

20. Exercícios

DIM0321

2015.1

1. (*Potência*). A n -ésima potência de um número x , denotada por x^n , pode ser computada recursivamente observando a seguinte fórmula:

$$x^n = \begin{cases} 1, & \text{se } n = 0 \\ x.x^{n-1}, & \text{se } n > 1 \end{cases}$$

Considere neste exercício que x e n são números inteiros. Escreva uma função recursiva com a seguinte interface:

```
1| int potR(int x, int n);
```

2. (*Euclides*). Escrever uma função recursiva que receba dois números inteiros a e b positivos e devolva o máximo divisor comum entre eles usando o algoritmo de Euclides. Assumindo que a vai ser sempre maior que b .
3. (*Soma*). Escrever uma função recursiva com a seguinte interface:

```
1| int soma_d(int n);
```

que receba um número inteiro positivo n e devolva a soma de seus dígitos.

4. (*Piso*). O **piso** de um número real é o arredondamento de sua parte inteira para baixo. Então, sendo x um número real seu piso i é um número inteiro tal que $i < x < i + 1$. Matematicamente o piso de x é escrito assim

$$\lfloor x \rfloor$$

Escrever uma função recursiva com a seguinte interface:

```
1| int piso_log2(int n);
```

n	15	16	31	32	63	64
$\lfloor \log_2 n \rfloor$	3	4	4	5	5	6

5. (*Potência (bis)*). Podemos calcular a potência x^n de uma maneira mais eficiente. Observe primeiro que se n é uma potência de 2 então x^n pode ser computada como uma sequência de quadrados.

Por exemplo: x^4 é o quadrado de x^2 e assim x^4 pode ser computado usando somente duas multiplicações ao invés de três. Esta técnica pode ser usada mesmo quando n não é uma potência de 2 usando a seguinte fórmula:

$$x^n = \begin{cases} 1, & \text{se } n = 0, \\ (x^{n/2})^2, & \text{se } n \text{ par}, \\ x.x^{n-1}, & \text{se } n \text{ mpar} \end{cases}$$

Escrever uma função com a interface:

```
1 | int potencia(int x, int n);
```

que receba dois números inteiros x e n e devolva x^n e seja **recursiva**.