

```
#include <stdio.h>
main()
{
    unsigned char i=0;
    printf("_____");
    while (i<128)
    {
        printf("\n|%5d | %4c |",i,i);
        printf("\n_____");
        i++;
    }
}
```

```
#include <stdio.h>
main()
{
    unsigned char i;
    printf("_____");
    for (i=0;i-128;i++)
    {
        printf("\n|%5d | %4c |",i,i);
        printf("\n_____");
    }
}
```

Aula Prática X

Aula Prática X – Teoria

Estruturas de Controle de Fluxo

Laços de repetição (continuação)

```
do
{
    <instrução1>
    .
    .
    .
    <instruçãon>
}
while (<condição>);
```

Estruturas de Controle de Fluxo

Laços de repetição (continuação)

Exemplo:

Dada uma reta $ax+by+c=0$ e cinco pontos, faça um programa para calcular, para cada ponto, o seguinte: se o ponto estiver no primeiro quadrante calcule e informe a distância do ponto a reta caso contrário escreva uma mensagem informando que o ponto não pertence ao primeiro quadrante.

Estruturas de Controle de Fluxo

```
#include <stdio.h>
#include <math.h>
main()
{
    float a,b,c,x,y;
    int contador=1;
    printf ("Equação da reta: ax+by+c=0\n");
    printf ("Coeficiente a da reta = ");
    scanf ("%f",&a);
    printf ("\nCoeficiente b da reta = ");
    scanf ("%f",&b);
    printf ("\nCoeficiente c da reta = ");
    scanf ("%f",&c);
```

```

do
{
    printf ("\nCoordenadas do ponto %d:\n", contador);
    printf ("\nCoordenada x do ponto = ");
    scanf ("%f",&x);
    printf ("\nCoordenada y do ponto = ");
    scanf ("%f",&y);
    if (x>=0.0 && y>=0)
        printf ("\nA distancia do ponto a reta eh: %f",
            fabs(a*x+b*y+c)/(float)sqrt(pow(a,2)+pow(b,2)));
    else
        printf ("\nO ponto nao esta no primeiro quadrante!");
    contador++;
} while (contador<=5);
}

```

Estruturas de Controle de Fluxo

Laços de repetição

```
for (<instrução1>;<condição>;<instrução2>)  
{  
    <instrução3>  
    .  
    .  
    .  
}
```

Obs.: A condição é uma expressão lógica.

Estruturas de Controle de Fluxo

Laços de repetição (*for*)

1^a executar a instrução1

2^a avaliar a condição, se verdadeira executar 3^a,
se falso sair do laço

3^a executar a instrução3

4^a executar a instrução2

5^a vá para o 2^a passo

Estruturas de Controle de Fluxo

Laços de repetição (observações)

```
for (i=0,j=0;i<10&&j<20;i++,j+=2)
```

```
{
```

```
  .
```

```
  .
```

```
  .
```

```
}
```

```
for (i=0;i<1000;i++);
```

```
printf (“\nEscreve qualquer coisa!\n”);
```

```
/*O printf anterior não está contido no for!*/
```

Estruturas de Controle de Fluxo

Laços de repetição (continuação)

Exemplo:

Dada uma reta $ax+by+c=0$ e cinco pontos, faça um programa para calcular, para cada ponto, o seguinte: se o ponto estiver no primeiro quadrante calcule e informe a distância do ponto a reta caso contrário escreva uma mensagem informando que o ponto não pertence ao primeiro quadrante.

Estruturas de Controle de Fluxo

```
#include <stdio.h>
#include <math.h>
main()
{
    float a,b,c,x,y;
    int contador;
    printf ("Equação da reta: ax+by+c=0\n");
    printf ("Coeficiente a da reta = ");
    scanf ("%f",&a);
    printf ("\nCoeficiente b da reta = ");
    scanf ("%f",&b);
    printf ("\nCoeficiente c da reta = ");
    scanf ("%f",&c);
```

```

for (contador=5;contador;contador--)
{
    printf ("\nCoordenadas do ponto %d:\n",6-contador);
    printf ("\nCoordenada x do ponto = ");
    scanf ("%f",&x);
    printf ("\nCoordenada y do ponto = ");
    scanf ("%f",&y);
    if (x>=0.0 && y>=0)
        printf ("\nA distancia do ponto a reta eh: %f",
            fabs(a*x+b*y+c)/(float)sqrt(pow(a,2)+pow(b,2)));
    else
        printf ("\nO ponto nao esta no primeiro quadrante!");
}
}

```

Aula Prática X – Exercícios

Exercícios

21) Adapte o programa gerado para solucionar o exercício número 18, fazendo com que o programa mostre o número de cada linha no início e no final da linha, conforme exemplo:

```
1 Sou um programa C! 1  
2 Sou um programa C! 2  
3 Sou um programa C! 3
```

Exercícios

22) Adapte sua versão do programa desenvolvido no exercício 18 para que a frase impressa faça uma escada como no exemplo:

Sou um programa C!

Sou um programa C!

Sou um programa C!

Exercícios

23a) Construa um programa que leia da entrada padrão dois números naturais A e B , respectivamente, e retorne na saída padrão o quociente e o resto da divisão de A por B . As únicas operações aritméticas que podem ser utilizadas na construção do programa são as de ***incremento***, ***subtração***. Obs.: As entradas devem ser validadas.

```
#include <stdio.h>
main()
{
    int A, B, quociente=0;
    do {
        printf ("Entre com o valor de A: ");
        scanf ("%d",&A);
    }while(A<0);
    do {
        printf ("Entre com o valor de B: ");
        scanf ("%d",&B);
    }while(B<=0);
    for (;A>=B; A-=B, quociente++);
    printf("\nO quociente da divisao de A por B eh: %d",
        quociente);
    printf("\nO resto da divisao de A por B eh: %d",A);
}
```

Exercícios

23b) Construa um programa que leia da entrada padrão dois números inteiro A e B, respectivamente, e retorne na saída padrão o quociente e o resto da divisão de A por B. As únicas operações aritméticas que podem ser utilizadas na construção do programa são as de ***incremento, decremento, subtração e soma.***