

# Leitura - 3

## Estruturas de controlo

Paulo Vieira

# Agenda

Atribuição e comparadores

Lógica booleana

A instrução IF

Ciclo While

Ciclo Do-While

Ciclo For

Instruções Break e Continue

Instrução Switch

scanf-getchar (considerações)

# Operações aritméticas: +, -, \*, / (inteiros, decimais), % (inteiros)

```
main.c
1  /*****
2
3      Online C Compiler.
4      Code, Compile, Run and Debug C program online.
5      Write your code in this editor and press "Run" button to compile and execute it.
6
7      *****/
8
9  #include <stdio.h>
10
11  int main()
12  {
13      int x=20;
14      int y=3;
15      printf("Hello World, %d,%d,%d\n",x+y,x-y,x*y);
16      printf("Hello World,%d, %d",x/y,x%y);
17      float u=2.01;
18      float v=3.234;
19      printf("\nHello World, %f,%f,%f,%f\n",u+v,u-v,u*v,u/v);
20      double z=2.01;
21      double t=3.234;
22      printf("\nHello World, %f,%f,%f,%f\n",z+t,z-t,z*t,z/t);
23
24      return 0;
25  }
26
```

# Atribuição e Comparadores

```
#include <stdio.h>
```

```
int main(){
```

```
    int x1,x2=2,x3=4;
```

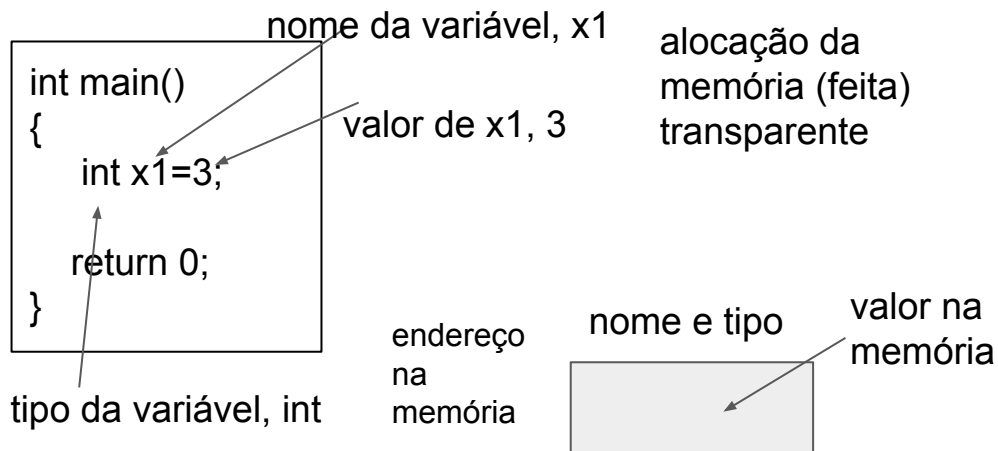
```
    x1=x2+x3; // o valor da soma de x2
```

```
                //com x3 é atribuído a x1
```

```
    printf("o valor de %d com %d é atribuído a  
%d",x2,x3,x1)
```

```
    return 0;
```

```
}
```



expressão	símbolo designação	descrição
$x1 = x2 + x3$	= símbolo de atribuição.	o soma do valor de x2 com o valor de x3 é atribuído a x1 como o seu valor
$x1 == x2$	== comparador	o valor de x1 é o mesmo que o valor de x2
$x1 != x2$	!= comparador	o valor de x1 é diferente do valor de x2
$x1 < x2$	< comparador	o valor de x1 é menor que o valor de x2
$x1 <= x2$	<= comparador	o valor de x1 é menor ou igual ao valor de x3
$x1 > x2$	> comparador	o valor de x1 é maior que o valor de x4
$x1 >= x2$	>= comparador	o valor de x1 é maior ou igual ao valor de x5

operações aritméticas:  $x+y$ ,  $x-y$ ,  $x*y$ ,  $x/y$  (divisão inteira),  $x\%y$  (resto da divisão de x por y)

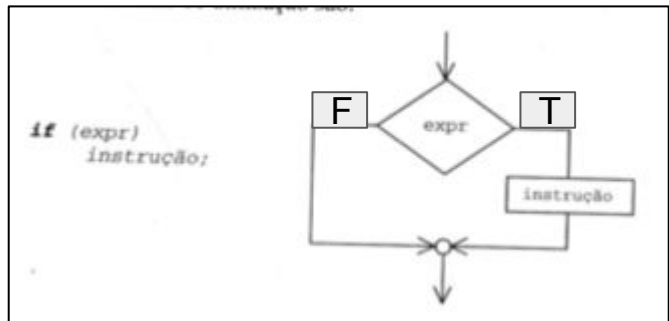
# Lógica booleana

		Não p	p e q	p ou q
p	q	!p	p && q	p    q
v	v	f	v	v
v	f	f	f	v
f	v	v	f	v
f	f	v	f	f

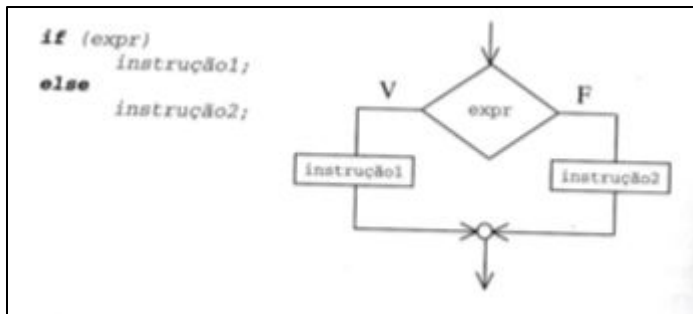
Exercício 1: Avalia as expressões que se seguem sabendo que  $x1=10$ ,  $x2=14$ ,  $x3=11$

- i)  $x1==x2$ ,
- ii)  $x1<x2$
- iii)  $x1!=x2$
- iv)  $(x1<x2) \&\& (x2<x3)$
- v)  $(x1<x2) || (x2<x3)$
- vi)  $!(x1<x2)$
- vii)  $(x1!=x2) \&\& !(x1>x3)$

# A instrução IF



```
#include <stdio.h>
int main()
{
    int ano;
    printf("\n Qual o ano que pretende testar: ");
    scanf("%d",&ano);
    if ((ano%400==0) || (ano%4==0 && ano%100!=0)){
        printf("\n O ano de %d É bissexto \n ",ano);
    }else{
        printf("\n O ano de %d NÃO é bissexto \n ",ano);
    }
    return 0;
}
```



```
#include <stdio.h>
int main()
{
    int a,b,c,max;
    printf("\n Introduza 3 valores distintos ");
    scanf("%d %d %d",&a,&b,&c);
    if(a>b){
        if(a>c) max=a;
        else max=c;
    }else{
        if(b>c) max=b;
        else max=c;
    }
    printf("\n O maior valor é %d ",max);
    printf("\n O maior valor é,%d,%d,%d,%d",a,b,c,max);
    return 0;
}
```

**Exercício:**  
Explique linha a linha cada um dos programas deste slide

# IF, Comparadores

```
#include <stdio.h>

int main(){

    int x1,x2=2,x3=4;

    x1=x2+x3; // o valor da soma de x2

               //com x3 é atribuído a x1

    printf("o valor de %d com %d é atribuído a
%d",x2,x3,x1)

    return 0;

}
```

```
#include <stdio.h>

int main()

{

    int x1=2,x2=3;

    printf("introduza dois valores: ");

    scanf("%d %d",&x1,&x2);

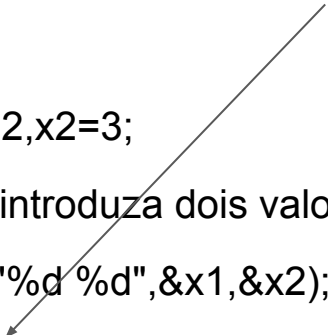
    if(x1==x2) printf("%d e %d são iguais",x1,x2);

    else printf("%d e %d são diferentes",x1,x2);

    return 0;

}
```

significa: o valor de x1 é igual  
ao valor de x2



# IF, Comparadores

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
    int x1=2,x2=3;
```

```
    printf("introduza dois valores: ");
```

```
    scanf("%d %d",&x1,&x2);
```

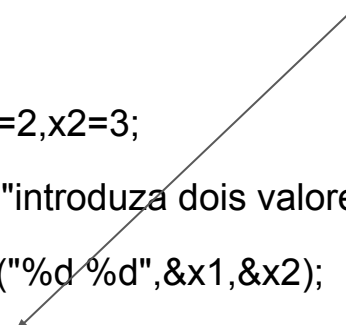
```
    if(x1!=x2) printf("%d e %d são diferentes",x1,x2);
```

```
    else printf("%d e %d são iguais",x1,x2);
```

```
    return 0;
```

```
}
```

significa: o valor de x1 é diferente do valor de x2



```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
    int x1=2,x2=3;
```

```
    printf("introduza dois valores: ");
```

```
    scanf("%d %d",&x1,&x2);
```

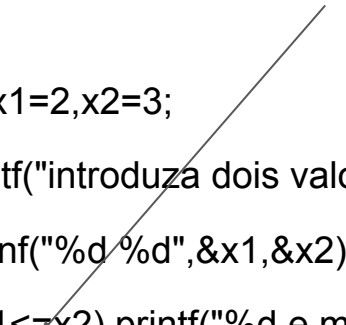
```
    if(x1<=x2) printf("%d e menor ou igual a %d",x1,x2);
```

```
    else printf("%d é maior do que %d ",x1,x2);
```

```
    return 0;
```

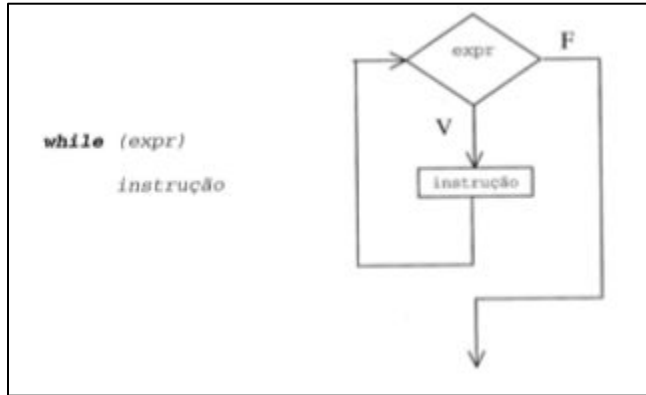
```
}
```

significa: o valor de x1 é menor ou igual ao valor de x2





# O ciclo While

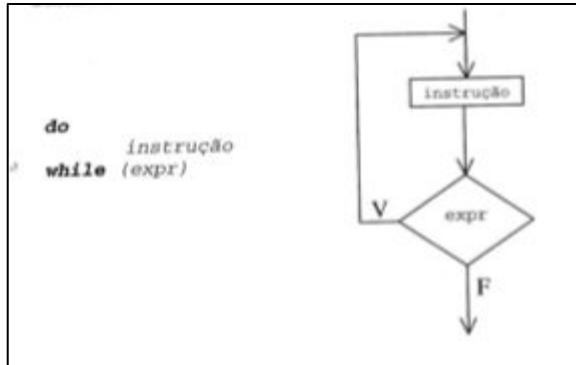


Exercício 2:  
Explique linha a linha o programa  
deste slide

O programa conta a quantidade de números (números inteiros) inseridos através do teclado (deve introduzir sempre numeros, o programa não está preparado para outras introduções)

```
#include <stdio.h>
int main(){
    long c_lidos=0;
    int c;
    scanf("%d",&c);
    while(c!= -1){
        scanf("%d",&c);
        c_lidos++; // abrevia c_lidos = c_lidos+1;
    }
    printf("\n Foram lidos %ld caracteres!",c_lidos);
    return 0;
}
```

# O ciclo Do-While



## Exercício 3:

i) pesquise o comando getchar() e explique o seu funcionamento

ii) Explique linha a linha cada um dos programas deste slide

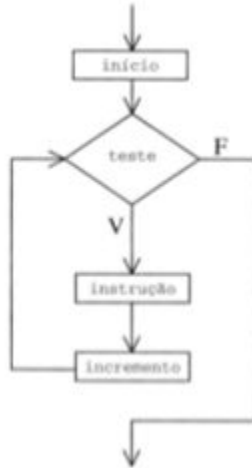
Função que lê um dígito e o programa que a chama. A função lê do teclado até que seja lido um dígito. O dígito é depois retornado

```
# include <stdio.h>  
int le_digito(); // protótipo de função  
int main(){  
    int c;  
    c=le_digito();  
    printf("O dígito lido foi %c com código ASCII %d ",c,c);  
    return 0;  
}
```

```
int le_digito(){  
    int c;  
    do{  
        c=getchar(); // retorna o valor inteiro na  
                    // tabela ASCII do caractere introduzido  
    }while (c<'0' || c>'9');  
    return c;  
}
```

# O ciclo For

`for(início; teste; incremento)  
instrução`



Exercício 4:  
Explique linha a linha cada um dos  
programa deste slide

Cálculo do factorial de um número introduzido pelo  
utilizador

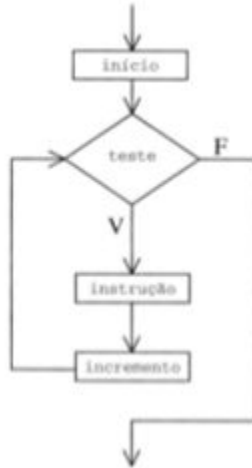
Resolução nº 1

```
#include <stdio.h>
unsigned long factorial(int num);
int main(){
    int numero;
    printf("\n CALCULO DO FACTORIAL \n");
    printf("\n Introduza um numero: ");
    printf("\n CALCULO DO FACTORIAL \n");
    scanf("%d",&numero);
    printf("%d!= %ld \n", numero,factorial(numero));
    return 0;
}

unsigned long factorial(int num){
    unsigned long res=1;
    for(; num>1;num--){
        res *=num; // abreviatura de, res =res * num;
    }
    return (res);
}
```

# O ciclo For

*for(início; teste; incremento)  
instrução*



## Exercício 5:

Explique linha a linha cada um dos programas deste slide

Cálculo do factorial de um número introduzido pelo utilizador

Resolução nº 2

```
#include <stdio.h>
long factorial(int num);
int main(){
    int numero;
    printf("\n CALCULO DO FACTORIAL \n");
    printf("\n Introduza um numero: ");
    printf("\n CALCULO DO FACTORIAL \n");
    scanf("%d",&numero);
    printf("%d!= %ld \n", numero,factorial(numero));
    return 0;
}

long factorial(int num){
    long res=1;
    if(num<0) res=-1;
    for(int val=1; val<=num;val++) // vall ++, abrevia val=val+1;
        res *=val; // abreviatura de, res =res * val;
    return (res);
}
```

# Instruções Switch

## SWITCH

```
switch(expressão){  
  
    case constante: instrução  
                    break;  
  
    case constante: instrução  
                    break;  
  
    .....  
  
    default: instrução  
            break;  
  
}
```

### Exercício 6:

Explique linha a linha cada um dos programas deste slide

```
#include <stdio.h>  
#define FIM '0'  
int main(){  
    printf("\n 1-Opção ");  
    printf("\n 2-Opção ");  
    printf("\n 0-Sair ");  
    int op;  
    do{  
        if(op==10){  
            printf("\n 1-Opção ");  
            printf("\n 2-Opção ");  
            printf("\n 0-Sair ");  
        }  
        op=getchar();  
        switch(op){  
            case '1': printf("Opção 1\n");  
                      break;  
            case '2': printf("Opção 2\n");  
                      break;  
            case FIM: printf("Pedi para terminar\n");  
                      break;  
            case 10: printf("\n");  
                     break;  
            default: printf("Opção inválida\n");  
                     break;  
        }  
    }while(op!=FIM);  
    return 0;  
}
```

programa que exemplifica  
um menu

## Instrução break|continue

### Exercício 7:

Explique cada um dos programas deste slide enfatizando o que os diferencia

```
#include <stdio.h>
int main(){
    for(int i=1;i<100;i++){
        if(i%3==0) break;
        printf("o valor do inteiro gerado é: %d \n",i);
    }
    return 0;
}
```

```
#include <stdio.h>
int main(){
    for(int i=1; i<100;i++){
        if(i%3==0) continue;
        printf("o valor do inteiro gerado é: %d \n",i);
    }
    return 0;
}
```

```
#include <stdio.h>
int main(){
    for(int i=1;i<100;i++){
        if(i%3==0) printf("Ok, ");
        printf("o valor do inteiro gerado é: %d \n ",i);
    }
    return 0;
}
```

# scanf-getchar() (considerações)

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
    char chr;
```

```
    int x;
```

```
    printf("Enter a inteiro: ");
```

```
    scanf("%d",&x);
```

```
    printf("%d\n",x);
```

```
    printf("Enter a character: ");
```

```
    scanf(" %c",&chr);
```

```
    printf("You entered %c,%d.", chr,x);
```

```
    return 0;
```

```
}
```

quando se introduz o inteiro e dp faz-se enter o compilador toma o enter como o próximo caracter

ultrapassa o problema

é necessário colocar scanf(" %c",&chr) (com espaço)  
nao escrever scanf("%c",&chr) (sem espaço)

## scanf-gectchar() (considerações)

```
#include <stdio.h>
#include <stdio.h>
int main()
{
    int x;

    printf("Enter a character: ");
    x=getchar();
    printf("You entered %c.", x);
    return 0;
}
```



FIM