



LISTA DE EXERCÍCIOS

1. Analise o código seguinte e determine o que aparecerá na tela se o mesmo for executado.

```
#include <stdio.h>

void opera(float *a, float *b)
{
    printf("Recebido: %.2f %.2f\n", *a, *b);
    *a=*a*b*3-1;
    printf("a alterado em opera: %.2f\n", *a);
}

void proc(float *n, int r) {
    int k;
    float w=0, z=1.5;
    for (k = 0; k < r; k++)    {
        *n+=4.5;
        w++;
        printf("%d %.2f %.2f\n", k, *n, w);
    }
    opera(&w, &z);
    *n = *n + w;
}

int main(void)
{
    int rep=2;
    float num=2.5;
    float x=8.0, y=2.5;
    unsigned char n1=13, n2=20;
    opera(&x, &y);
    printf("Parte 1: %.2f\n", x);
    proc(&num, rep);
    printf("Parte 2: %.2f\n", num);
    printf("Op 1: %d\n", n1&n2);
    printf("Op 2: %d\n", n1<<3);
    printf("Op 3: %d\n", n2|n1);
    printf("Op 4: %d\n", n2>>2);

    return 0;
}
```

2. Analise o programa seguinte e escreva o que aparecerá na tela se o mesmo for executado.

```
#include <stdio.h>

void chg(int *p, int v)
{
    *p = *p + v;
}

int main(void) {
    int y=1, *py, x[]={88, 77, 66}, *px;
    py = &y;
    px = &x[0];
    printf ("A. val. apontado por px = %d\n", *px);
    px++;
    printf ("B. val. apontado por px = %d\n", *px);
    chg(py, -3);
    printf ("C. y = %d\n", y);
    y--;
    printf("D. val. apontado por py = %d\n", *py);
    *px = *py - 10;
    printf("E. val. apontado por px = %d\n", *px);
    printf("F. x[0]: %d, x[2]: %d\n", x[0], x[2]);
    printf("G. Op: = %d\n", (x[0]-80) | (x[2]-60));
    return(0);
}
```

3. Faça um programa para ler um arquivo csv escrito no formato:

"hora,minuto,temperatura em Celsius,pressão atmosférica em mmHg,local"

e armazenar os dados em uma estrutura *Meteoro* (vetor com no máximo 100 elementos) com os seguintes campos:

```
int hora;
int minuto;
float tcelsius;
float pammhg;
char local[30];
```

Ignore a primeira linha do arquivo. Depois, escreva uma função para mostrar em que local, hora e minuto ocorreu a menor pressão atmosférica. A função deve ter o seguinte protótipo (assinatura):

```
void mostra_min(struct Meteoro met[], int numdados);
```

A função principal (*main*) deve solicitar o nome do arquivo de entrada.

Exemplo de funcionamento:

Digite o arquivo de entrada: *entrada.txt*
Foram lidos 15 dados.
A pressão mínima de 918 mmHg ocorreu às 18h e 13 minutos, em EPAGRI-ITAJAI.

4. Faça um programa que crie as seguintes estruturas

Ponto2D com os seguintes campos:

```
float x;  
float y;
```

CorRGB com os seguintes campos:

```
int R;  
int G;  
int B;
```

Quadrilatero com os seguintes campos:

```
struct Ponto2D vA;           //vertice 1  
struct Ponto2D vB;           //vertice 2  
struct Ponto2D vC;           //vertice 3  
struct Ponto2D vD;           //vertice 4  
struct CorRGB cor;           //cor
```

Depois, escreva uma função para calcular a área de um quadrilátero armazenado nessa estrutura, com o seguinte protótipo (assinatura da função):

```
float calc_area(struct Quadrilatero *qua);
```

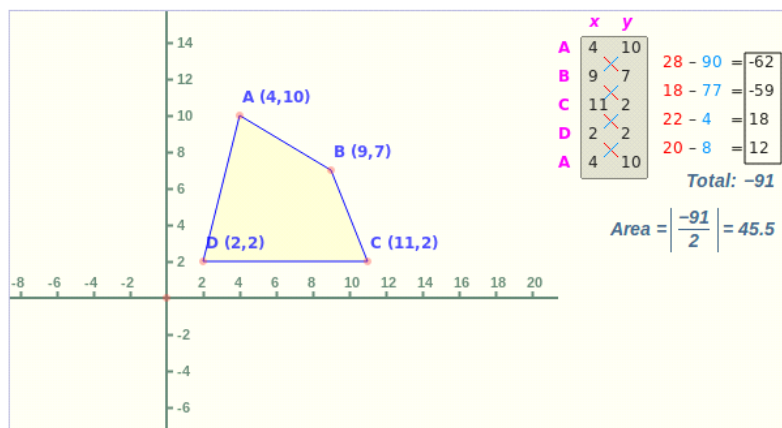
A área é calculada com

$$A_{\text{quad.}} = \frac{1}{2} |x_1 y_2 + x_2 y_3 + x_3 y_4 + x_4 y_1 - x_2 y_1 - x_3 y_2 - x_4 y_3 - x_1 y_4|$$

onde as barras indicam a operação de valor absoluto.

Exemplo de funcionamento:

Digite os vértices de um quadrilátero (coordenadas x,y): 4 10 9 7 11 2 2 2
Digite a cor (três valores RGB): 100 35 255
A área desse quadrilátero é: 45.5



Fonte: <https://www.mathopenref.com/coordpolygonarea.html>