

ENSINO MÉDIO INTEGRADO - INFORMÁTICA

Disciplina de Linguagem de Programação

Aula 06: Laço de Repetição - *while()*

Gil Eduardo de Andrade

Conceitos Preliminares

(<https://docs.microsoft.com/pt-br/cpp/c-language/?view=msvc-170>)

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:

- `while()`
- `do{ } while()`
- `for(;;)`

Utilização Prática

Um laço de repetição pode ser usado em diversos casos, como por exemplo:

- Aplicativos que possuem menus, onde o usuário pode selecionar, repetidamente, uma opção por vez, até escolher sair do programa;
- Em rotinas mais simples, onde precisamos percorrer ou inicializar estruturas como vetores e matrizes;

Laço de Repetição *while()*

Para que seja possível utilizar o laço de repetição ***while()*** (enquanto) é necessário definir um parâmetro de condição, que determina o momento de parada da repetição. Portanto, a condição de parada laço de repetição ***while()*** deve ser definida entre os seus parênteses. Exemplo de uso:

while(tempo < 50) {

```
printf("%i ", tempo);  
tempo++;  
}
```

No exemplo anterior, o laço de repetição irá executar as duas linhas até que a variável tempo alcance o valor 50, visto que a condição, assim como no comando if(), é tempo < 50. Efetuando uma leitura/interpretação, já traduzida, do comando while(), dizemos que: “enquanto a variável tempo for menor que 50, as linhas printf(); e tempo++; devem ser executadas repetidamente”. Supondo que o valor inicial da variável tempo seja “1”, teríamos como resultado para o exemplo, a seguinte saída, no terminal :

1 2 3 4 5 6 7 8 9 10 ... 49

Isso ocorre porque o comando **printf()** está exibindo o valor da variável **tempo** a cada repetição. Ao mesmo tempo que o comando **tempo++** está aumentando em “1” o valor da variável **tempo**.

Sintaxe

Parâmetro: condição de parada, o laço se repete enquanto for verdadeira.

```
while( tempo < 50 ) {  
    // código que será repetido  
}
```



Codificação – Linguagem de Programação C

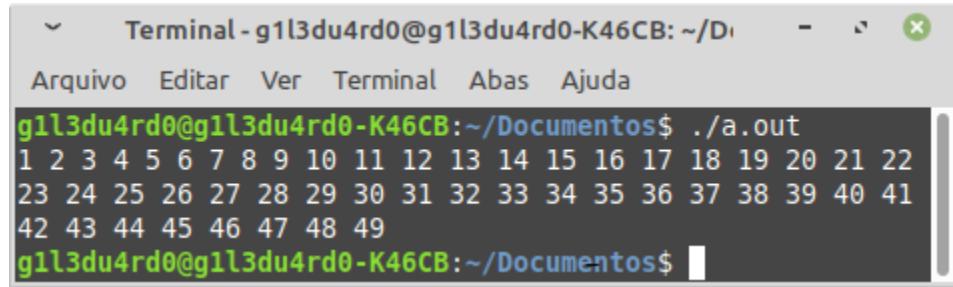
Exemplo Básico - Variável “int”

```
#include <stdio.h>

int main() {
    int tempo;
    // inicializa a variável tempo com valor 1
    tempo = 1;
    // repete as linhas entre as "{}" enquanto o valor da variável tempo for menor que 50
    while(tempo < 50) {
        printf("%i ", tempo);
        // incrementa em "1" o valor da variável tempo
        tempo++;
    }
    printf("\n");
    return 0;
}
```

Variável “int” como condição.

(Arquivo-fonte: 06 - Laço de Repetição - while/basico_int.c)



```
Terminal - g1l3du4rd0@g1l3du4rd0-K46CB: ~/Documentos - x
Arquivo Editar Ver Terminal Abas Ajuda
g1l3du4rd0@g1l3du4rd0-K46CB:~/Documentos$ ./a.out
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22
23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41
42 43 44 45 46 47 48 49
g1l3du4rd0@g1l3du4rd0-K46CB:~/Documentos$
```

Resultado da execução do código anterior

Exemplo Básico - Variável “char”

```
#include <stdio.h>

int main() {
    char resp;
    // inicializa a variável resp com a letra 's'
    resp = 's';
    // repete as linhas entre as "{}" enquanto o valor da variável resp for 's' ou 'S'
    while(resp == 's' || resp == 'S') {
        printf("deseja continuar (s) ou (n)? ");
        scanf(" %c", &resp);
    }
    printf("\n");
    return 0;
}
```

Variável “char” como condição.

(Arquivo-fonte: 06 - Laço de Repetição - while/basico_char.c)



```
Terminal - g1l3du4rd0@g1l3du4rd0-K46CB: ~/Documents - 
Arquivo Editar Ver Terminal Abas Ajuda
g1l3du4rd0@g1l3du4rd0-K46CB:~/Documents$ ./a.out
deseja continuar (s) ou (n)? s
deseja continuar (s) ou (n)? S
deseja continuar (s) ou (n)? n

g1l3du4rd0@g1l3du4rd0-K46CB:~/Documents$
```

Resultado da execução do código anterior