

ESTRUTURAS DE DADOS

Exercícios de Fixação: Recursividade

1. O Maior Divisor Comum de dois inteiros é o maior inteiro pelo qual os dois números podem ser divididos. O MDC é definido da seguinte forma:

Se $y = 0 \rightarrow \text{MDC}(x, y) = x$
 Caso contrário $\rightarrow \text{MDC}(x, y) = \text{MDC}(y, \text{resto}(x/y))$

Calcular o MDC de dois inteiros positivos de forma recursiva.

2. Escrever uma função recursiva que calcule x^y
3. Manuscritamente resolva os itens a) e b) por meio da função *Rec*:

```
int Rec(int n, int m)
{
    if (n==m || n==0)
        return 1;
    else return Rec(n-1,m) + Rec(n-1,m+1);
}
```

- a) Qual o valor de $\text{Rec}(5,3)$?
 - b) Quantas chamadas serão feitas na avaliação acima?
4. Elaborar um módulo que ao receber um vetor de números inteiros e seu tamanho lógico, exiba-o de modo que seus elementos apareçam de forma inversa, conforme o exemplo:

456	123	543	997	890	547	854	Lixo	Lixo	Lixo
1	2	3	4	5	6	TL=7	8	9	10

A saída produzida pelo programa seria a seguinte:

854 547 890 997 543 123 456

5. Defina uma função recursiva para determinar o maior divisor comum entre dois números naturais x e y , baseando-se nas regras abaixo. Em seguida, apresente uma versão iterativa do algoritmo capaz de realizar a mesma tarefa.

$\text{mdc}(x,y) = y$	se $x=y$
$\text{mdc}(x,y) = \text{mdc}(y,x)$	se $x < y$
$\text{mdc}(x,y) = \text{mdc}(x-y,y)$	se $x > y$

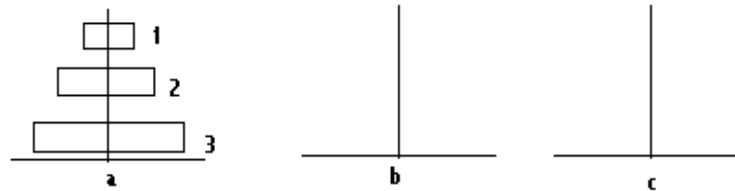
6. Escrever um procedimento recursivo que tome n palavras da entrada e as escreva na ordem reversa. Ex.: se a chamada `PalavrasOrdemReversa(char Frase[100])` fosse executada e as palavras digitadas fossem:

O elefante bebe Água

A saída produzida pelo programa seria a seguinte:

Água bebe elefante O

7. Elabore um algoritmo que mova três discos de uma Torre de Hanói, que consiste em três hastes (a – b – c), uma das quais serve de suporte para 3 discos de tamanhos diferentes (1 – 2 – 3), os menores sobre os maiores. Pode-se mover um disco de cada vez para qualquer haste contanto que nunca seja colocado um disco maior sobre o menor. O objetivo é transferir os 3 discos para outra haste.



8. A Série de Fibonacci (1, 1, 2, 3, 5, 8, 13, 21, 34, ...) pode ser definida da seguinte maneira:

$F(n) = n$	se $n=0$ ou $n=1$
$F(n) = F(n-2)+F(n-1)$	se $n \geq 2$

- a) É possível definir uma rotina recursiva **F(n)** tal que ela imprima os **n** primeiros termos da série de Fibonacci? Se for, apresente o algoritmo; senão, explique por que não é possível.
- b) Seja **F(n)** uma função que retorna o n-ésimo termo da série de Fibonacci. Defina para **F(n)** uma versão recursiva e outra iterativa e compare, sucessivamente, o tempo gasto de executar cada uma das versões com valores crescentes de **n** [Dica: escreva um programa para automatizar as comparações e gerar uma tabela contendo os valores de **n** e os tempos gastos por cada uma das versões].
9. Faça uma função recursiva que realize o cálculo do resto da divisão de dois números.
Dica: realizar sucessivas subtrações.
10. Construa uma função que calcule a multiplicação de dois números positivos, maiores que zero, através de sucessivas somas do primeiro número.
11. Cálculo recursivo da seguinte fórmula:

$$\sum_{i=1}^n (i * i)$$