
ARQUITETURA DE COMPUTADORES

Prática 2

Resolva os seguintes exercícios, escrevendo a solução usando RISC-V *Assembly Language*.

Exercício 1

a) Considere o seguinte cálculo

$$3 + 7 - 4$$

Escreva o programa que faz este cálculo. Utilize os registos x10, x11 e x12 para armazenar o valor das 3 constantes e coloque o resultado em x10.

b) Altere o programa para fazer

$$8 * (3 + 7 - 4)$$

Utilizando uma instrução sll para efetuar a multiplicação por 8.

c) Considere a constante 824. Determine o valor do segundo símbolo hexadecimal usando as instruções and e srl.

d) Inverta o valor do 9º bit do número 824 utilizando a instrução xor.

Exercício 2

a) Considere o seguinte código em C

```
int a = 3, b = 7, c = 4, d;  
d = a + b - c + 2c;
```

Escreva o mesmo código em Assembly. Reserve espaço para as variáveis na memória utilizando a diretiva ".word". Utilize registos sempre que possível, já que um acesso a este é mais rápido do que um acesso à memória.

b) Escreva um programa que determine a média aritmética de quatro valores em memória nomeadamente:

```
vals:    .word 3, 7, 4, 6
```

Exercício 3

a) Escreva o programa que conta o número de bits com o valor '1' numa constante de quatro bits utilizando as instruções srl e and. Teste o programa utilizando a constante 0xD.

b) Considere a seguinte lista de números:

```
vals:    .word 3, 7, 4, 6
```

Construa uma segunda lista (vacc) cujo elemento i guarda o somatório dos elementos da lista vals até i . Para o exemplo fornecido, a segunda lista seria constituída com os números 3,10,14,20.

c) Escreva o programa que corresponde ao seguinte código em C.

```
int a=3, b=15, c=4, d=0;
d = 128*a + b/4 + 8*c
```

d) Escreva o programa em *assembly* que permite calcular o valor do polinómio $y = 3x^3 - 2x^2 + 5x + 7$ dado o valor de x inteiro. As variáveis x e y estão armazenadas em memória.

e) Considere o seguinte código em C:

```
char a = 3;          // 8 bits
short b = 4523;      // 16 bits
long c = 100000;     // 32 bits
```

Considere que estes operandos estão armazenados em memória utilizando as seguintes diretivas:

```
a:          .byte      3
b:          .half      4523
c:          .word      100000
```

Escreva o código que lê estes valores da memória e obtém a sua soma.