



INSTITUTO FEDERAL
SANTA CATARINA
Campus Lages



VisuALG – Estruturas de Repetição

Professores:

Vilson Heck Junior

vilson.junior@ifsc.edu.br

Felipe Schneider Costa

felipe.costa@ifsc.edu.br



Agenda

- O Problema.
- Estruturas de Repetição:
 - Introdução;
 - Repita – ate;
 - Exemplo;
 - Enquanto – faça;
 - Exemplo;
 - Para – faça;
 - Exemplo;
 - Exercícios.

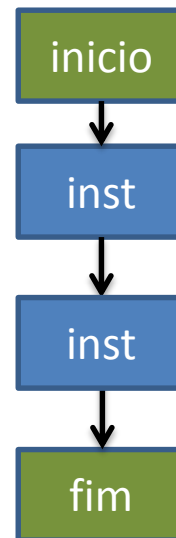


O Problema

- **Exercício:**
 1. Escreva um algoritmo que leia 20 valores inteiros e ao final exiba:
 - a) a soma dos números positivos;
 - b) a quantidade de valores negativos.

O Problema

- Apresentação no Visualg



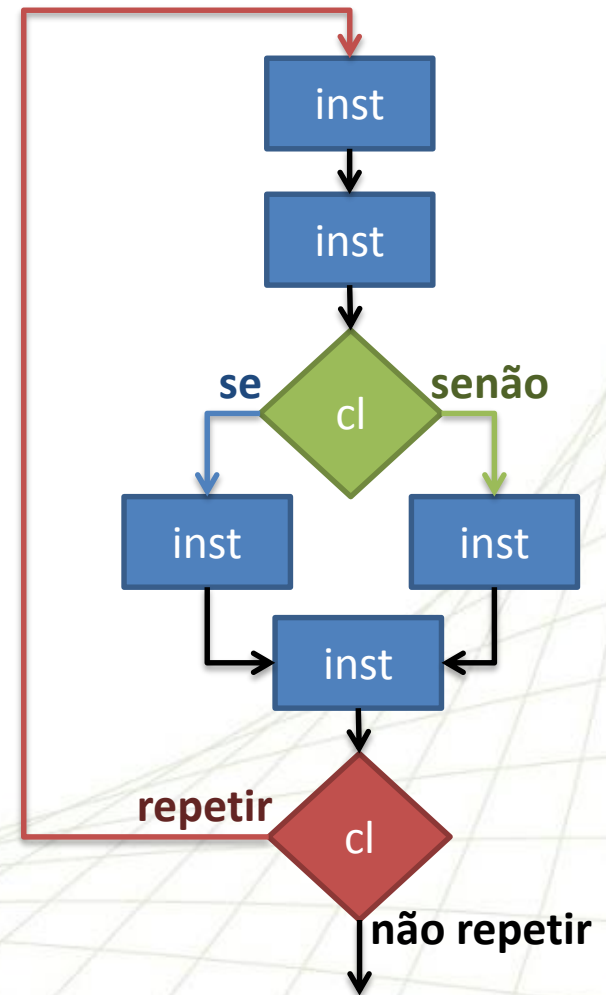
- Legenda:
 - inst = Instrução;
 - cl = Cláusula;

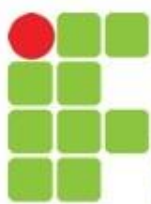
O Problema

- Apresentação no Visualg.

- Legenda:

- inst = Instrução;
- cl = Cláusula;





INSTITUTO FEDERAL
SANTA CATARINA
Campus Lages



VisuALG: Introdução a

ESTRUTURAS DE REPETIÇÃO

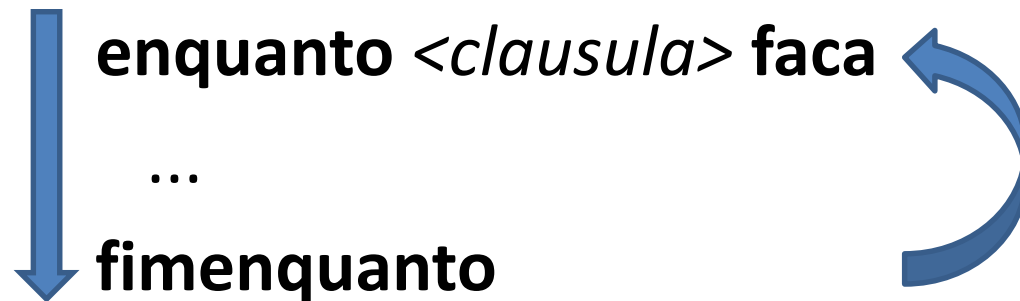
Estruturas de Repetição

- Executar tarefas repetitivas;
- Três principais opções:
 - enquanto *<clausula>* faça
 - fimenquanto
 - repita ... ate *<clausula>*
 - para *<contagem>* faça
 - fimpara

enquanto – faça

- Verifica primeiro, executa depois;
- Repete somente enquanto **<clausula> = verdade;**

enquanto <clausula> faça
...
fimenquanto



Exemplo – Contagem até 10

```
x: inteiro  
x ← 1  
enquanto (x ≤ 10) faça  
    escreval(x)  
    x ← x + 1  
fimenquanto
```

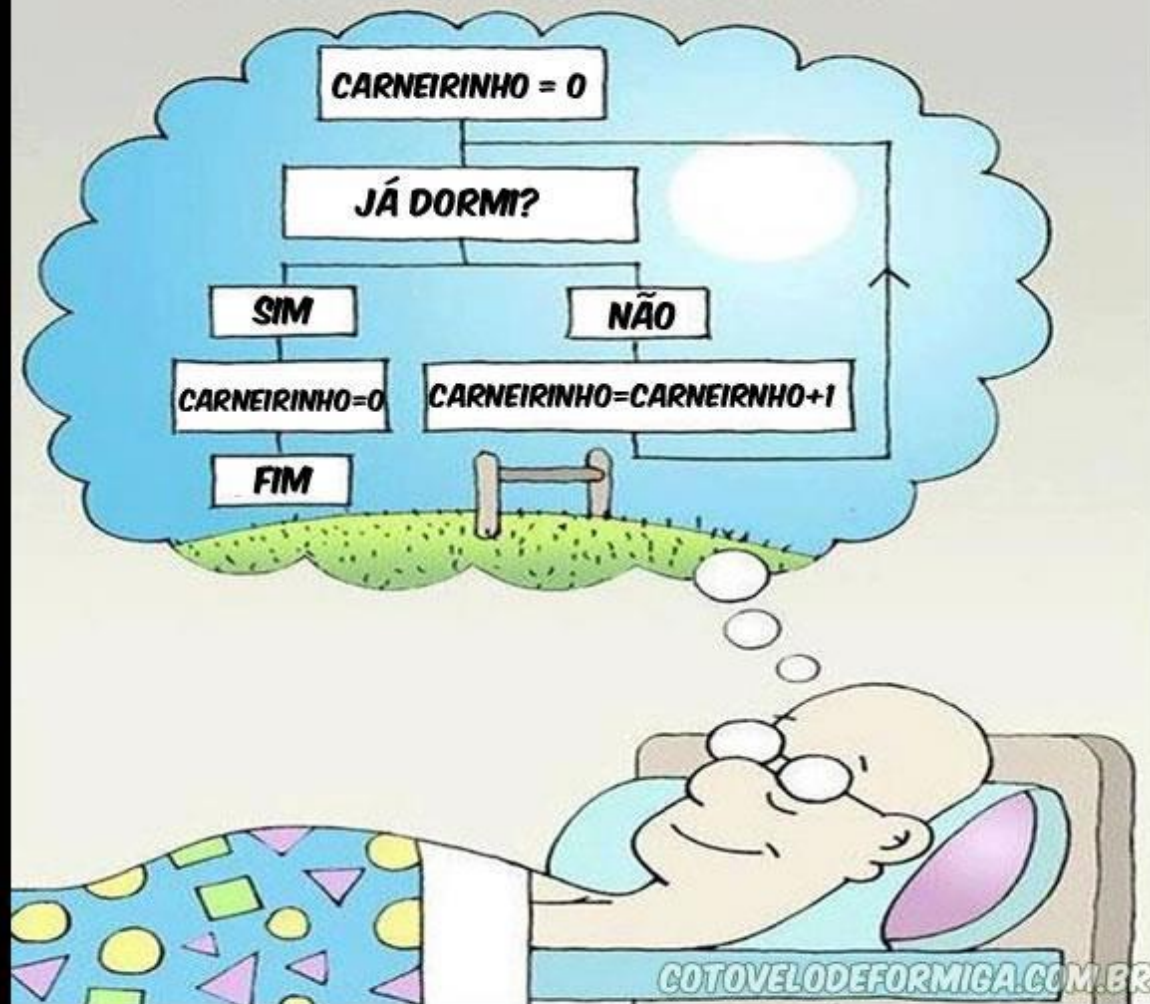

enquanto – faça

- Exemplo:
 nota: **real**
 escreva("Digite uma nota:")
 leia(nota)
 enquanto (nota < 0) **ou** (nota > 10) **faça**
 escreval("Erro! A nota deve ser entre 0 e 10")
 escreva("Digite novamente a nota: ")
 leia(nota)
 fimenquanto
 se (nota >= 7) **entao**
 escreval("Aluno aprovado!")
 senao
 escreval("Aluno reprovado!")
 fimse



INSTITUTO
SANTA CATARINA
Campus

UM PROGRAMADOR COM INSÔNIA



PROGRAMADORES
SÃO ESPECIALISTAS ATÉ NA HORA DE DORMIR

Exercícios - enquanto

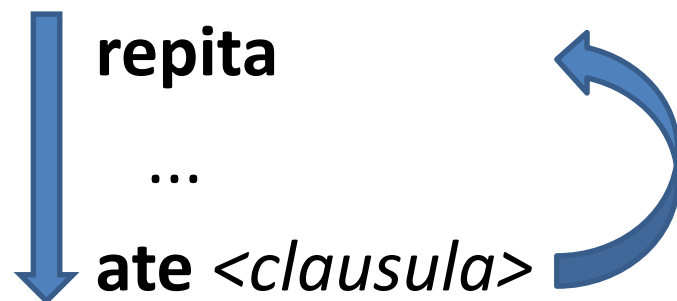
1. Faça um algoritmo que escreva na tela os números de um número inicial a um número final. Os números inicial e final devem ser informados pelo usuário;
2. Escrever um algoritmo que imprima a tabuada de um número informado pelo usuário;
3. Escrever um algoritmo que gera e escreve os números ímpares entre 100 e 200;
4. Em uma turma há 10 alunos. Cada aluno tem 2 notas. Um professor precisa calcular a média das duas notas de cada aluno. Crie um programa que resolve este problema.

Exercícios

1. Escrever um programa de computador que leia 10 números inteiros e, ao final, apresente a soma de todos os números lidos;
2. Faça o mesmo que antes, porém, ao invés de ler 10 números, o programa deverá ler e somar números até que o valor digitado seja zero (0).

repita – ate

- Executa primeiro, verifica depois:
- Repete somente enquanto: **<clausula> = falso;**



Exemplo – Contagem até 10

```
x: inteiro  
x ← 1  
repita  
  escreval(x)  
  x ← x + 1  
ate (x > 10)
```

- Exemplo:

opcao: **inteiro**

repita

escreval("1 - Dizer olá!")

escreval("2 – Dizer oi! ")

escreval("0 - Sair do programa")

leia(opcao)

se (opcao = 1) **entao**

escreval("Olá!")

fimse

se (opcao = 2) **entao**

escreval("Oi!")

fimse

ate (opcao = 0)

Exercícios - repita

1. 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);
2. Escreva um algoritmo que leia valores inteiros e encontre o maior e o menor deles. Termine a leitura se o usuário digitar zero (0);

Exercícios - repita

3. Escreva um programa que lê o sexo de uma pessoa. O sexo deverá ser com o tipo de dado **caractere** e o programa deverá aceitar apenas os valores “M” ou “F”.
4. Escreva um programa que leia dois valores reais. Ambos valores deverão ser lidos até que o usuário digite um número no intervalo de 1 a 100. Apresentar a soma dos dois valores lidos.

para – faça

- Repetição condicionada a uma contagem:

para <variável> **de** <valor inicial> **ate** <valor final> **[passo** <incremento> **]** **faca**
...
fimpara

Diagram illustrating the loop structure with a blue arrow pointing down on the left and a blue bracket labeled "Opcional" above the optional increment part. A blue curved arrow points from the end of the loop back to the start.

Exemplo – Contagem até 10

```
x: inteiro
para x de 1 ate 10 faca
  escreval(x)
fimpara
```

Exemplo – Contagem até 10 c/ passo

```
x: inteiro
para x de 1 ate 10 passo 1 faca
  escreval(x)
fimpara
```

Atenção: a <variável> utilizada nesta estrutura terá seu valor alterado.

para – faça

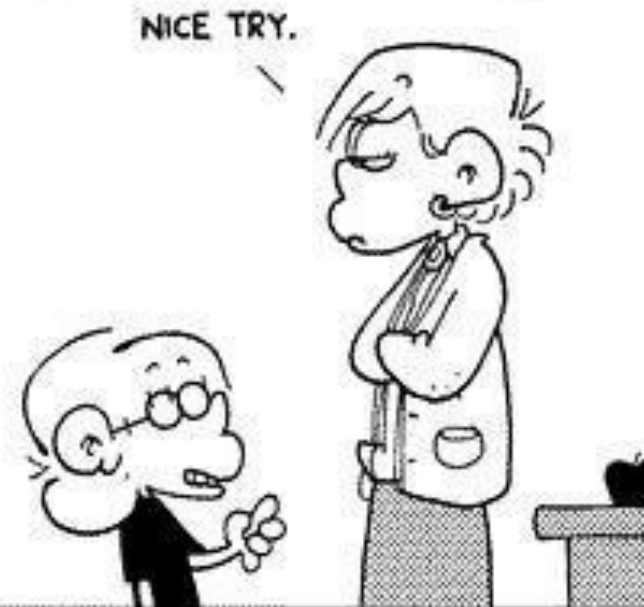
- Exemplo:
soma, num: **real**
i, n: **inteiro**
escreva("Digite o número de elementos: ")
leia(n)
soma <- 0
para i **de** 1 **ate** n **faça**
 escreva("Digite um número: ")
 leia(num)
 soma <- soma + num
fimpara
escreval("Total dos ", n, " elementos: ", soma)



Exemplo Prático

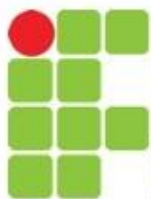
```
#include <stdio.h>
int main(void)
{
    int count;

    for (count = 1; count <= 500; count++)
        printf("I will not throw paper airplanes in class.");
    return 0;
}
```



Exercícios - para

1. Escreva um programa que lido um número, calcule e informe o seu fatorial.
Ex.: $5! = 5 * 4 * 3 * 2 * 1 = 120$.
2. Escreva um programa que leia um valor correspondente ao número de jogadores de um time de vôlei. O programa deverá ler uma altura para cada um dos jogadores e, ao final, informar a altura média do time.



INSTITUTO FEDERAL
SANTA CATARINA
Campus Lages



Estruturas de Repetição

CONDIÇÕES DE PARADA

Condições de Parada

- Se uma estrutura de repetição serve para executar uma determinada tarefa várias vezes, é necessário indicar quando as repetições devem parar:
 - Pode ser feita por um número pré-determinado de vezes;
 - Pode ser feita até que um evento ocorra;
 - Pode ser feita até a solicitação do usuário;

Condições de Parada

- **Para Faca** é mais simples e adequada para uso nos casos de um número pré determinado de repetições;
 - Principalmente em intervalos numéricos, contadores e outros do gênero;
- **Repita Ate()** e **Enquanto () Faca** são mais adequadas para repetir até que “algo” aconteça:
 - Algo pode ser um comando do usuário ou até se obter uma informação ou valor esperado;

Condições de Parada

- Caso a condição de parada para qualquer estrutura de repetição nunca seja satisfeita, teremos problema com uma repetição infinita;
- A repetição infinita irá travar o *software*, que não será capaz de terminar a operação.

Condições de Parada

- Exemplos de loop infinito com **Enquanto Faca**:
cont: inteiro
cont <- 1
enquanto (cont < 10) faca
 escreva(cont)
fimenquanto
- Exemplos de loop infinito com **Para Faca**:
cont: inteiro
para cont de 1 ate 10 passo -1 faca
 escreva(cont)
fimpara

Variáveis de Controle

- Ao estabelecer uma condição de parada, automaticamente precisamos comparar uma ou mais variáveis;
- Temos que cuidar muito do uso dessas variáveis, pois se alterarmos o seu conteúdo de forma indiscriminada, podemos tornar o comportamento da repetição instável.

Variáveis de Controle

- Exemplos de variáveis de controle com Enquanto:

cont: inteiro

cont <- 1

enquanto (cont < 10) faça

 escreva(cont)

 cont <- cont + 1

fimenquanto

- Exemplos de variáveis de controle com Para Faça:

cont: inteiro

para cont de 1 ate 10 faça

 escreva(cont)

fimpara



INSTITUTO FEDERAL
SANTA CATARINA
Campus Lages



Estruturas de Repetição

ACUMULADORES E CONTADORES

Acumuladores e Contadores

- Contagens, somas e multiplicações acumulativas:
 - São utilizadas com frequência em algoritmos;
 - Geralmente associadas à repetições;
 - São representadas por variáveis numéricas;
 - E são incrementadas/alteradas conforme padrões recorrentes ocorrem.

Contadores

- Contagem:
 - É usada para, por exemplo:
 - contabilizar o número de execuções de uma repetição;
 - determinar o número de vezes que um particular valor (ou ação) ocorre em uma determinada sequência;
 - entre outros;
 - A variável associada geralmente inicia no valor zero (elemento neutro da soma);
 - Sempre que conveniente, acrescenta-se 1 ao contador.

5 x 4

Acumuladores e Contadores

- Somas ou Produtos Acumulativos:
 - São freqüentes em cálculos de somatórias, produtórias ou consolidações de resultados (totais, médias, ...);
 - Somatório:
 - Geralmente são inicializadas no valor zero (elemento neutro da soma);
 - Sempre que apropriado, soma-se outros valores à própria variável;
 - Produtório:
 - São inicializadas usualmente com o valor um (elemento neutro da multiplicação);
 - Sempre quando necessário, são atualizadas com o resultado do seu valor corrente multiplicado por um novo termo.



INSTITUTO FEDERAL
SANTA CATARINA
Campus Lages



Estruturas de Repetição

EXEMPLOS

Exemplo (repita – ate)

- Exibir a tabuada de um número:
 1. x, y : inteiro
 2. escreva("Informe o número da tabuada: ")
 3. leia(x)
 4. $y \leftarrow 0$
 5. repita
 6. escreval(x , "x", y , "=", $x * y$)
 7. $y \leftarrow y + 1$
 8. ate($y > 10$)



Contador

Exemplo (enquanto – faça)

- Capturar números inteiros do usuário até que ele digite 0 (zero);
- Informar a soma dos números e a média;

1. dig, soma, qtdade: **inteiro**

2. media: **real**

3. soma <- 0

4. qtdade <- 0

5. **escreva**("Informe um número(0 para sair): ")

6. **leia**(dig)

7. **enquanto** (dig <> 0) **faca**

8. soma <- soma + dig

9. qtdade <- qtdade + 1

10. **escreva**("Informe um número(0 para sair): ")

11. **leia**(dig)

12. **fimenquanto**

13. media <- soma / qtdade

14. **escreva**("Qtdade: ", qtdade, " Soma: ", soma, " Média: ", media)

Somatório

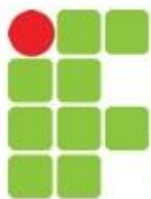
Contador

Exemplo (para – faca)

- Multiplicar dois números apenas com somas:

1. x, y, z , multi: **inteiro**
2. **escreva**("Informe x: ")
3. **leia**(x)
4. **escreva**("Informe y: ")
5. **leia**(y)
6. $\text{multi} \leftarrow 0$
7. **para** z **de** y **ate** 1 **passo** -1 **faca**
8. $\text{multi} \leftarrow \text{multi} + x$
9. **fimpara**
10. **escreva**("Multiplicação = ", multi)

Somatório



INSTITUTO FEDERAL
SANTA CATARINA
Campus Lages



Estruturas de Repetição

EXERCÍCIOS

Exercícios - Repetição

1. Em um concurso de *miss IFSC – Lages*, os jurados precisam digitar o nome das 16 candidatas e suas respectivas notas (0 a 10). Crie um programa que leia estas informações e que, ao final do programa, apresente apenas o nome e a nota da vencedora.
2. Uma loja deseja fazer a avaliação com base na opinião de seus clientes. Para isto, ela irá disponibilizar um computador que irá perguntar a cada cliente a sua **idade** e a **nota** que ele avalia a loja. Você deverá elaborar o programa de computador que fará a leitura da opinião dos 100 primeiros clientes e deverá informar os dados solicitados abaixo:

NOTAS	INFORMAÇÕES
A. Ótimo	1. A quantidade de respostas A (Ótimo);
B. Bom	2. A média de idade das pessoas que responderam D (Ruim);
C. Regular	3. A percentagem de respostas E (Péssimo) e a menor idade de quem informou esta resposta;
D. Ruim	
E. Péssimo	4. A maior idade de quem respondeu A (Ótimo) e a maior idade de quem respondeu D (Ruim);



INSTITUTO FEDERAL
SANTA CATARINA
Campus Lages



Estruturas de Repetição

ANINHAMENTO

Aninhamento

- Algumas vezes precisamos trabalhar com conjuntos e sub-conjuntos de dados;
 - Neste caso precisamos de uma estrutura de repetição para percorrer todos os conjuntos;
 - Mas para cada conjunto, precisamos percorrer um sub-conjunto de dados;
 - Isso chamamos de aninhamento de estruturas de repetição.

Aninhamento

- Imagine um problema:
 - O usuário precisa somar Notas Fiscais (NF);
 - Ele possui um número indeterminado de notas fiscais;
 - Cada NF possui um número indeterminado de itens;
 - Cada item possui um valor que o usuário irá digitar;
 - Ao terminar de digitar cada nota, deverá ser exibido o valor total de soma dos itens da NF digitada;
 - Ao terminar de digitar TODAS as NFs, deverá ser exibido o valor total de soma de todas as NFs



NOME

Endereço completo

CNPJ

Inscrição Estadual

MEI - dispensado

Destinatário

End.	CNPJ/CPF
------	----------

NOTA FISCAL DE VENDA A CONSUMIDOR – MOD 2 Série D

Data de emissão/...../.....

n°

NNN

Nome, endereço, inscrição estadual e CNPJ do impressor da nota, data e quantidade de impressão, número de ordem da primeira e última nota impressa e respectiva série e número da Autorização de Impressão de Documentos Fiscais-AIDF

Aninhamento

As variáveis de controle são DIFERENTES!

somaNF, somaTotal, val: real
nfa, nnf, ia, ni: inteiro

```
escreva("Digite o número de NFs a serem somadas: ")  
leia(nnf)  
somaTotal <- 0
```

```
para nfa de 1 ate nnf faca  
  escreval("===== Nota Fiscal ", nfa, " =====")  
  escreva("Digite o número de itens da NF", nfa, ": ")  
  leia(ni)  
  somaNF <- 0
```

Loop
externo

```
    para ia de 1 ate ni faca  
      escreva("Digite o valor do item ", ia, ": ")  
      leia(val)  
      somaNF <- somaNF + val  
    fimpara
```

Loop
interno

```
  somaTotal <- somaTotal + somaNF  
  escreval("Soma da NF", nfa, ": ", somaNF)  
fimpara  
escreval("=====")  
escreval("Soma de Todas as NFs: ", somaTotal)
```



INSTITUTO FEDERAL
SANTA CATARINA
Campus Lages



Estruturas de Repetição Aninhadas

EXERCÍCIOS

Exercícios – Repetição

Implemente algoritmos que:

1. Informe se um número digitado é primo. Caso não for, informe por quais números ele é divisível;
2. Digitados dois números (base e expoente – b^e), calcule o resultado utilizando apenas multiplicações;

Exercícios – Repetição

(Repetição Aninhada)

3. Calcule a soma de todos os números primos existentes entre 1 e 100;
4. Faça o mesmo que em 2, mas usando apenas somas;