

BCC 201 - Introdução à Programação

# Controle de Fluxo

## Comandos de repetição

Guillermo Cámara-Chávez  
UFOP

# Comandos de Repetição (Laços) I

Como imprimir os três primeiros número ímpares?

```
#include <stdio.h>
int main()
{
    printf("1 \n");
    printf("3 \n");
    printf("5 \n");
    return 0;
}
```

## Comandos de Repetição (Laços) II

Como imprimir os 100 primeiros número ímpares?

```
#include <stdio.h>
int main()
{
    printf("1 \n");
    printf("3 \n");
    // vários comandos depois
    printf("199 \n");
    return 0;
}
```

## Comandos de Repetição (Laços) III

Os comandos de repetição são um recurso que permite que um certo trecho do código de um programa seja repetido um certo número de vezes. Na linguagem C/C++ existem três comandos de repetição:

- ▶ `while`,
- ▶ `do-while` e,
- ▶ `for`

# Ciclos repetitivos (*loops*) I

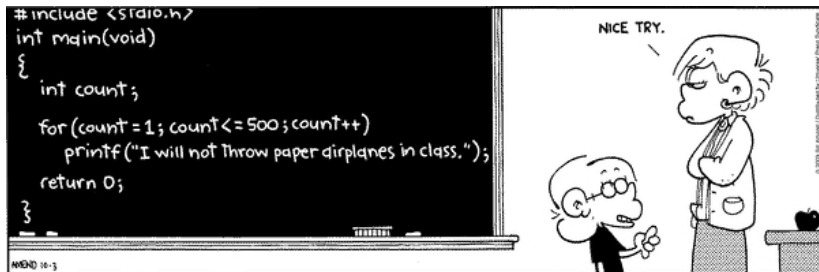
Uma professora ordenou a uma aluna que estava jogando aviões de papel na aula que ela escrevesse 500 vezes a frase: “Não devo jogar aviões de papel em aula”.

## Ciclos repetitivos (*loops*) II



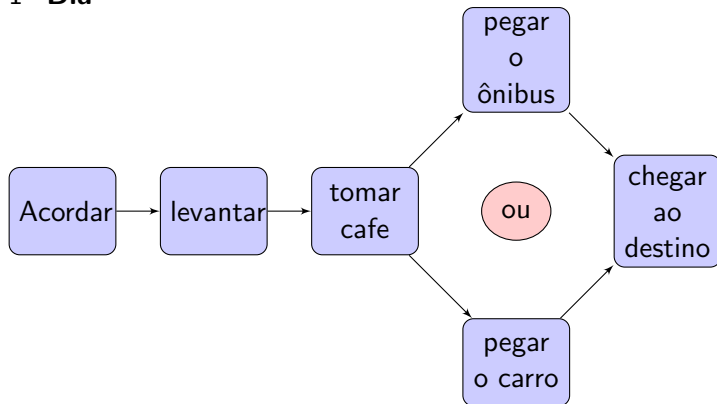
... mas a menina, mais espertinha teve uma melhor idéia ...

## Ciclos repetitivos (*loops*) III



## Ciclos repetitivos (*loops*) IV

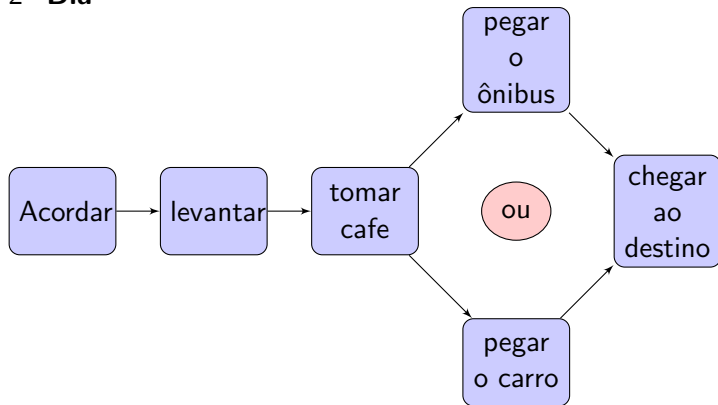
Exemplo: Seqüência de ações para chegar ao trabalho/universidade  
1º **Dia**





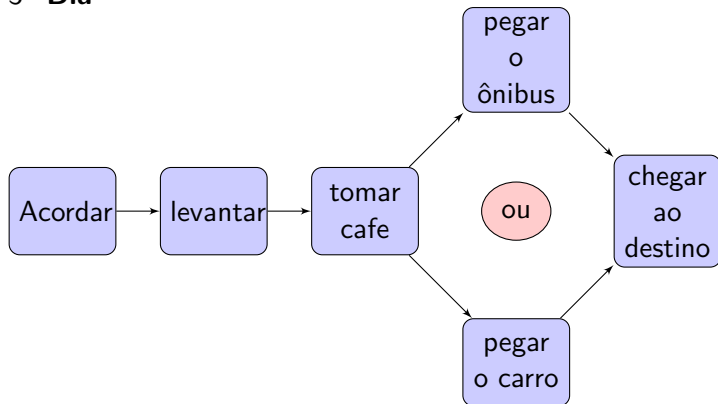
# Ciclos repetitivos (*loops*) V

Exemplo: Seqüência de ações para chegar ao trabalho/universidade  
2º Dia



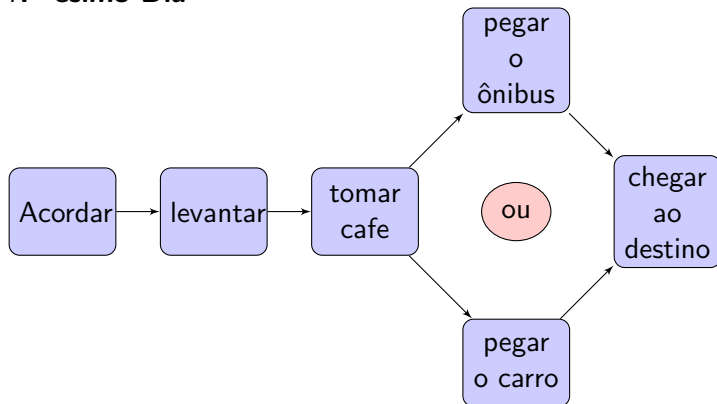
## Ciclos repetitivos (*loops*) VI

Exemplo: Seqüência de ações para chegar ao trabalho/universidade  
3º Dia



## Ciclos repetitivos (*loops*) VII

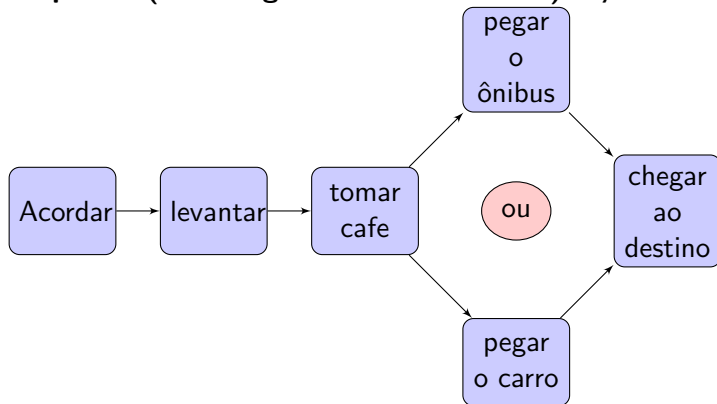
Exemplo: Seqüência de ações para chegar ao trabalho/universidade  
***N*-ésimo Dia**



## Ciclos repetitivos (*loops*) VIII

Como as ações se repetem durante um período ou até que um evento ocorra (chegar ao fim de semana), pode-se melhorar a escrita da sequência do exemplo acima, como:

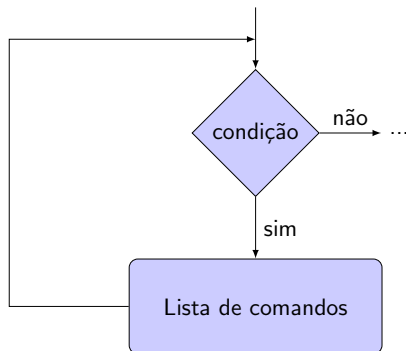
**Enquanto (não chegar ao fim de semana) faça**



# While I

O comando **while** permite que um certo trecho de programa seja executado **ENQUANTO** uma certa condição for **verdadeira**.

## Fluxograma



# While II

## Portugol

```
enquanto (condição) faça  
    lista de comandos;  
fim_enquanto
```

A forma do comando **while** em C/C++ é a seguinte:

```
while (condição)  
{  
    comando1; // comandos a serem repetidos  
    comando2; // comandos a serem repetidos  
}  
comando3; // comandos após o while
```

# While III

O funcionamento é o seguinte:

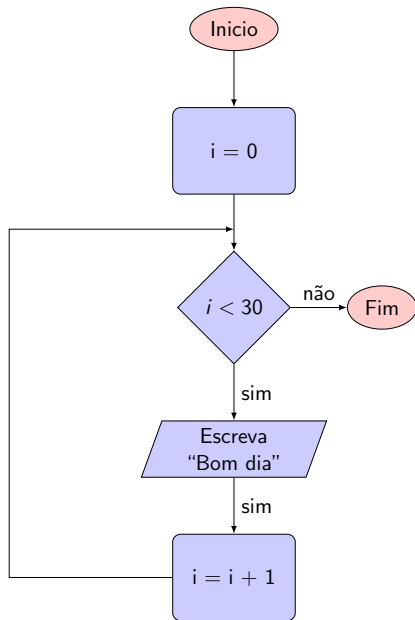
1. Testa a condição;
2. Se a condição for **falsa** então:
  - ▶ pula todos os comandos do bloco subordinado ao **while** e
  - ▶ passa a executar os comandos após o bloco do **while**
3. Se a condição for **verdadeira** então executa cada um dos comandos do bloco subordinado ao **while**
4. Após executar o último comando do bloco do **while** volta ao passo 1.

# While IV

Mostrar a mensagem “Bom Dia!” 30 vezes



# While V



## While VI

```
Inicio
  inteiro: i;
  i ← 0;
  Enquanto i < 30
    Escreva("Bom dia");
    i ← i + 1;
  fim_enquanto
Fim
```

## While VII

```
#include <stdio.h>
int main()
{
    int i;
    i = 0;
    while (i < 30)
    {
        printf("Bom Dia! \n");
        i = i + 1;
    }
    return 0;
}
```

## While VIII

Mostrar os 20 primeiros números: 1, 2, 3, . . . , 19, 20

# While IX

```
#include <stdio.h>
int main()
{
    int num;
    num = 1;
    while (num <= 20)
    {
        printf("%d \n", num);
        num++; //num = num + 1
    }
    return 0;
}
```

# While X

Mostrar os  $n$  primeiro números ímpares

# While XI

```
#include <stdio.h>
int main()
{
    int impar, i, n;
    printf("Número de impares");
    scanf("%d", &n);
    impar = 1;
    i = 0;
    while (i < n)
    {
        printf("%d \n", impar);
        impar += 2; //impar = impar + 2
        i++;
    }
    return 0;
}
```

## While XII

Outra forma:

```
#include <stdio.h>
int main()
{
    int impar, i, n;
    printf("Número de impares");
    scanf("%d", &n);
    i = 1;
    while (i < n)
    {
        impar = 2*i - 1;
        printf("%d \n", impar);
        i++;
    }
    return 0;
}
```



## While XIII

Mostrar os  $n$  primeiros números ímpares em ordem decrescente.

## While XIV

```
#include <stdio.h>
int main()
{
    int impar, i, n;
    printf("Número de impares");
    scanf("%d", &n);
    i = n;
    while (i > 0)
    {
        impar = 2*i - 1;
        printf("%d \n", impar);
        i--;
    }
    return 0;
}
```

## While XV

Implementar um programa que calcula a idade média de um grupo de pessoas. A finalização da entrada de números é dada por um -1 (solução sgte slide)

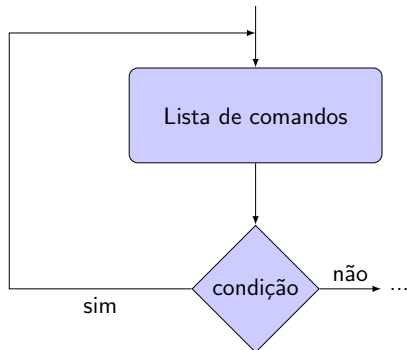
## While XVI

```
#include <stdio.h>
int main(){
    int idade = 0, cont = 0;
    float acum = 0, media = 0;
    while(idade != -1){
        printf("Insere idade (-1 para finalizar) \n");
        scanf("%d", &idade);
        if (idade >= 0)
        {
            acum = acum + idade;
            cont++;
        }
    }
    if (cont > 0){
        media = acum / cont;
        printf("a média eh: %f \n", media);
    }
    return 0;
}
```

# Comando do-while I

Parecido com o comando **while** com a diferença que a expressão é avaliada no final do ciclo repetitivo.

## Fluxograma



# Comando do-while II

## Portugol

```
Faça  
    Lista de comandos;  
enquanto (condição);
```

## C/C++

```
do {  
    comando1; // comandos a serem repetidos  
    comando2; // comandos a serem repetidos  
} while (condição);  
comando3; // comandos após o "do-while"
```

## Comando do-while III

O funcionamento é o seguinte:

1. Executa os comando dentro do bloco **do-while**;
2. Testa a condição;
3. Se a condição for **falsa** então executa o comando que está logo após o bloco subordinado ao **do-while** .
4. Se condição for **verdadeira** então volta ao passo 1.

Calcular a idade média de um grupo de pessoas. A finalização da entrada de números é dada por um -1 (solução sgte slide)

## Comando do-while IV

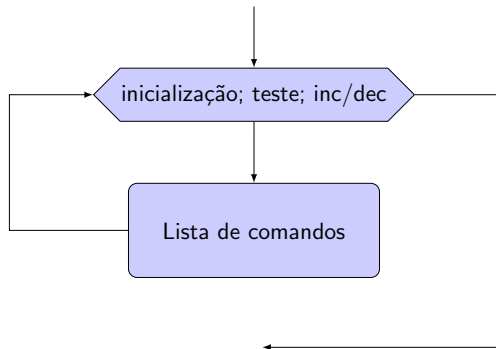
```
#include <stdio.h>
int main(){
    int idade, cont = 0;
    double acum = 0, média = 0;
    do{
        printf("Insere idade (-1 para finalizar) \n");
        scanf("%d", &idade);
        if (idade >= 0)
        {
            acum = acum + idade;
            cont++;
        }
    } while(idade != -1);
    if (cont > 0){
        media = acum / cont;
        printf("a média eh: %lf \n", media);
    }
    return 0;
}
```



# Comando for I

A forma do comando **for** é a seguinte:

## Fluxograma



# Comando for II

## Portugol

```
para variável de valor_inicial até valor_final faça  
    lista de comandos;  
fim_para
```

## C/C++

```
for (inicialização; teste; incremento/decremento)  
{  
    comando1; //comandos a serem repetidos  
    comando2; //comandos a serem repetidos  
}  
comando3; /comandos após o for
```

## Comando for III

O funcionamento é o seguinte:

1. Executa os comandos de inicialização;
2. Testa a condição;
3. Se a condição for falsa então executa o comando que está logo após o bloco subordinado ao **for** .
4. Se condição for **verdadeira** então executa os comandos que estão subordinados ao **for**;
5. Executa os comandos de incremento/decremento;
6. Volta ao passo 2.

# Comando for IV

Mostrar os 10 primeiros números naturais

## Comando for V

```
#include <stdio.h>
int main()
{
    int i;
    for (i = 0; i < 10; i++)
        printf("%d \n", i);
    return 0;
}
```

## Comando for VI

Mostrar os 10 primeiros número naturais em ordem decrescente

## Comando for VII

```
#include <stdio.h>
int main()
{
    int i;
    for (i = 9; i >= 0; i--)
        printf("%d \n", i);
    return 0;
}
```

## Comando for VIII

Dado um número inserido por teclado, responder se é primo



# Comando for IX

```
#include <stdio.h>
int main()
{
    int num, cont = 0;;
    printf("Inserire numero \n");;
    scanf("%d", &num);

    for (int i = 2; i <= num; i++)
        if (num % i == 0)
            cont++;

    if (cont == 1)
        printf("Numero primo \n");;
}
```

## Comando for X

Escrever um programa que lê 5 valores, um de cada vez, e conta quantos destes valores são negativos, escrevendo esta informação

## Comando for XI

```
int main(){
    int a, cont = 0, i = 0;
    while (i < 5)
    {
        printf("Inserir numero %d ", i+1);
        scanf("%d", &a);
        if (a < 0)
            cont++;
        i++;
    }
    printf("Total de números negativos: %d", cont);
    return 0;
}
```

## Comando for XII

Escreva um algoritmo que calcule a média dos números digitados pelo usuário, se eles forem pares. Termine a leitura se o usuário digitar zero (0).

## Comando for XIII

```
int main(){
    int cont = 0, num;
    double media, acum = 0;
    do
    {
        printf("Inserir numero");
        scanf("%lf", &num);
        if (num != 0)
        {
            if (num % 2 == 0)
            {
                acum += num;
                cont++;
            }
        }
    } while (num != 0);
    printf("A media eh: %lf", acum/cont);
    return 0;
}
```

FIM