

## Lista 2 - Algoritmos recursivos e ordenação

**Data de entrega: 22/01/2009**

Prof. Wanderley Guimarães

EXERCÍCIO 1. [PF] Escreva uma *função recursiva* que calcule a soma dos dígitos de um inteiro positivo  $n$ . Enuncie uma definição recursiva do problema e prove. A função precisa ter o seguinte protótipo:

```
int soma_digitos_r (int n).
```

EXERCÍCIO 2. [FOCS] Definimos  $n^2$  recursivamente da seguinte forma:

**Base:** Para  $n = 1$ ,  $1^2 = 1$ ;

**Indução:** se  $n^2 = m$ , então  $(n + 1)^2 = m + 2n + 1$ .

Escreva uma *função recursiva* que calcule  $n^2$ . Prove por indução em  $n$  a correção da definição. A função precisa ter o seguinte protótipo:

```
int nquad (int n).
```

EXERCÍCIO 3. Escreva uma *função recursiva* que calcule o índice do elemento mínimo do vetor  $v[0..(n - 1)]$  de inteiros. Use o seguinte fato: para determinar o índice do elemento mínimo de  $v[0..(n - 1)]$ , podemos determinar os índices dos elementos mínimos de  $v[0..\lfloor \frac{n}{2} \rfloor]$  e  $v[(\lfloor \frac{n}{2} \rfloor + 1)..(n - 1)]$ , e em seguida, determinar o índice do elemento mínimo de  $v[0..(n - 1)]$ . A função precisa ter o seguinte protótipo:

```
int minimo_r2 (int n, int v[]).
```

EXERCÍCIO 4. [PF] Escreva uma função recursiva eficiente que receba inteiros positivos  $k$  e  $n$  e calcule  $k^n$ . (Suponha que  $k^n$  cabe em um **int**.) A função precisa ter o seguinte protótipo:

```
int pot (int k, int n).
```

EXERCÍCIO 5. [PF] Escreva uma versão do algoritmo de inserção que tenha o seguinte invariante: no início de cada iteração, o vetor  $v[j + 1..n - 1]$  é crescente. A função precisa ter o seguinte protótipo:

```
void insercao2 (int n, int v[]).
```

EXERCÍCIO 6. [PF] Escreva uma função que coloque em ordem lexicográfica um vetor de  $n$  strings, onde  $1 \leq n \leq 50$  e cada string possui comprimento entre 1 e 50. Use o algoritmo de inserção. A função precisa ter o seguinte protótipo:

```
void ordena_strings (int n, char s[50][51]).
```