

Fundamentos de Estruturas de Dados

Lista de Exercícios 01 - Recursividade

OBS: Para essas questões, não devem ser utilizadas instruções de repetição for, while ou do-while.

Questão 01

Utilizando a recursividade, escreva métodos recursivos capazes de fazer as seguintes operações (utilize métodos auxiliares, se necessário):

1. Somar todos os números primos de um intervalo fechado de números;
2. Multiplicar todos os elementos pares para a enésima sequência de Fibonacci.

Exemplo: `public int multiFib(int n);`

Entrada	Saída
n = 1	0
n = 4	2
n = 6	16

Questão 02

Implemente um método recursivo que receba por parâmetro um número inteiro positivo N para linha e outro número M inteiro positivo para a coluna, retorne valor correspondente da posição N, M do triângulo de pascal.

Ex: `public int pascal (4, 2) = 6`

Linha 0	1																			
Linha 1	1	1																		
Linha 2	1	2	1																	
Linha 3	1	3	3	1																
Linha 4	1	4	6	4	1															
Linha 5	1	5	10	10	5	1														
Linha 6	1	6	15	20	15	6	1													
Linha 7	1	7	21	35	35	21	7	1												
Linha 8	1	8	28	56	70	56	28	8	1											
Linha 9	1	9	36	84	126	126	84	36	9	1										
Linha 10	1	10	45	120	210	252	210	120	45	10	1									
...

Questão 03

Implemente um método recursivo que retorna a potência de um número.

Exemplo:

Entrada: calcularPotencia(base: 2, expoente: 3)

Saída: $2 \times 2 \times 2 = 8$

Questão 04

Dado um número natural n na base decimal, escreva um método recursivo que retorna seu valor na base binária.

Exemplo:

Entrada: 4

Saída: 100

Questão 05

Dado um array de inteiros, crie um método recursivo para que seja retornada a soma de todos os elementos do array.

Exemplo:

Entrada: [2, 5, -20, 7, 100, 1, -14, 78, 30, 0]

Saída: 189

Questão 06

Escreva um método recursivo que retorna n termos (quantidade informada pelo usuário) de uma P.A, tendo informado, também, seu elemento inicial e sua razão.

Ex: public String termosPA(int termos, double atual, double razao);

Exemplo:

Entrada: termos = 5, atual = 0.80, razao = 0.58

Saída: >> 0.8 >> 1.38 >> 1.96 >> 2.54 >> 3.12

Questão 07

Dado um array de inteiros e o seu número de elementos, recursivamente, inverta a posição dos seus elementos.

Exemplo:

Entrada: 1 - 2 - 3 - 4 - 5

Saída: 5 - 4 - 3 - 2 - 1

Questão 08 (ADAPTADA - CEPERJ 2012)

```
variáveis  
inteiro: YZ, NR, PC;  
BRASIL(XW: inteiro) : inteiro{  
  adicionar 1 a PC;  
  se XW < 2 retorna 4;  
  senão retorna PC + BRASIL(XW - 1);  
}
```

```
início  
atribuir 3 a NR;  
atribuir 0 a PC;  
atribuir BRASIL(NR) a YZ  
mostrar(YZ, PC);  
fim_programa;
```

Observe o pseudocódigo acima, referente a um programa de computador que ilustra o uso de uma função recursiva. Após a execução, a quantidade de vezes que a função foi executada e o valor final para YZ serão, respectivamente:

- a) 4 e 6
- b) 4 e 7
- c) 3 e 5
- d) 3 e 6
- e) 3 e 7

Questão 09

Implemente uma função utilizando recursividade que recebe um número inteiro N e um dígito inteiro K e retorna quantas vezes o dígito K aparece no número N.

Exemplo: O dígito 8 no número 62489538718 aparece 3 vezes.

Dica: Você pode criar uma função auxiliar que retorna o valor 0 e 1 para uma entrada booleana.

Questão 10

Escreva um método recursivo que informe se um número inteiro passado como argumento é primo ou não. Dica: use o operador '%'.

Exemplo:

ENTRADA SAÍDA

1 false

2 true

29 true

Questão 11

Considerando um número arbitrário máximo 1000, crie um método recursivo que imprime na tela uma sequência de números, primeiro em ordem crescente e depois decrescente, inicialmente de um número passado como argumento, e então multiplicando-o por 2 sem passar do número máximo.

Exemplo:

```
public void seq(int n);
```

```
seq(100) = 100, 200, 400, 800, 800, 400, 200, 100
```

Questão 12

Crie um método recursivo que verifica se todas as letras presentes em uma primeira String também estão contidas na segunda.

Exemplos:

Entrada: "oi" e "abacaxi" → Saída: false

Entrada: "lua" e "australia" → Saída: true

Questão 13

O hiperfatorial de um número N, escrito $H(n)$, é definido por:

$$H(n) = \prod_{k=1}^n k^k = 1^1 \cdot 2^2 \cdot 3^3 \cdots (n-1)^{n-1} \cdot n^n$$

Sendo assim, faça uma função recursiva que receba um número inteiro positivo N e retorne o hiperfatorial desse número.

Questão 14

Considere um sistema numérico que não tenha a operação de adição implementada e que você disponha somente dos operadores (funções) sucessor e predecessor. Então, pede-se para escrever uma função recursiva que calcule a soma de dois números x e y através desses dois operadores: sucessor e predecessor.

Questão 15

Escreva uma função recursiva que calcule a soma dos dígitos de um número inteiro.

Exemplo: Se a entrada for 123, a saída deverá ser $1+2+3 = 6$.