

# Estrutura de Repetição

## While



# Estrutura de Repetição **While**

O comando **WHILE** é utilizado quando necessitamos fazer um teste a cada iteração do laço.

# Comando **while**

Estrutura recomendada quando o número de repetições for desconhecido, sendo necessário um teste (condição) para interromper a repetição.

Sintaxe:

```
while (CONDIÇÃO) {  
    comando A  
    comando B  
    ...  
} //Fim while
```

Fluxo de execução:

- 1) Avalia a condição, resultando True ou False
- 2) Se resultar False, sai do laço
- 3) Se resultar True executa todos os comandos e retorna ao inicio

# Comando **while**

Alguma variável da CONDIÇÃO deverá ter seu valor alterado através de um comando de leitura ou de atribuição, antes da estrutura de repetição e dentro da estrutura (na maioria das vezes como último comando do laço).

```
int main(){ //Programa Exemplo
    int teste = 1;
    while (teste) {
        printf("\nLaco executado.\n");
        printf("\nDigite 1 para repetir ou 0 para sair: ");
        scanf("%i", &teste);
    } //fim while
}
```

**Boa prática de programação!**  
Procure comentar o código para indicar o término de um laço. Não é obrigatório mas facilita a leitura do programa.

# Comando **while**

## Exemplos errados

```
//Exemplo errado 1
while (num > 0) {
    printf("Num: ");
    scanf("%i",num);
    printf("Dobro de %i = %i", num, 2*num);
}
```

# Comando **while**

## Exemplos errados

```
//Exemplo errado 1
while (num > 0) { // ERRO: faltou inicializar "num" antes do laço.
    printf("Num: ");
    scanf("%i",num);
    printf("Dobro de %i = %i", num, 2*num);
}
```

# Comando **while**

```
//Exemplo errado2  
int quadrado, valor = 0;  
while (valor <= 0) {  
    quadrado = pow(valor,2);  
    printf("O quadrado de %i = %i", valor, quadrado);  
}
```

# Comando **while**

Este código está em *loop* infinito, ou seja, a condição sempre será True.

```
//Exemplo errado2
int quadrado;
int valor = 0;
while (valor <= 0) {
    quadrado = pow(valor,2);
    printf("O quadrado de %i = %i", valor, quadrado);
    // ERRO: faltou atualizar "valor" dentro do laço.
}
```



# Comando **while**

Exemplo: Substitua um *for* por *while*

```
//Exemplo for  
for (int k=0; k<10; k++){  
    printf("k = %i\n",k);  
}
```

# Comando **while**

Exemplo: Substitua um *for* por *while*

```
//Exemplo for
for (int k=0; k<10; k++){
    printf("k = %i\n",k);
}
```

```
//Equivalente while
int k = 0;
while (k < 10){
    printf("%i\n",k);
    k = k + 1;
}
```

# Comando **while** - Exercícios

- 1) Ler **vários** números inteiros e informar o triplo de cada um. O algoritmo acaba quando entra Zero.

# Comando **while** - Exercícios

- 1) Ler vários números inteiros e informar o triplo de cada um. O algoritmo acaba quando entra Zero.

```
// Le numeros e informa o triplo de cada um
int num,triplo;
printf("num: ");  scanf("%i",&num);
while (num != 0){
    triplo = 3*num;
    printf("O triplo de %i = %i\n", num,triplo);
    printf("\tDigite 0 para sair.\n");
    printf("\nnum: ");  scanf("%i",&num);
}
```

# Comando **while** - Exercícios

- 2) Entrar com vários números enquanto forem positivos.  
Informar quantos números foram digitados e a média deles.

## Comando **while** – Exercício 2

```
// Retorna a quantidade de numeros digitados e a media deles
float num, soma = 0, media;
int cont = 0;
printf("NUM: ");  scanf("%f",&num);
while (num > 0){
    cont++;
    soma += num;
    printf("NUM: ");  scanf("%f",&num);
}
media = soma/cont;
printf("\nForam digitados %i numeros.",cont);
printf("\nA media dos numeros = %.1f\n", media);
```

## Comando **while** – Exercício 2

```
// Retorna a quantidade de números digitados e a média deles
float num, soma = 0, media;
int cont = 0;
printf("NUM: ");  scanf("%f",&num);
while (num > 0){
    cont++;
    soma += num;
    printf("NUM: ");  scanf("%f",&num);
}
if (cont>0){
    media = soma/cont;
    printf("\nForam digitados %i numeros.",cont);
    printf("\nA media dos numeros = %.1f\n", media);
}
```

# Comando **while** - Exercícios

- 3) Faça um algoritmo para calcular a área de um triângulo retângulo. Esse programa não pode permitir a entrada de dados inválidos, ou seja, medidas menores ou iguais a 0.



## Comando **while** – Exercício 3

```
// Cálculo da área de um triângulo retângulo
float area, base=0, alt=0;
while (base <= 0){
    printf("Base: ");
    scanf("%f", &base);
}
while (alt <= 0){
    printf("Altura: ");
    scanf("%f", &alt);
}
area = base * alt / 2;
printf("Area = %5.2f", area);
```

# Comando **while** - Exercícios

- 4) Dados **vários** números inteiros positivos com três algarismos, informar se o algarismo da casa das centenas é PAR ou ÍMPAR. Interromper o laço se o número tiver menos ou mais de três algarismos.

# Comando **while** – Exercício 4

```
// Retornar se algarismo da casa das centenas é PAR ou ÍMPAR
int num, cent;
printf("NUM = ");
scanf("%i", &num);
while (99 < num && num < 1000){
    cent = num/100;
    if (cent%2 == 0)
        printf("Numero %i PAR\n", cent);
    else
        printf("Numero %i IMPAR\n", cent);
    printf("\nNUM = ");
    scanf("%i", &num);
}
```

# Comando **while** - Exercícios

- 5) Ler vários números inteiros e informar quantos entre 100 e 200 foram digitados. Quando o valor 0 (zero) for lido, o algoritmo deverá cessar sua execução.

# Comando **while** – Exercício 5

```
// Informa quantos números entre 100 e 200 foram digitados
int num, cont=0;
printf("NUM = ");
scanf("%i", &num);
while (num != 0){
    if (100 < num && num < 200)
        cont += 1;
    printf("NUM = ");
    scanf("%i", &num);
}
printf("\nForam digitados %i numeros entre 100 e 200.\n", cont);
```

# Comando **while** - Exercícios

- 6) Faça um algoritmo que leia uma quantidade indeterminada de números inteiros positivos e conte quantos deles estão nos seguintes intervalos: [0-25], [26-50], [51-75] e [76-100]. A entrada de dados deverá terminar quando for lido um número negativo.

# Comando **while** – Exercício 6

```
// Informa a quantidade de número nos intervalos
int num,cont1,cont2,cont3,cont4;
printf("NUM = ");  scanf("%i", &num);
cont1 = cont2 = cont3 = cont4 = 0;
while (num >= 0){
    if (num <= 25)
        cont1 += 1;
    else if (num <= 50)
        cont2 += 1;
    else if (num <= 75)
        cont3 += 1;
    else if (num <=100)
        cont4 += 1;
    printf("NUM = ");      scanf("%i", &num);
}
printf("\nQuant. de numeros no intervalo [0-25] = %i", cont1);
printf("\nQuant. de numeros no intervalo [26-50] = %i", cont2);
printf("\nQuant. de numeros no intervalo [51-75] = %i", cont3);
printf("\nQuant. de numeros no intervalo [76-100] = %i", cont4);
```