

# Arquitectura de Computadores I

## Exercícios

Licenciatura em Engenharia Informática

### Aula Prática #5 (Branches e Jumps)

1. Assumindo que os valores das variáveis  $x$  e  $y$  estão nos registos  $t0$  e  $t1$ , respectivamente, implemente em assembly RISC-V cada um dos troços de código seguintes:

(a) 

```
/* ----- */  
if (x > y)  
    x--;
```

(b) 

```
/* ----- */  
if (x < 0)  
    y = -x;  
else  
    y = x;
```

(c) 

```
/* ----- */  
y = 0;  
for (x = 1; x ≤ 10; x++)  
    y += x;
```

(d) 

```
/* ----- */  
switch (x) {  
    case 1:  
        y = 10;  
        break;  
    case 2:  
        y = 15;  
        break;  
    default:  
        y = 0;  
}
```

2. Analise o seguinte código assembly e escreva em linguagem C um pedaço de código equivalente. Considere que o registo  $t0$  corresponde à variável  $x$ .

```
A:    blt t0, zero, B  
      addi t0, t0, -1  
      jal zero, A  
B:
```

3. Analise o seguinte troço de código e determine o valor do registo  $a0$  no ponto marcado com a label AQUI.

```
addi a0, zero, 3  
jal ra, A  
addi a0, a0, 1  
jal ra, A  
addi a0, a0, -1  
AQUI: # qual o valor de a0?
```

```
A:      add a0, a0, a0  
        jalr zero, 0(ra)
```

4. A função de Collatz tem a seguinte definição:

$$\text{Collatz}(x) = \begin{cases} x/2 & \text{se } x \text{ é par,} \\ 3x + 1 & \text{se } x \text{ é ímpar.} \end{cases}$$

Implemente esta função em que o argumento  $x$  está no registo **a0** e o resultado da função é também colocado em **a0**. Supondo que a função foi chamada com a instrução **jal ra, Collatz**, escreva a função de modo a esta retornar ao ponto de onde foi chamada (instrução seguinte ao **jal**).