

# MC102 - Algoritmos e Programação de Computadores

## Lista de Exercícios 8

19 de Junho de 2018

1. Escreva um algoritmo iterativo em Python para avaliar  $a * b$  usando a adição, onde  $a$  e  $b$  são inteiros não negativos.
2. Escreva uma definição recursiva para o cálculo de  $a * b$  e em seguida escreva um algoritmo recursivo para o cálculo de  $a * b$ , onde  $a$  e  $b$  são inteiros não-negativos.
3. Faça uma função recursiva em Python que calcula o elemento máximo de um vetor.
4. Faça uma função recursiva em Python que calcula a média dos elementos de um vetor.
5. Faça uma representação da memória do computador considerando as chamadas das funções recursivas abaixo (vistas em aula). Faça um modelo passo a passo como nos exemplos visto em sala de aula:
  - `fatorial(6)`
  - `fibonacci(5)`
6. Determine o que a seguinte definição recursiva para uma função  $f$  calcula. A definição da função  $f$  é dada abaixo:
  - Se  $n == 0$  retorne 0.
  - Se  $n > 0$  retorne  $n + f(n - 1)$ .
7. Execute a função `ff` abaixo com os argumentos 7 e 0.

```
def ff(n, ind):
    for i in range(0, ind):
        print(' ', end='')
    print('ff', n, ind)
    if n == 1:
        return 1;
    if n % 2==0:
        return ff(n//2, ind + 1);
    return ff((n-1)//2, ind +1) + ff((n+1)//2, ind +1)

print(ff(7, 0))
```

8. Escreva uma função recursiva que calcule  $\lfloor \lg n \rfloor$ , ou seja, o *piso* do logaritmo de  $n$  na base 2.
9. Escreva uma função recursiva para a busca sequencial em um vetor.

10. Escreva uma função recursiva para a busca binária em um vetor.
11. Escreva uma função recursiva para o cálculo do máximo divisor comum de dois números inteiros não negativos. Lembre-se que
- $\text{mdc}(x,y) = x$  se  $y$  é zero,
  - $\text{mdc}(x,y) = \text{mdc}(y,x\%y)$  caso contrário.
12. Suponha que uma matriz binária quadrada  $M$  represente a ligação entre um conjunto de  $n$  cidades. Desta forma  $M[i, j] = 1$  indica que existe uma estrada da cidade  $i$  para a cidade  $j$ , e  $M[i, j] = 0$  caso contrário. Por exemplo, na matriz abaixo temos que a cidade 0 possui estradas para 1 e 2, já a cidade 1 possui estrada apenas para 2. Note que existe estrada de 0 para 1, mas não de 1 para 0, pois só há estradas de 2 e 3 para 0.

```

0 1 1 0
0 0 1 0
1 1 0 1
1 0 1 0

```

Escreva uma função recursiva que dada uma matriz  $M$  e uma cidade  $i$  determina todas as cidades que podem ser alcançadas a partir de  $i$ .

13. Faça uma função recursiva para calcular  $\binom{n}{k}$  sabendo que

$$\binom{n}{k} = \binom{n-1}{k} + \binom{n-1}{k-1},$$

e

$$\binom{n}{n} = 1 \text{ e } \binom{n}{1} = n.$$

14. Considere a seguinte variação do problema das torres de Hanoi. O objetivo continua sendo em levar os  $n$  discos de A para C com as restrições (1) Somente um disco e que está no topo pode ser movimentado por vez, e (2) um disco só pode ser posicionado sobre um disco de maior tamanho, mas agora adicionamos a restrição (3) de que não é possível mover um disco diretamente de A para C (ou C para A), ou seja todos os movimentos requerem usar a estaca intermediária B. Escreva um algoritmo que gera a solução para este problema. O protótipo da função é

```

#chamada inicial -> hanoi(n, 'A', 'C', 'B');
def hanoi(n, ini, fim, inter);

```

