

Lista 2

Esta lista contém exercícios típicos para a prova. Na verdade é uma coleção de questões que eu já pedi em alguma prova de MC102. As questões dessa lista envolvem principalmente arrays, vetores e funções. As questões não estão por ordem de dificuldade.

Exercício 1. O ponto de corte de um vetor é um elemento do vetor tal que todos os elementos do vetor a esquerda (com índice menor) são menores que o elemento de corte, e todos os elementos a direita são maiores que o elemento de corte. Também vamos assumir que o elemento de corte não pode ser nem o primeiro nem o ultimo elemento do vetor. Alguns vetores não tem elemento de corte e alguns tem mais que um elemento de corte. Por exemplo o vetor 2 4 3 5 7 12 10 tem pontos de corte nas posições 4 e 5.

Assuma que a função `MAXO` esta definida como

```
int maxo(int v[], int i, int j );
```

e dado um vetor `v` a função `maxo` retorna o índice do maior elemento de `v` entre `i` e `j`. Por exemplo se o vetor for 1 3 4 2 7 4 7 4 3 e `i= 3` e `j = 6` a função vai retornar 5 pois o elemento na posição 5 é o maior entre os elementos da posição 3 e da posição 6.

Assuma também que a função `mino` esta definida como `maxo` mas retorna o menor elemento do vetor no intervalo entre `i` e `j`

Escreva a função `corte` com a definição (protótipo) abaixo que calcula e retorna o índice de algum ponto de corte do vetor `x` de tamanho `n`, ou 0 se o vetor `x` não tem um ponto de corte.

```
int corte(int x[], int n);
```

Exercício 2. Dado que a função `primo` esta definida como

```
int primo (int n );
```

e que retorna 1 quando o argumento é primo. Escreva a função `xxx` com a especificação

```
int xxx(int m[50][],int n);
```

que retorna 1 quando o argumento (uma matriz `n` por `n`) contem apenas números primos na diagonal principal, e 0 se não.

Exercício 3. Escreva um programa que lê uma sequência de caracteres, terminada por `*` (todos na mesma linha). Essa sequência será uma sequência `boa` se só contiver as letras de A até Z, o branco e o ponto, excetuando-se o `*`

que marca o final (pode haver outros caracteres após o *, mas eles não serão considerados). O programa deverá dizer se a sequência é boa ou não, e se for boa, o número de letras (de A a Z) na sequência.

Exercício 4. Escreva um programa que lerá $n > 2$ e depois n inteiros, e imprime a média dos **últimos 3** números lidos. Você não deve assumir nada a respeito de n - se você assumir, por exemplo, que $n < 10000$ seu programa só valerá no máximo 1.5 pontos.

Exercício 5. Escreva um programa que a partir de 99 até o 3, imprima todos os inteiros divisíveis por 3 na ordem decrescente. Após imprimir cada inteiro o programa pergunta se o usuário quer continuar e lê um caractere. Se esse caractere for **s** o programa imprime o próximo inteiro. Se não for o programa para. Após imprimir o 3 e se o usuário entrar com **s** o programa deverá voltar a imprimir o 99 e recomeçar o processo.

Exercício 6. Numa matriz quadrada $n \times n$ nós vamos definir um **pico** como sendo um elemento i, j da matriz tal que $1 < i, j < n$, cujo valor é maior que de seus oito vizinhos. O **desnível de um pico** é a diferença do valor do pico e seu vizinho de maior valor. O **desnível de uma matriz** é o maior desnível entre seus picos.

Por exemplo

1	1	1	1	1
1	4	5	3	2
2	3	2	3	4
2	5	2	9	2
1	3	3	4	1

os picos estão nas posições (1,3) (4,2) e (4,4). Os respectivos desníveis são 1, 3, 5. O desnível da matriz é 5.

Dado

```
typedef int mat[50][50];
```

escreva a função **desnivel** que dado uma matriz e seu tamanho retorna o desnível da matriz.

```
int desnivel(mat x, int n);
```

Exercício 7. Escreva um programa que lê um inteiro positivo maior que 3 e imprime um quadrado de números em sequência (começando com 1) de lado

igual ao numero lido. Por exemplo, se o numero lido for 5, o programa deve imprimir o “quadrado” de lado 5 abaixo:

```
1  2  3  4  5
6  7  8  9 10
11 12 13 14 15
16 17 18 19 20
21 22 23 24 25
```

Exercício 8. Assuma a seguinte declaração de tipo:

```
typedef int arr[50][50];
```

Escreva uma função que recebe como argumentos uma matriz do tipo `arr` e retorna `1` se a matriz for simétrica (a matriz é igual a sua transposta), ou seja se a linha 1 da matriz for igual a coluna 1, e se a linha 2 for igual a coluna 2, e assim por diante. Se a matriz não for simétrica, a função deve retornar `0`.

Exercício 9. Quando eu estava no primário, eu usava um esquema de código secreto para me comunicar com meus amigos. O código se chamava

ZENIT
POLAR

A idéia do código era trocar cada letra da mensagem original pela letra correspondente, acima ou abaixo na tabela acima. Ou seja, trocava-se-se cada `z` por `p`, cada `a` por `i`, e assim por diante. O `h`, por exemplo, não era trocado por nenhuma letra.

Escreva uma função que dado um string, retorna aquele string codificado pelo código Zenit Polar.