

Estruturas de Controle - estrutura de repetição -

**Algoritmos I
2014/1**

Profa: Daniela Scherer dos Santos
daniela.santos37@ulbra.edu.br



Estruturas de Controle

- ♦ A estrutura de controle de um algoritmo indica o fluxo de execução que deverá ser seguido pelo algoritmo para que ele processe a saída desejada;
- ♦ Estruturas básicas de controle:
 - ♦ sequencial;
 - ♦ de seleção (condicional);
 - ♦ de repetição.

Estruturas de Controle

- ♦ A estrutura de controle de um algoritmo indica o fluxo de execução que deverá ser seguido pelo algoritmo para que ele processe a saída desejada;
- ♦ Estruturas básicas de controle:
 - ♦ ~~sequencial;~~
 - ♦ ~~de seleção (condicional);~~
 - ♦ de repetição.

Exercício exemplo:

1) Faça um algoritmo que calcule as médias de 20 alunos de uma turma, considerando que cada aluno tem duas notas e que estará aprovado aquele aluno cuja média aritmética das notas for igual ou superior a 7. Utilizar como dados de entrada o nome do aluno e suas notas. O algoritmo deverá fornecer como resultado o nome do aluno, sua média e a informação “Aprovado” ou “Reprovado”.

Estrutura de Repetição

- uma estrutura de repetição permite a repetição de uma instrução, ou bloco de instruções, por um número determinado ou indeterminado de vezes.

Estrutura de Repetição

- uma estrutura de repetição permite a repetição de uma instrução, ou bloco de instruções, por um número **determinado** ou indeterminado de vezes.

quando se conhece a quantidade de repetições necessárias

Estrutura de Repetição

- uma estrutura de repetição permite a repetição de uma instrução, ou bloco de instruções, por um número determinado ou **indeterminado** de vezes.

quando NÃO se conhece a quantidade de repetições necessárias

Estrutura de Repetição

- ♦ Repetição determinada:
 - ♦ Comando para;
- ♦ Repetição indeterminada:
 - ♦ Comando enquanto;
 - ♦ Comando repita;

Estrutura de Repetição

- ♦ Repetição determinada:
 - ♦ Comando para;
- ♦ Repetição indeterminada:
 - ♦ Comando enquanto;
 - ♦ Comando repita;

Comando *para*

- ♦ A estrutura de repetição com comando *para* repete a execução de um bloco de comandos um número DEFINIDO de vezes, pois ela possui limites fixos.
- ♦ SINTAXE:

```
para varDeControle = valorInicial até valorFinal faça  
inicio  
    <bloco de comandos>;  
fim
```

Comando *para*

```
para varDeControle = valorInicial até valorFinal faça  
início  
    <bloco de comandos>;  
fim
```

- ♦ *para*, *até* e *faça* → são palavras reservadas do comando;

Comando *para*

```
para varDeControle = valorInicial até valorFinal faça  
início  
    <bloco de comandos>;  
fim
```

- ♦ *para*, *até* e *faça* → são palavras reservadas do comando;
- ♦ *varDeControle* → é uma variável ou expressão pertencente ao tipo escalar (exceto real);

Comando *para*

```
para varDeControle = valorInicial até valorFinal faça  
início  
    <bloco de comandos>;  
fim
```

- ♦ *para*, *até* e *faça* → são palavras reservadas do comando;
- ♦ *varDeControle* → é uma variável ou expressão pertencente ao tipo escalar (exceto real);
- ♦ *valorInicial* e *valorFinal* → indicam, respectivamente, os valores inicial e final da variável *varDeControle* (delimitam a repetição do bloco de comandos).

Comando *para*

Exercícios Exemplo: Faça um algoritmo que calcule a média de 20 alunos de uma turma, considerando que cada aluno tem duas notas e que estará aprovado aquele aluno cuja média aritmética das notas for igual ou superior a 7. Utilizar como dados de entrada o nome do aluno e suas notas. O algoritmo deverá fornecer como resultado o nome do aluno, sua média e a informação “Aprovado” ou “Reprovado”.

Comando *para*

```
algoritmo notaFinal
var
    caracter nome;
    real nota1, nota2, media;
    inteiro x;
inicio
    para x=1 até 20 faça
        inicio
            leia(nome);
            leia(nota1,nota2);
            media = (nota1+nota2)/2;
            se (media >= 7) então
                inicio
                    (nome, “está aprovado com média”, media);
                fim
            senão
                inicio
                    (nome, “está reprovado e sua média é”, media);
                fim
        fim
    fim
fim
```

Comando *para*

```
algoritmo notaFinal
```

```
var
```

```
  caracter nome;
```

```
  real nota1, nota2, media;
```

```
  inteiro x;
```

```
inicio
```

```
  para x=1 até 20 faça
```

```
  inicio
```

```
    leia(nome);
```

```
    leia(nota1,nota2);
```

```
    media = (nota1+nota2)/2;
```

```
    se (media >= 7) então
```

```
    inicio
```

```
      (nome, "está aprovado com média", media);
```

```
    fim
```

```
  senão
```

```
  inicio
```

```
    (nome, "está reprovado e sua média é", media);
```

```
  fim
```

```
  fim
```

```
fim
```

x → variável de controle

x inicia com o valor 1 e terá seu valor incrementado em uma unidade até que ultrapasse o valor limite estabelecido em 20

Comando *para*

```
algoritmo notaFinal
var
    caracter nome;
    real nota1, nota2, media;
    inteiro x;
inicio
    para x=1 até 20 faça
        inicio
            leia(nome);
            leia(nota1,nota2);
            media = (nota1+nota2)/2;
            se (media >= 7) então
                inicio
                    (nome, "está aprovado com média", media);
                fim
            senão
                inicio
                    (nome, "está reprovado e sua média é", media);
                fim
        fim
    fim
fim
```

a cada novo valor da variável de controle x, será executado o bloco de instruções delimitado por **inicio** e **fim**, onde são lidos os dados para cada aluno e determinado o seu resultado.

Comando *para*

```
algoritmo notaFinal
var
    caracter nome;
    real nota1, nota2, media;
    inteiro x;
inicio
    para x=1 até 20 faça
        inicio
            leia(nome);
            leia(nota1,nota2);
            media = (nota1+nota2)/2;
            se (media >= 7) então
                inicio
                    (nome, “está aprovado com média”, media);
                fim
            senão
                inicio
                    (nome, “está reprovado e sua média é”, media);
                fim
        fim
    fim
fim
```

a cada novo valor da variável de controle x, será executado o bloco de instruções delimitado por **inicio** e **fim**, onde são lidos os dados para cada aluno e determinado o seu resultado.

a repetição dos comandos será executada até que a variável x atinja o valor 21, que ultrapassa o limite superior estipulado para x.

Comando *para*

- o acréscimo dado a variável de controle do comando *para* é denominado PASSO. Caso o problema exija, podemos determinar um acréscimo de controle maior do que 1 (um).

```
para varDeControle = valorInicial até valorFinal passo n faça  
inicio  
    <bloco de comandos>;  
fim
```

Comando *para*

```
para varDeControle = valorInicial até valorFinal passo n faça  
início  
    <bloco de comandos>;  
fim
```

- ♦ *passo* → palavra reservada do comando
- ♦ *n* → valor referente ao acréscimo que deverá ser atribuído na variável de controle. Por exemplo: se *n* for 2 então a variável de controle será incrementada de 2 em 2.

Comando *para*

- ◆ Atenção!
- ◆ a variável de controle não deve ser alterada dentro do bloco de comandos. Sua alteração (incremento ou decremento) é efetuada e controlada EXCLUSIVAMENTE pelo comando *para* de forma automática, como se fosse um contador.

Referências Utilizadas

FORBELLONE, A. L. V.; EBERSPÄCHER, H. F.. Lógica de Programação: a construção de algoritmos e estruturas de dados. São Paulo: Makron Books, 2000.

LUX, B.. Notas de Aula.2009.