

 <p>INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PARAÍBA</p>	<p>Engenharia de Computação Disciplina: Algoritmos e Computação Semestre Letivo: 2016 Professor: Marcelo Siqueira / Henrique Cunha</p>
<p>Assunto:</p>	<p>Recursividade</p>
<p>Objetivos:</p>	<ol style="list-style-type: none"> 1. Analisar a sintaxe de códigos escritos em Python 2. Entender o funcionamento de algoritmos recursivo 3. Desenvolver programas recursivos em Python

ROTEIRO DE AULA 15 – 08/08/2016

- 1 Mostre, através de teste de mesa, o resultado das seguintes funções, considerando que para cada uma delas são realizadas as seguintes chamadas: $f(0)$, $f(1)$, $f(5)$.

```
a def f(n):
    if (n == 0):
        return 1
    return(n * f(n-1))

b def f(n):
    if (n == 0) or (n == 1):
        return 1
    return(f(n-1)+ 2 * f(n-2))

c def f(n):
    if (n == 0):
        print("Zero ")
    else:
        print(n)
        print(n)
        f(n-1)
```

- 2 Considere a função abaixo:

```
def x(a):
    if ( a <= 0 )
        return 0;
    else
        return a + x(a-1);
```

Explique o que essa função faz.

3. Escreva uma função não-recursiva que resolve o mesmo problema.
4. Escreva uma função que retorne a soma dos n primeiros termos da sequência de Fibonacci.
5. Faça uma função recursiva que permita inverter um número inteiro N.Ex: 123 será transformado em 321.
6. Faça um programa recursivo para calcular o produto dos elementos de um vetor de inteiros.
7. Faça um programa recursivo para verificar se uma palavra é um palíndromo.
8. O máximo divisor comum dos inteiros x e y é o maior inteiro que divide x e y. Escreva uma função recursiva mdc em C, que retorna o máximo divisor comum de x e y. O mdc de x e y é definido como segue: se y é igual a 0, então mdc(x,y) é x; caso contrário, mdc(x,y) é mdc (y, x%y), onde % é o operador resto da divisão.
9. Crie um programa que contenha uma função recursiva para encontrar o menor elemento em um vetor. A leitura dos elementos do vetor e impressão do menor elemento devem ser feitas no programa principal.
10. Escreva um programa que leia e armazene valores inteiros não repetidos em uma lista de 10 posições. Em seguida escreva um programa com repetição e um recursivo que identifique a posição da lista em que se encontra um número '0' (zero).
11. Escreva um programa que leia e armazene valores inteiros e positivos não repetidos em um lista de 10 posições. Em seguida o programa deve ler um número qualquer da entrada padrão e, usando uma função recursiva, identificar se ele está contido ou não no lista.
12. Escreva um programa que leia um conjunto de caracteres e armazene em um lista de 10 posições. O programa deve imprimir o seu conteúdo de forma inversa e recursiva.
13. Escreva um programa que leia e armazene valores inteiros em um lista de 10 posições. Em seguida, escreva uma função que some todos os elementos do lista ($0 < N < 10$).
14. Escreva um programa que leia e armazene valores inteiros em um lista de 10 posições. Em seguida, escreva uma função que some os N números da lista ($0 < N < 10$).
15. Escreva um programa que realize o somatório dos N termos da série abaixo:

$$S = 1/2 + 2/6 + 3/10 + 4/14 + 5/18 + 6/22....$$
16. Escreva um programa que realize o somatório dos N termos da série abaixo:

$$S = 1/3 + 1 + 2/4 + 1 + 3/5 + 1 + 4/6....$$
17. Escreva um programa que realize o produto de dois números n e m quaisquer utilizando somas sucessivas. Por exemplo, $2 * 3 = 2 + 2 * 2$; $4 * 5 = 4 + 4 * 4$