

Algoritmos e Programação

Estruturas de Repetição em C
`while, do-while`



Universidade Federal do Pampa

Estruturas de Repetição

- Como vimos em pseudocódigo, as **estruturas** de **repetição** **servem** para **repetir** um mesmo **trecho** de código a **quantidade** de **vezes** que for **necessário sem** ter de **reescrevê-lo!**
- O trecho do código em repetição também é chamado de **laço**, **loop** ou **laço de repetição**.

Estruturas de Repetição

- Como vimos em pseudocódigo, as **estruturas de repetição servem** para **repetir** um mesmo **trecho** de código a **quantidade de vezes** que for **necessário sem** ter de **reescrevê-lo!**
- O trecho do código em repetição também é chamado de **laço**, **loop** ou **laço de repetição**.
- Em reação à quantidade de repetição, os laços podem ser:
 - **Pré-determinados**: Sabe-se, a priori, a quantidade de execuções.
 - **Indeterminados**: Não se sabe a quantidade de execuções.

Estruturas de Repetição

- Como vimos em pseudocódigo, as **estruturas** de **repetição** **servem** para **repetir** um mesmo **trecho** de código a **quantidade** de **vezes** que for **necessário sem** ter de **reescrevê-lo!**
- O trecho do código em repetição também é chamado de **laço**, **loop** ou **laço de repetição**.
- Em reação à quantidade de repetição, os laços podem ser:
 - **Pré-determinados**: Sabe-se, a priori, a quantidade de execuções.
 - **Indeterminados**: Não se sabe a quantidade de execuções.
- Quanto ao critério de parada, C possui 3 tipos de laços de repetição:
 - Teste no início: **while**
 - Teste no final: **do-while**
 - Variável de controle: **for** (próxima aula teórica)

Conteúdo

- Comando *while*
- Comando *do-while*

while

- Laço que **verifica** antes de cada execução se “**é permitido**” **executar** o **trecho** do algoritmo.
- É um laço que se **mantém repetindo enquanto** uma **condição** permanecer **verdadeira**.

while

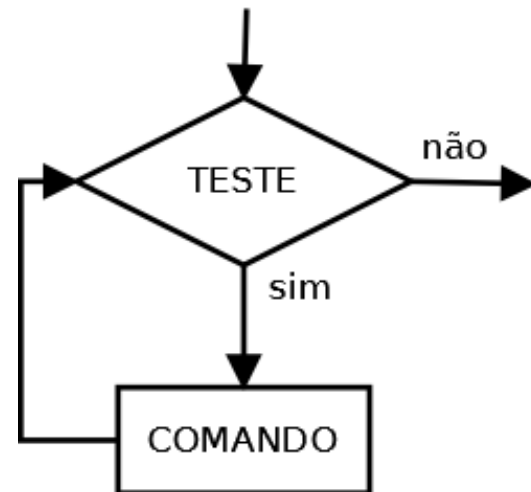
- Laço que **verifica** antes de cada execução se “**é permitido**” **executar** o **trecho** do algoritmo.
- É um laço que se **mantém repetindo enquanto** uma **condição** permanecer **verdadeira**.

O comando **while** em C é essencialmente a tradução do **enquanto** em pseudocódigo.

Sintaxe:

```
while (TESTE)  
    COMANDO ;
```

O fluxograma correspondente é:



while: exemplo

```
#include <stdio.h>

int main() {
    int x;

    printf("Qual o valor de 25-4*3:\n");
    scanf("%d", &x);
    while (x != 13)
        scanf("%d", &x);
    printf("Parabéns! Você acertou!\n");
}
```

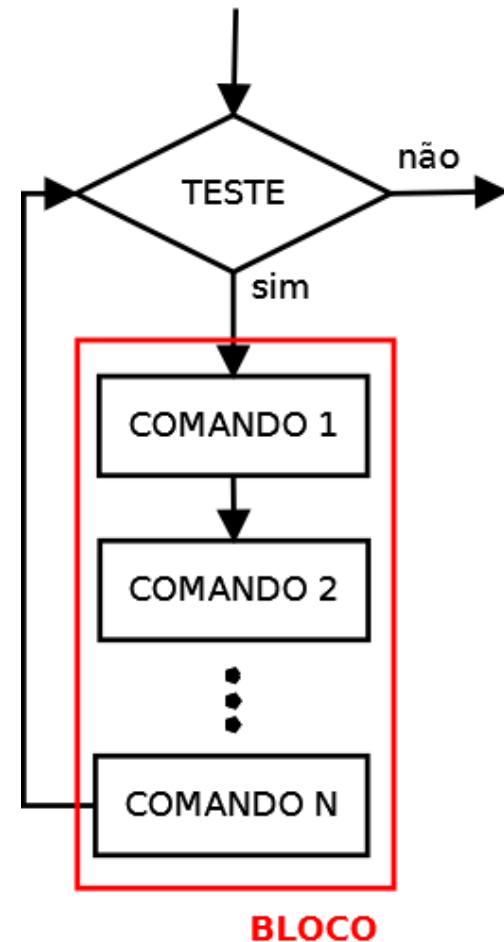

while com bloco de comando

É mais comum repetir diversos comandos enquanto uma condição é verdadeira

Para tal, usamos o while com um bloco de comandos!

```
while (TESTE) {  
    COMANDO-1 ;  
    COMANDO-2 ;  
    . . .  
    COMANDO-N ;  
}
```

O fluxograma correspondente é:



while com bloco de comando: exemplo

```
#include <stdio.h>

int main() {
    int A, B;
    printf("Digite A e B, onde A < B: ");
    scanf("%d %d", &A, &B);
    printf("Os números de %d a %d:\n", A, B);
    while (A <= B) {
        printf("%d\n", A);
        A = A + 1;
    }
}
```

while com bloco de comando: exemplo

```
#include <stdio.h>

int main() {
    int A, B;
    printf("Digite A e B, onde A < B: ");
    scanf("%d %d", &A, &B);
    printf("Os números de %d a %d:\n", A, B);
    while (!(A > B)) {
        printf("%d\n", A);
        A = A + 1;
    }
}
```

Também podemos repetir comandos **enquanto** uma **condição** for **falsa**. Para isso devemos **utilizar o operador lógico de negação (!)**.

Contadores

- Assim como o comando ***enquanto*** em pseudocódigo, o comando ***while*** em **C** não oferece um recurso para contar a quantidade de vezes que um laço é executado.

Contadores

- Assim como o comando *enquanto* em pseudocódigo, o comando *while* em **C não oferece** um **recurso** para **contar** a **quantidade** de **vezes** que um **laço** é **executado**.
 - É o **programador** que **deve estabelecer** um **modo** de **contagem**.
 - Para isso **utilizamos** um **contador**: uma **variável** com um **valor inicial** que é **incrementada** a cada **passagem** no **laço**.

Contadores: exemplo

Um programa que conta até 5. A variável n é um contador.

```
#include <stdio.h>
int main() {
    int n;
    n = 1; //inicialização da variável
    while(n <= 5) { //teste
        printf("%d\n", n); //imprime a variável n
        n++; //++ é o operador de incremento. Equivalente: n = n + 1;
    }
}
```

Conteúdo

- Comando *while*
- Comando *do-while*

Laço com execução prévia

Em alguns programas com repetições, devemos **executar** o **bloco** a ser repetido uma vez **antes** de **realizar** o **teste**.

Laço com execução prévia

Em alguns programas com repetições, devemos **executar** o **bloco** a ser repetido uma vez **antes** de **realizar** o **teste**.

Exemplo clássico é o jogo de adivinhação:

- Ler o primeiro valor
- Testa se é diferente do número secreto.
- Se for diferente, volta para a leitura. Se for igual, continua a execução do programa.

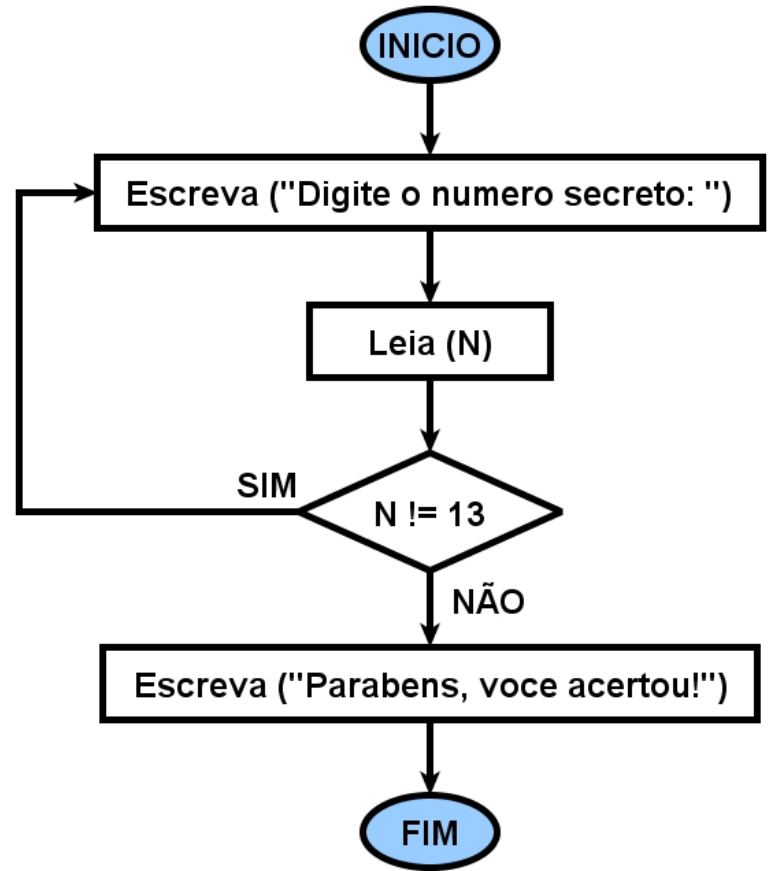
Fluxograma →

Laço com execução prévia

Em alguns programas com repetições, devemos **executar o bloco** a ser repetido uma vez **antes de realizar o teste**.

Exemplo clássico é o jogo de adivinhação:

- Ler o primeiro valor
- Testa se é diferente do número secreto.
- Se for diferente, volta para a leitura. Se for igual, continua a execução do programa.



Fluxograma →

Laço com execução prévia

Versão em C com o comando While:

```
#include <stdio.h>
int main() {
    int n, numeroSecreto = 13;
    printf("Adivinhe o numero secreto: ");
    scanf("%d", &n);
    while(n != numeroSecreto) {
        printf("Adivinhe o numero secreto: ");
        scanf("%d", &n);
    }
    printf("Parabens! Voce acertou!\n");
}
```

do-while

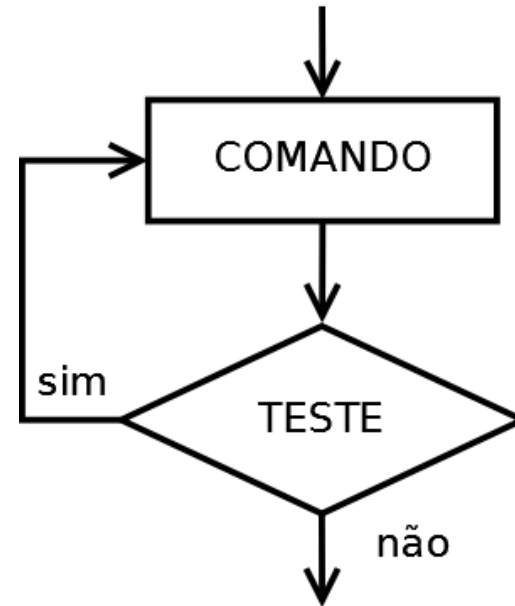
Sintaxe:

```
do {  
    COMANDO ;  
} while (TESTE) ;
```

Equivalente com **while**:

```
COMANDO ;  
while (TESTE) {  
    COMANDO ;  
}
```

Fluxograma equivalente:



Ao contrário do laço **while**, o laço **do-while** executa o primeiro comando, para então testar se é permitido executar o próximo.

do-while

O mesmo programa de adivinhação com laço **do-while**:

```
#include <stdio.h>
int main() {
    int n, numeroSecreto = 13;
    do {
        printf("Adivinhe o numero secreto:");
        scanf("%d", &n);
    } while(n != numeroSecreto);
    printf("Parabens! Voce acertou!\n");
}
```

Exercício 1

- Construa um programa em C que calcule a média aritmética de 3 alunos, onde cada aluno possui quatro notas bimestrais quaisquer fornecidas pelo usuário. Avalie a situação do aluno quanto à aprovação, e reprovação, considerando a média para aprovação superior ou igual a 6.0.
 - **Dados de entrada:** 3 x quatro notas bimestrais (N1, N2, N3, N4).
 - **Dados de Saída:** número de cada aluno acompanhado de sua média aritmética anual (MA), e a informação se o aluno foi aprovado ou reprovado.

Exercício 1: Resposta

```
#include <stdio.h>
int main() {
    int cont; //contador
    // notas e media do aluno
    float n1, n2, n3, n4, mediaAluno;
    cont = 1; // inicialização do contador
    while(cont <= 3) { //teste
        printf("Digite as 4 notas do aluno %d: ", cont);
        //leitura das 4 notas
        scanf("%f %f %f %f", &n1, &n2, &n3, &n4);
        //calcula da media
        mediaAluno = (n1+n2+n3+n4)/4;
        printf("Media do aluno %d = %g\n", cont, mediaAluno);
        //teste se aluno foi aprovado ou reprovado
        if(mediaAluno >= 6.0)
            printf("Aluno %d aprovado!\n", cont);
        else
            printf("Aluno %d reprovado!\n", cont);
        cont++; // incremento do contador
    }
}
```

Exercício 1: Resposta

```
#include <stdio.h>

int main() {
    int cont; //contador
    float n1, n2, n3, n4, mediaAluno; // notas e media do aluno
    cont = 1; // inicialização do contador
    while(cont <= 3) { //teste
        printf("Digite as 4 notas do aluno %d: ", cont);
        //leitura das 4 notas
        scanf("%f %f %f %f", &n1, &n2, &n3, &n4);
        //calcula da media
        mediaAluno = (n1+n2+n3+n4)/4;
        printf("Media do aluno %d = %g\n", cont, mediaAluno);
        //teste se aluno foi aprovado ou reprovado
        if(mediaAluno >= 6.0)
            printf("Aluno %d aprovado!\n", cont);
        else
            printf("Aluno %d reprovado!\n", cont);
        cont++; // incremento do contador
    }
}
```


Exercício 2

- Construa um programa em C utilizando o laço *do-while* que leia um conjunto indeterminado de valores inteiros positivos. No momento que o usuário digitar um valor negativo (ou nulo), o algoritmo deve encerrar a leitura de valores e informar para o usuário a quantidade de valores positivos lidos e a média dos valores positivos lidos.

Exercício 2: Resposta

```
#include <stdio.h>
int main() {
    int n, cont;
    float media;
    cont = 0;
    media = 0;
    do {
        printf("Digite um numero: ");
        scanf("%d", &n);
        if(n > 0) {
            cont++;
            media = media + n;
        }
    } while(n > 0);
    printf("Voce digitou %d valores positivos.\n", cont);
    if(cont > 0) {
        media = media / cont;
        printf("A media dos valores eh %f.\n", media);
    } else
        printf("Nenhum valor positivo foi digitado.\n");
}
```

Exercício 2: Resposta

```
#include <stdio.h>

int main() {
    int n, cont;
    float media;
    cont = 0;
    media = 0;
    do {
        printf("Digite um numero: ");
        scanf("%d", &n);
        if(n > 0) {
            cont++;
            media = media + n;
        }
    } while(n > 0);
    printf("Voce digitou %d valores positivos.\n", cont);
    if(cont > 0) {
        media = media / cont;
        printf("A media dos valores eh %f.\n", media);
    } else
        printf("Nenhum valor positivo foi digitado.\n");
}
```

Exercício 3

- Construa um programa em C que calcule a média aritmética de 3 alunos, onde cada aluno possui quatro notas bimestrais quaisquer fornecidas pelo usuário. Avalie a situação do aluno quanto à aprovação, e reprovação, considerando a média para aprovação superior ou igual a 6.0. Além disto, após calcular a média de todos alunos, informe a média da turma.
 - **Dados de entrada:** 3 x quatro notas bimestrais (N1, N2, N3, N4).
 - **Dados de Saída:** número de cada aluno acompanhado de sua média aritmética anual (MA), e a informação se o aluno foi aprovado ou reprovado. Ao final, informe a média da turma.

```
#include <stdio.h>
int main() {
    int cont; //contador
    // notas, media do aluno e acumulador para media da turma
    float n1, n2, n3, n4, mediaAluno, mediaTurma;
    cont = 1; // inicialização do contador
    mediaTurma = 0; // inicialização do acumulador
    while(cont <= 3) { //teste
        printf("Digite as 4 notas do aluno %d: ", cont);
        //leitura das 4 notas
        scanf("%f %f %f %f", &n1, &n2, &n3, &n4);
        //calcula da media
        mediaAluno = (n1+n2+n3+n4)/4.0;
        mediaTurma = mediaTurma + mediaAluno;
        printf("Media do aluno %d = %g\n", cont, mediaAluno);
        //teste se aluno foi aprovado ou reprovado
        if(mediaAluno >= 6.0)
            printf("Aluno %d aprovado!\n", cont);
        else
            printf("Aluno %d reprovado!\n", cont);
        cont++; // incremento do contador
    }
    mediaTurma = mediaTurma/3.0; // calculo da media
    printf("Media da turma = %g\n", mediaTurma);
    if(mediaTurma >= 6.0)
        printf("Turma acima da media\n");
    else
        printf("Turma abaixo da media\n");
}
```

Exercício 3: Resposta

```
#include <stdio.h>
int main() {
    int cont; //contador
    float n1, n2, n3, n4, mediaAluno, mediaTurma; // notas, media do aluno e acumulador para media da turma
    cont = 1; // inicialização do contador
    mediaTurma = 0; // inicialização do acumulador
    while(cont <= 3) { //teste
        printf("Digite as 4 notas do aluno %d: ", cont);
        //leitura das 4 notas
        scanf("%f %f %f %f", &n1, &n2, &n3, &n4);
        //calcula da media
        mediaAluno = (n1+n2+n3+n4)/4.0;
        mediaTurma = mediaTurma + mediaAluno;
        printf("Media do aluno %d = %g\n", cont, mediaAluno);
        //teste se aluno foi aprovado ou reprovado
        if(mediaAluno >= 6.0)
            printf("Aluno %d aprovado!\n", cont);
        else
            printf("Aluno %d reprovado!\n", cont);
        cont++; // incremento do contador
    }
    mediaTurma = mediaTurma/3.0; // calculo da media
    printf("Media da turma = %g\n", mediaTurma);
    if(mediaTurma >= 6.0)
        printf("Turma acima da media\n");
    else
        printf("Turma abaixo da media\n");
}
```