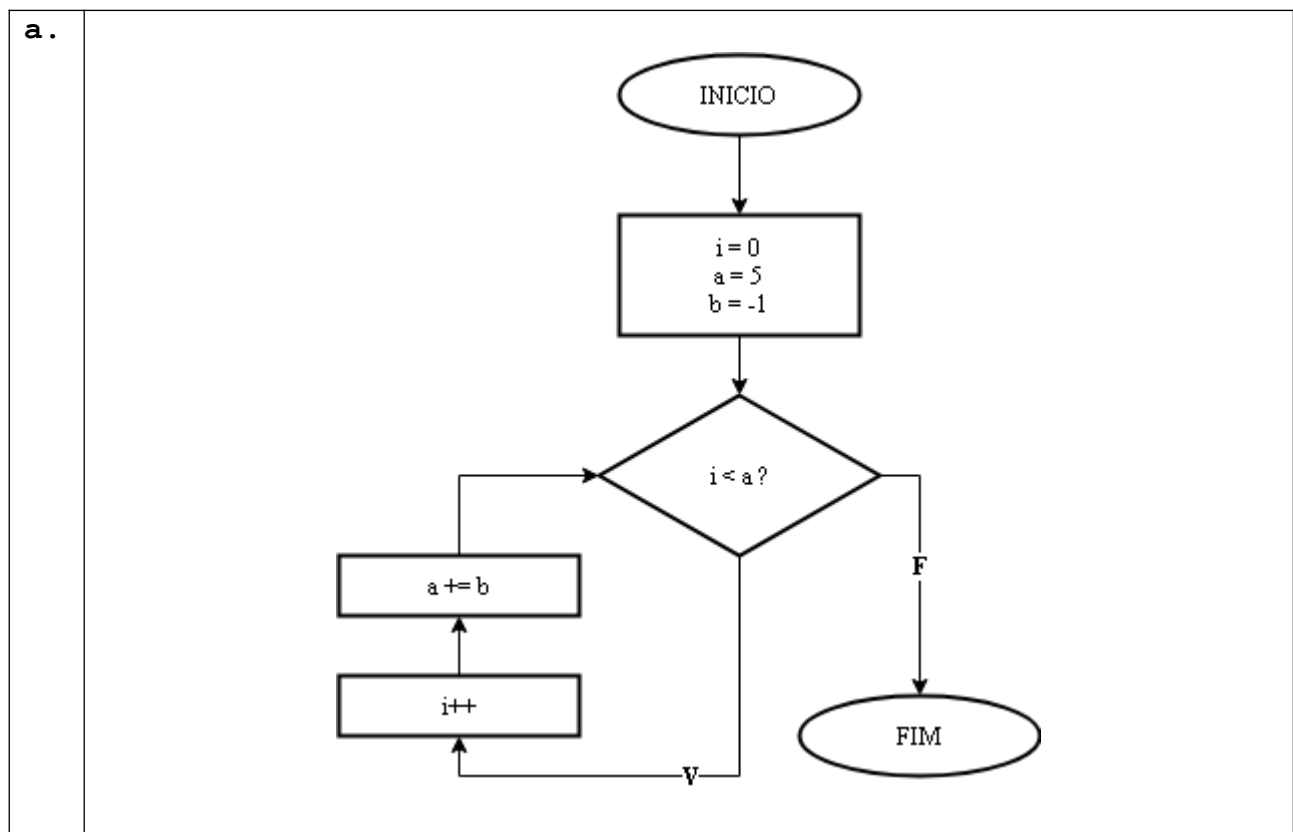
2 - Instruções: A linguagem de Máquina

Exercício 2.18

Para este problema, a tabela mantém algum código C. Você deverá avaliar essas instruções de código C no código assembly do MIPS.

| | |
|----|--|
| a. | <pre>for(i=0; i<a; i++) a += b;</pre> |
|----|--|

2.18.1 [5] <2.7> Para a tabela anterior, desenhe um gráfico de fluxo de controle do código C. Sugestão: utilize a plataforma gratuita draw.io para elaborar o fluxograma do laço de repetição FOR.



2.18.2 [5] <2.7> Para a tabela anterior, traduza o código C para o código assembly do MIPS. **Use um número mínimo de instruções.** Suponha que os valores de a, b e i estejam nos registradores \$s0, \$s1 e \$t0, respectivamente. Além disso, suponha que o registrador \$s2 mantenha o endereço de base do array D.

| | |
|-----------|--|
| a. | <pre> .data a: .word 5 b: .word -1 i: .word 0 .text .globl main main: lw \$s0, a lw \$s1, b lw \$s3, i Loop: add \$t0, \$zero, \$s3 slt \$t2, \$t0, \$s0 beq \$t2, \$zero, Exit add \$s0, \$s0, \$s1 addi \$s3, \$t0, 1 j Loop Exit: </pre> |
|-----------|--|

Para estes problemas, a tabela mantém fragmentos de código assembly do MIPS. Você deverá avaliar cada um dos fragmentos de código, familiarizando-se com as diferentes instruções de desvio do MIPS.

| | |
|-----------|---|
| a. | <pre> addi \$t1, \$0, 50 LOOP: lw \$s1, 0(\$s0) add \$s2, \$s2, \$s1 lw \$s1, 4(\$s0) add \$s2, \$s2, \$s1 addi \$s0, \$s0, 8 subi \$t1, \$t1, 1 bne \$t1, \$0, LOOP </pre> |
| b. | <pre> addi \$t1, \$0, \$0 LOOP: lw \$s1, 0(\$s0) add \$s2, \$s2, \$s1 addi \$s0, \$s0, 4 addi \$t1, \$t1, 1 slti \$t2, \$t1, 100 bne \$t2, \$0, LOOP </pre> |

Dica: os itens acima se referem ao laço de repetição FOR.

2.18.5 [5] <2.7> Traduza esses loops para C. Suponha que o inteiro *i* em nível de C seja mantido no registrador *\$t1*, *\$s2* mantenha o inteiro em nível de C chamado *result*, e *\$s0* mantenha o endereço de base do inteiro *MemArray*.

| | |
|-----------|---|
| a. | <pre>for(i=50; i > 0; i--){ result += MemArray[s0]; result += MemArray[s0+1]; s0 += 2; }</pre> |
| b. | <pre>for(i = 0; i < 100; i++){ result += MemArray[s0]; s0 += 1; }</pre> |