

Ponteiros II

O código abaixo está certo?

```
void troca (int *i, int *j) {  
    int *temp;  
    *temp = *i;  
    *i = *j;  
    *j = *temp;  
}
```

Faça o teste de mesa para o código abaixo

```
int teste() {  
    int t, j, *p, *g;  
    t = 5;  
    p = &t;  
    j = 3;  
    g = &j;  
    *p = (*g + t) * 2;  
    *g = 4;  
    g = p;  
    j = p;  
    *p = j;  
    return (*p);  
}
```

Identifique o que está errado com código

Faça o teste de mesa para o código abaixo

```
int teste() {  
    int t, j, *p, *g;  
    t = 5;  
    p = &t;  
    j = 3;  
    g = &j;  
    *p = (t**g) ;  
    return (*p);  
}
```

```
typedef struct {  
    int base;  
    int altura;  
} area;
```

Implemente a seguinte função:

```
void teste (area v[100], area *Maiorbase, area * menoraltura );
```

Esta função recebe um vetor de estruturas quad_t, identifica a maior base e salva no argumento Maiorbase. Da mesma forma, a função identifica a menor altura e salva na estrutura MenorAltura.

Alocação dinâmica

```
Int main(){  
    int *p, *q;  
    p = malloc (sizeof (int));  
    *p = 123;  
    q = malloc (sizeof (int));  
    *q = *p;  
    q = p;  
    free (p);  
    free (q); // má ideia...  
}
```

Alocação dinâmica

```
Int main(){
    int *v;
    int n, i;
    scanf ("%d", &n);
    v = (int *)malloc (n * sizeof (int));
    for (i = 0; i < n; ++i)
        scanf ("%d", &v[i]);
    for (i = n; i > 0; --i)
        printf ("%d ", v[i-1]);
    free (v);
}
```

Revisão de Matrizes

```
#include <stdio.h>
#include <stdlib.h>
void main()
{
    float matriz [2][2];
    float media, quant;
    matriz[0][0] = 9.0;
    matriz[0][1] = 8.0;
    matriz[1][0] = 7.0;
    matriz[1][1] = 7.0;
    quant = matriz[0][0] + matriz[0][1]
           + matriz[1][0] + matriz[1][1];
    media = quant / 4.0;
    printf (" A média dos 4 números é: %f.", media);
}
```



<i>Índices</i>	<i>0</i>	<i>1</i>
<i>0</i>	9.0	8.0
<i>1</i>	7.0	7.0

Matrizes e Funções

- Passagem de matrizes por parâmetro:

```
#define NUM_ALUNOS 3
```

```
float media_geral ( float notas[][2], int n )
```

```
{.....
```

```
}
```

```
int main()
```

```
{
```

```
float notas [NUM_ALUNOS][2], media_turma;
```

```
....
```

```
media_turma = media_geral ( notas, NUM_ALUNOS );
```

Matrizes e Funções

```
#include <stdio.h>
#include <stdlib.h>
#define NUM_ALUNOS 3
int main()
{
float notas [NUM_ALUNOS][2], media_turma;
int i;
    for ( i = 0; i < NUM_ALUNOS; i++ )
    {
        printf ("Digite a 1a. nota do %d o. aluno \n: ", i+1);
        scanf ("%f", &notas[i][0]);
        printf ("Digite a 2a. nota do %d o. aluno \n: ", i+1);
        scanf ("%f", &notas[i][1]);
    }
media_turma = media_geral ( notas, NUM_ALUNOS );
printf ( "A media da turma eh %.2f \n", media_turma);
system ("Pause");
return 0;
}
```

Matrizes e Funções

```
float media_geral ( float notas[][2], int n )
```

```
{  
int i = 0;  
float m = 0.0;  
float notaaluno = 0.0;  
for ( i = 0; i < n; i++ )  
{  
    notaaluno = ( notas[i][0] + notas[i][1] )/2;  
    m = m + notaaluno;  
}  
m = m / n;  
return m;  
}
```

Atividade

-
- Faça uma função que receba uma matriz de inteiros de 20x20. A função retorna um inteiro que corresponde à quantidade de diagonais com números **crescentes** . Uma diagonal é formada no sentido “esquerda-direita, de cima para baixo”, e é composta por, pelo menos, 3 números.

30	1	7	77	8
40	24	3	5	98
10	39	22	9	2
1	12	13	12	10
3	2	25	99	7