

ALGORITMOS E LÓGICA DE PROGRAMAÇÃO

Estruturas de Controle

Profa. Dra. Jaqueline Brigladori Pugliesi

1

Estrutura Repetição

2

Estruturas de Repetição

- São aquelas que permitem executar mais de uma vez (repetir) um determinado trecho do algoritmo
- O trecho do algoritmo em repetição é também chamado de laço (loop)
- As repetições devem ser sempre finitas
- Quanto a quantidade de repetições, os laços podem ser:
 - Pré-determinados: sabe-se antes a quantidade de execuções
 - Indeterminados: não se conhece a quantidade de execuções
- Quanto ao critério de parada, os laços podem utilizar:
 - Teste no início
 - Teste no final
 - Variável de controle

3

Repetição com Teste no Início

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

```
enquanto (<expressão-lógica>)  
{  
    <sequência-de-comandos>  
}
```

4

Repetição com Teste no Início

- Contador: variável que reproduz o processo de contagem

início	
inteiro: CON;	
CON ← 0;	
enquanto CON < 3 faça	
CON ← CON + 1;	
fimenquanto;	
fim.	

CON
3

5

Média Aritmética de cada um dos 50 alunos com aprovação ou reprovação

```
programa
{
    funcao inicio()
    {
        real N1, N2, N3, N4, media
        inteiro cont

        cont = 0
        enquanto (cont < 50)
        {
            escreva("Notas: \n")
            leia(N1, N2, N3, N4)
            media = (N1 + N2 + N3 + N4) / 4
            escreva("A média é: ", media, "\n")
            se (media >= 7)
            {
                escreva("Aluno Aprovado. Parabéns!\n")
            }
            senao
            {
                escreva("Aluno Reprovado. Estude mais!\n")
            }

            cont = cont + 1
        }
    }
}
```

6

Repetição com Teste no Início

E se fosse preciso calcular a média
geral da turma?

$$\frac{M1 + M2 + M3 + M4 + M5 + \dots + M49 + M50}{50}$$

Isto se torna inviável!

7

Repetição com Teste no Início

- Acumulador: variável que reproduz o processo de acumulação

início
inteiro: CON, X, ACM;
CON ← 0;
ACM ← 0;
enquanto CON < 3 faça
CON ← CON + 1;
leia (X);
ACM ← ACM + X;
fimenquanto ;
fim.

CON	ACM	X
3	11	4

8

acumular X contar

- O processo de acumulação é muito similar ao processo de contagem.
- A única diferença é que na acumulação o valor adicionado pode variar, enquanto na contagem o valor adicionado é constante (geralmente sempre 1).

9

Média Aritmética das notas da turma de 50 alunos

```
programa
{
    funcao inicio()
    {
        real media, acm, MediaTurma
        inteiro cont

        cont = 0
        acm = 0.0
        enquanto (cont < 50)
        {
            escreva("Digite a média do aluno: ")
            leia(media)
            acm = acm + media
            cont = cont + 1
        }
        MediaTurma = acm/50
        escreva("Média da turma = ", MediaTurma)
    }
}
```

10

Exercícios

- Fazer um algoritmo que imprima a tabuada de um número fornecido pelo usuário, utilizando **Teste no Início**.

11

Repetições indeterminadas

- Conhecer antecipadamente a quantidade de alunos da turma da qual se desejava a média geral, permitiu construir um laço de repetição com quantidade pré-determinada de execuções.
- E se não soubéssemos quantos eram os alunos, o que faríamos para controlar o laço de repetição?
- Precisaríamos de um laço que fosse executado por uma quantidade indeterminada de vezes e teríamos que encontrar um outro critério de parada, que possibilitasse que o laço fosse finalizado após a última média anual ter sido informada, independente de quantas sejam.

12

Repetições indeterminadas

```
...  
// teste da nova condição de parada  
enquanto (MA <> -1)  
{  
  leia (MA)  
  se (MA <> -1)  
  {  
    ...  
    ..  
  }  
}  
}
```

```
...  
// teste da nova condição de parada  
leia (MA)  
enquanto (MA <> -1)  
{  
  ...  
  ...  
  leia (MA)  
}
```

13

Média da turma com -1 como valor de finalização

```
programa  
{  
  funcao inicio()  
  {  
    real media, acm, MediaTurma  
    inteiro cont  
    cont = 0  
    acm = 0.0  
    escreva("Digite a média do aluno: ")  
    leia(media)  
    enquanto (media != -1)  
    {  
      acm = acm + media  
      cont = cont + 1  
      escreva("Digite a média do aluno: ")  
      leia(media)  
    }  
    se (cont == 0)  
    {  
      escreva("Nenhum aluno inscrito")  
    }  
    senao  
    {  
      MediaTurma = acm/cont  
      escreva("Média da turma = ", MediaTurma)  
    }  
  }  
}
```

14

Exercício de fixação I

- Construa um algoritmo que calcule a média aritmética de um conjunto de números pares fornecidos pelo usuário. O usuário irá fornecer um total de 10 números.
Observe que nada impede que o usuário forneça quantos números ímpares quiser, com a ressalva de que eles não poderão ser acumulados.

15

Exercício de fixação 2

- Construa um algoritmo que calcule a média aritmética de um conjunto de números pares que forem fornecidos pelo usuário. O valor de finalização será a entrada do número 0 (zero).
Observe que nada impede que o usuário forneça quantos números ímpares quiser, com a ressalva de que eles não poderão ser acumulados.

16

Repetição com Teste no Final

- Laço que verifica depois de cada execução, se é “permitido” continuar executando o trecho do algoritmo

```
faca  
{  
    <sequência-de-comandos>  
} enquanto (<expressão-lógica>)
```

17

Repetição com Teste no Final

- Observe que toda a sequência de comandos será executada pelo menos uma vez, independentemente da validade da condição.
- Isto ocorre porque a inspeção da condição é feita após a execução do bloco, o que representa a característica principal desse modelo de repetição.

18

Média Aritmética da turma com Teste no Final

```
programa
{
    funcao inicio()
    {
        real media, acm, MediaTurma
        inteiro cont

        cont = 0
        acm = 0.0
        faca
        {
            escreva("Digite a média do aluno: ")
            leia(media)
            acm = acm + media
            cont = cont + 1
        } enquanto (cont < 50)

        MediaTurma = acm/50
        escreva("Média da turma = ", MediaTurma)
    }
}
```

19

Exemplo

```
programa
{
    funcao inicio()
    {
        inteiro idade

        faca
        {
            escreva("Informe sua idade (valores
aceitos de 5 a 130): ")
            leia(idade)
        } enquanto (idade < 5 ou idade > 130)

        // A partir daqui a idade está correta.

        escreva("\nCorreto!\n")
    }
}
```

20

Exercício

- Faça um algoritmo que para imprimir a tabuada de um número fornecido pelo usuário, utilizando **Teste no Final**.

21

Exercício de fixação

- Imagine uma brincadeira entre dois colegas, na qual um pensa um número e o outro deve fazer chutes até acertar o número imaginado. Como dica, a cada tentativa é dito se o chute foi alto ou foi baixo. Elabore um algoritmo dentro deste contexto, que leia o número imaginado e os chutes, ao final mostre quantas tentativas foram necessárias para descobrir o número.
Observe que a estrutura de repetição não possui um número determinado de iterações, pois o laço continuará sendo executado até que o usuário acerte o número pensado.
O laço é executado pelo menos uma vez, e se for o caso o usuário teve bastante sorte e acertou o número na primeira tentativa.

22

Repetição com Variável de Controle

- Laço simplificado para utilização em repetições de quantidade determinada
- Incorpora internamente o funcionamento de um contador de repetições

```
para (VInicial; Condição; Incremento)
{
    <sequência-de-comandos>
}
```

23

Média Aritmética da turma com Para

```
programa
{
    funcao inicio()
    {
        real media, acm, MediaTurma
        inteiro cont
        acm = 0.0
        para (cont = 0; cont < 50; cont++)
        {
            escreva("Digite a média do aluno: ")
            leia(media)
            acm = acm + media
        }
        MediaTurma = acm/50
        escreva("Média da turma = ", MediaTurma)
    }
}
```

24

Exercício

- Faça um algoritmo que para imprimir a tabuada de um número fornecido pelo usuário, utilizando **para**.

25

Exercício de fixação

- Faça um algoritmo que simule uma contagem regressiva de 10 minutos, ou seja, mostre: 10:00, 9:59, 9:58, 9:57, ..., 8:59, 8:58, até 0:00.

26

Comparação entre Estruturas de Repetição

- Aprendemos 3 maneiras de construir laços de repetição:
 - Teste no início
 - Teste no final
 - Variável de controle
- É importante perceber que existem laços mais adequados ou convenientes para cada situação

27

Exercícios gerais

1. Faça um algoritmo que permita fazer um levantamento do estoque de vinhos de uma adega, tendo como dados de entrada tipos de vinho, sendo: 'T' para tinto, 'B' para branco e 'R' para rosê. Especifique a porcentagem de cada tipo sobre o total geral de vinhos; a quantidade de vinhos é desconhecida, utilize para finalizador do algoritmo a opção 'F' de fim.
2. Faça um algoritmo que efetue a soma de todos os números ímpares que são múltiplos de 3 e que se encontram no conjunto dos números de 1 até 500.

28

Exercícios gerais

3. Faça um algoritmo que utilize as três estruturas de repetição para imprimir a tabuada do número 5.
4. Modifique o algoritmo para que ele imprima a tabuada de quaisquer números, sendo que esses são fornecidos pelo usuário, até encontrar como finalizador o número -1.

29

FIM

30