



# Linguagem de Programação

## Laços de Repetição

*for() / while() / do{}while()*

**Gil Eduardo de Andrade**





# Laços de Repetição

## Introdução

- Um laço de repetição permite que um determinado trecho de código seja executado repetidamente, enquanto uma determinada condição, pré-definida, for verdadeira;
- A linguagem C possui, basicamente, três tipos de laços de repetição: ***for(;;)***, ***while()*** e ***do{}while();***





# Laços de Repetição

## Exemplos de Uso:

- Um laço de repetição pode ser usado em diversos casos, como por exemplo:
  - Aplicativos que possuem menus, onde o usuário pode selecionar uma opção de cada vez, até escolher sair do programa;
  - Em rotinas mais simples, onde precisamos percorrer ou inicializar estruturas como vetores e matrizes;





# Laço de Repetição *for(;;)*

- Para que seja possível utilizar o laço de repetição *for(;;)* (tradução: *para(;;)*) é necessário definir, por padrão, ao menos um dos seus parâmetros, relativo a condição de parada;
- Contudo o laço *for(;;)* ainda permite que sejam utilizados mais dois parâmetros em sua sintaxe: *inicialização das variáveis* e *operação matemática* sobre elas;



# Laço de Repetição *for(;;)*

## Sintaxe

**1º Parâmetro:** inicialização das variáveis, podendo ser apenas uma ou várias;

**2º Parâmetro:** condição de parada, o laço irá se repetir enquanto ela for verdadeira, tal condição pode conter operadores lógicos como “&&” e “||”;

**3º Parâmetro:** operação matemática aplicada a(s) variável(eis) inicializada(s) no primeiro parâmetro, normalmente é uma operação de incremento (++) ou de decremento (--) sobre o(s) valor(es) dessa(s) variável(eis);

```
for(a=0; a<5; a++) {
```

```
    // Código a ser executado repetidamente
```

```
}
```

Término da condição de parada;

Término da inicialização das variáveis;

# Laço de Repetição *for(;;)*

## Exemplo Básico:

```
#include <stdio.h>

int main() {

    char resp = 's';

    for(;resp == 's' || resp == 'S'){
        printf("Deseja continuar (s) ou (n)? ");
        __fpurge(stdin); //fflush(stdin)
        scanf("%c", &resp);
    }

    return 0;
}
```

Observe que o laço de repetição *for()* está utilizando apenas um de seus parâmetros, referente a sua condição de parada. Importante observar também que a condição possui o operador lógico de combinação *ou (||)*, ou seja, se uma das condições (*resp == 's'* ou *resp == 'S'*) forem verdadeiras, a condição como um todo também será, e o laço continuará executando o trecho de código definido entre as *chaves ({})*



# Laço de Repetição *for(;;)*

## Exemplo Básico:

```
g1l3du4rd0@asus-ultrabook-g1l:/run/media/g1l3du4rd0/GEA/VirtualBox/Compartilhada/IFPR/2015/PC/aula07/exemplos... x
```

```
Arquivo Editar Ver Pesquisar Terminal Ajuda
```

```
[g1l3du4rd0@asus-ultrabook-g1l pc_exdoc07]$ gcc for_basico.c  
[g1l3du4rd0@asus-ultrabook-g1l pc_exdoc07]$ ./a.out  
Deseja continuar (s) ou (n)? s  
Deseja continuar (s) ou (n)? s  
Deseja continuar (s) ou (n)? n  
[g1l3du4rd0@asus-ultrabook-g1l pc_exdoc07]$
```



# Laço de Repetição *for(;;)*

## Exemplo Simples:

```
#include <stdio.h>

int main() {

    int a;

    for(a=0; a<5; a++){
        printf("%i ", a);
    }

    printf("\n");
    return 0;
}
```

Observe que o laço de repetição *for()* está utilizando todos os seus três parâmetros, referentes a inicialização de variáveis, condição de parada e operação matemática. Importante observar também que a condição do laço é única, ou seja, enquanto o valor da variável "*a*" for menor que 5 (*a<5*) o laço continuará executando o trecho de código definido entre as *chaves ({}).*

Quando o código-fonte chega no comando *for(;;)*, para executá-lo, o primeiro parâmetro deste (*a=0*) é executado, esse procedimento ocorre apenas uma única vez. Após isso é executado o segundo parâmetro (*a<5*), sendo a condição verdadeira o fluxo de código entra no laço (*{}*). Quando o código entre as chaves termina o terceiro parâmetro é executado (*cont++*) e a condição é novamente testada, tal procedimento continuará enquanto a condição for verdadeira.





# Laço de Repetição *for(;;)*

## Exemplo Simples:

```
g1l3du4rd0@asus-ultrabook-g1l:/run/media/g1l3du4rd0/GEA/VirtualBox/Compartilhada/IFPR/2015/PC/aul... x
Arquivo  Editar  Ver  Pesquisar  Terminal  Ajuda
[g1l3du4rd0@asus-ultrabook-g1l pc_exdoc07]$ gcc for.c
[g1l3du4rd0@asus-ultrabook-g1l pc_exdoc07]$ ./a.out
0 1 2 3 4
[g1l3du4rd0@asus-ultrabook-g1l pc_exdoc07]$ █
```



# Laço de Repetição *for(;;)*

## Exemplo Avançado:

```
#include <stdio.h>

int main() {
    int a, b;

    for(a=0, b=5; a<6 && b>-1; a++, b--){
        printf("%i %i\n", a, b);
    }

    return 0;
}
```

Observe que o laço de repetição *for()* está utilizando todos os seus três parâmetros, referentes a inicialização de variáveis, condição de parada e operação matemática. Observe ainda que mais de uma variável é utilizada pelo laço *for()*, influenciando sua condição de parada (*a<6 && b>-1*) e o trecho de código definido entre as *chaves ({}).*



# Laço de Repetição *for(;;)*

## Exemplo Avançado:

```
g1l3du4rd0@asus-ultrabook-g1l/run/media/g1l3du4rd0/GEA/VirtualBox/Compartilhada/IFPR/2015/PC/aula07/exemplos/pc_... x
Arquivo Editar Ver Pesquisar Terminal Ajuda
[g1l3du4rd0@asus-ultrabook-g1l pc_exdoc07]$ gcc for_avancado.c
[g1l3du4rd0@asus-ultrabook-g1l pc_exdoc07]$ ./a.out
0 5
1 4
2 3
3 2
4 1
5 0
[g1l3du4rd0@asus-ultrabook-g1l pc_exdoc07]$ █
```





# Laço de Repetição *while()*

- Para que seja possível utilizar o laço de repetição ***while()*** (*tradução: equanto()*) é necessário definir, por padrão, o parâmetro (único) referente a sua condição de parada;
- Portanto a condição de parada laço ***while()*** deve ser definida entre os seus parênteses, como por exemplo:
  - ***Exemplo: while(a < 5) { }***





# Laço de Repetição *while()*

## Sintaxe

**Parâmetro:** condição de parada, o laço irá se repetir enquanto ela for verdadeira, tal condição pode conter operadores lógicos como “&&” e “||”;



```
while(a < 5) {  
    // Código a ser executado repetidamente  
}
```



# Laço de Repetição *while()*

## Exemplo Básico:

```
#include <stdio.h>

int main() {

    char resp = 's';

    while(resp != 'n' && resp != 'N'){
        printf("Deseja continuar (s) ou (n)? ");
        __fpurge(stdin); //fflush(stdin)
        scanf("%c", &resp);
    }

    return 0;
}
```

Observe que a condição, único parâmetro do laço *while()*, possui o operador lógico de combinação *e (&&)*, ou seja, as duas condições (*resp != 'n'* e *resp != 'N'*) precisam ser verdadeiras para que a condição como um todo também seja, e o laço continue executando o trecho de código definido entre as *chaves ({}).*



# Laço de Repetição *while()*

## Exemplo Básico:

```
g1l3du4rd0@asus-ultrabook-g1l:/run/media/g1l3du4rd0/GEA/VirtualBox/Compartilhada/IFPR/2015/PC/aula07/exemplos/pc.... x
Arquivo Editar Ver Pesquisar Terminal Ajuda
[g1l3du4rd0@asus-ultrabook-g1l pc_exdoc07]$ gcc while_basico.c
[g1l3du4rd0@asus-ultrabook-g1l pc_exdoc07]$ ./a.out
Deseja continuar (s) ou (n)? s
Deseja continuar (s) ou (n)? s
Deseja continuar (s) ou (n)? n
[g1l3du4rd0@asus-ultrabook-g1l pc_exdoc07]$ █
```





# Laço de Repetição *while()*

- O laço de repetição *while()* também pode ser utilizado de forma análoga ao *for()* no que diz respeito a manipulação de uma variável inteira e de uma operação matemática;
- Entretanto, para tal, é necessário iniciar a variável inteira antes do início do laço e ainda, especificar dentro dele (entre as chaves - *{}*) a operação matemática a ser efetuada sob variável inteira em questão;





# Laço de Repetição *while()*

## Exemplo Análogo ao *for()*:


```
#include <stdio.h>

int main() {

    int a;

    a = 0;
    while(a<5){
        printf("%i ", a);
        a++;
    }

    printf("\n");
    return 0;
}
```



Observe que uma linha antes da codificação chegar ao laço de repetição *while()* o primeiro procedimento a ser efetuado é iniciar a variável “*a*” (*a=0*). O laço *while()*, como visto anteriormente, possui apenas um parâmetro, que é a sua condição (*a<5*). Após toda a codificação que deve ser executada pelo laço, coloca-se então, por último, a operação matemática que alterará o valor da variável “*a*” e assim, após um determinada quantidade de repetições, esta fará com que a condição deixe de ser verdadeira e o laço seja finalizado.



# Laço de Repetição *while()*

## Exemplo Análogo ao *for()*:

```
g1l3du4rd0@asus-ultrabook-g1l:/run/media/g1l3du4rd0/GEA/VirtualBox/Compartilhada/IFPR/2015/PC/aula07/exemplo... x
Arquivo  Editar  Ver  Pesquisar  Terminal  Ajuda
[g1l3du4rd0@asus-ultrabook-g1l pc_exdoc07]$ gcc while_como_for.c
[g1l3du4rd0@asus-ultrabook-g1l pc_exdoc07]$ ./a.out
0 1 2 3 4
[g1l3du4rd0@asus-ultrabook-g1l pc_exdoc07]$
```





# Laço de Repetição *do{} while()*

- Para que seja possível utilizar o laço de repetição *do{} while()* (*tradução: faça{} enquanto()*) é necessário definir, por padrão, o parâmetro (único) referente a sua condição de parada;
- O laço de repetição *do{} while()* funciona de forma análoga ao laço *while()*, porém, diferentemente deste, o fluxo de código (dentro do laço) é executado antes que sua condição de parada seja testada;



# Laço de Repetição *do{} while()*

## Exemplo Básico:

```
#include <stdio.h>

int main() {
    char resp;
    do {
        printf("Deseja continuar (s) ou (n)? ");
        __fpurge(stdin); //fflush(stdin)
        scanf("%c", &resp);
    } while(resp != 'n' && resp != 'N');

    return 0;
}
```

Observe que a variável *resp* não precisa ser inicializada com um valor, como visto nos laços anteriores, isso é possível pelo fato do código contido no laço *do{} while()* ser executado antes que sua condição seja testada, ou seja, o fluxo já encontra-se dentro do laço quando a condição de parada é testada pela primeira vez. Já os laços *for(;;)* e *while()* primeiro testam suas condições de parada, antes que o fluxo de código entre no laço, e por isso suas condições precisam ser verdadeiras no momento em que o código-fonte chega a eles.



# Laço de Repetição *do{} while()*

## Exemplo Básico:

```
g1l3du4rd0@asus-ultrabook-g1l:/run/media/g1l3du4rd0/GEA/VirtualBox/Compartilhada/IFPR/2015/PC/aula07/exemplo... x
Arquivo  Editar  Ver  Pesquisar  Terminal  Ajuda
[g1l3du4rd0@asus-ultrabook-g1l pc_exdoc07]$ gcc dowhile_basico.c
[g1l3du4rd0@asus-ultrabook-g1l pc_exdoc07]$ ./a.out
Deseja continuar (s) ou (n)? s
Deseja continuar (s) ou (n)? s
Deseja continuar (s) ou (n)? n
[g1l3du4rd0@asus-ultrabook-g1l pc_exdoc07]$ █
```





# Laço de Repetição *do{} while()*

- O laço de repetição *do{} while()* também pode ser utilizado de forma análoga ao *for()* no que diz respeito a manipulação de uma variável inteira e de uma operação matemática;
- Entretanto, para tal, é necessário iniciar a variável inteira antes do início do laço e ainda, especificar dentro dele (entre as chaves - *{}*) a operação matemática a ser efetuada sob variável inteira em questão;



# Laço de Repetição *do{} while()*

## Exemplo Básico:

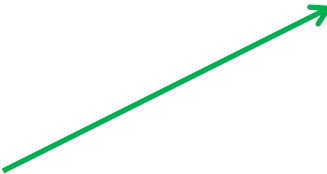
```
#include <stdio.h>

int main() {

    int a;

    a = 0;
    do {
        printf("%i ", a);
        a++;
    } while(a<5);

    printf("\n");
    return 0;
}
```



Observe que uma linha antes da codificação chegar ao laço de repetição *do{} while()* o primeiro procedimento a ser efetuado é iniciar a variável “*a*” (*a=0*). O laço *do{} while()*, como visto anteriormente, possui apenas um parâmetro, que é a sua condição (*a<5*). Após toda a codificação que deve ser executada pelo laço, coloca-se então, por último, a operação matemática que alterará o valor da variável “*a*” e assim, após um determinada quantidade de repetições, esta fará com que a condição deixe de ser verdadeira e o laço seja finalizado.





# Laço de Repetição *do{} while()*

## Exemplo Básico:

```
g1l3du4rd0@asus-ultrabook-g1l:run/media/g1l3du4rd0/GEA/VirtualBox/Compartilhada/IFPR/2015/PC/aula07/exemplos/... x
Arquivo Editar Ver Pesquisar Terminal Ajuda
[g1l3du4rd0@asus-ultrabook-g1l pc_exdoc07]$ gcc dowhile_como_for.c
[g1l3du4rd0@asus-ultrabook-g1l pc_exdoc07]$ ./a.out
0 1 2 3 4
[g1l3du4rd0@asus-ultrabook-g1l pc_exdoc07]$ █
```







# Operadores de Condição

## Exemplos Utilizados no Documento

[http://www.gileduardo.com.br/ifpr/pci/downloads/pc\\_exdoc07.zip](http://www.gileduardo.com.br/ifpr/pci/downloads/pc_exdoc07.zip)

## Mais Exemplos sobre o Conteúdo

[http://www.gileduardo.com.br/ifpr/pci/downloads/pc\\_ex07.zip](http://www.gileduardo.com.br/ifpr/pci/downloads/pc_ex07.zip)

## Exercícios sobre o Conteúdo

[http://www.gileduardo.com.br/ifpr/lp/downloads/lp\\_pratica07.pdf](http://www.gileduardo.com.br/ifpr/lp/downloads/lp_pratica07.pdf)

