

# ESTRUTURAS COMPOSTAS

## Variáveis Compostas Unidimensionais VETOR

Introdução à Ciência da  
Computação

# ESTRUTURAS COMPOSTAS

---

- Pode-se organizar os dados dos tipos simples em tipos mais complexos formando-se as ESTRUTURAS COMPOSTAS
- Exemplo:
  - variáveis compostas unidimensionais (VETOR)

# VETOR

---

- O conceito de VETOR será introduzido através de um exemplo.
- Suponhamos o seguinte problema:

Calcular a média aritmética das notas de 3 alunos.

Exibir a média e as notas que estão abaixo da média

Calcular a média aritmética das notas de 3 alunos.  
Exibir a média e as notas que estão abaixo da média

---

## 1ª Solução (PÉSSIMA)

- Ler as três notas uma primeira vez para calcular a média.
- Ler novamente cada nota para comparar com a média e verificar quais notas estão abaixo da média

programa PESSIMO

declarações

inicio

S ← 0

para I de 1 até 3

faça início

ler NOTA

    S ← S + NOTA

fim

fim para

MEDIA ← S/3

escrever (MEDIA)

para I de 1 até 3

faça início

ler NOTA

se NOTA < MEDIA

então escrever (NOTA, "abaixo da média")

fim-se

fim

fim para

fim

Ler as notas uma primeira  
vez para calcular a média

Ler novamente cada nota para  
comparar com a média e  
verificar quais notas estão  
abaixo da média

programa PESSIMO

declarações

início

S ← 0

para I de 1 até 3

faça início

ler **NOTA**

S ← S + NOTA

fim

fim para

MEDIA ← S/3

escrever (MEDIA)

para I de 1 até 3

faça início

ler **NOTA**

se NOTA < MEDIA

então escrever (NOTA, “abaixo da média”)

fim-se

fim

fim para

fim

**INEFICIENTE**

Calcular a média aritmética das notas de 3 alunos.  
Exibir a média e as notas que estão abaixo da média

---

## 2ª Solução (RUIM)

- Ler as três notas e armazenar na memória do computador, dando um nome diferente para cada nota.

programa RUIM  
declarações  
inicio

ler (NOTA1,NOTA2,NOTA3)  
MEDIA  $\leftarrow$  (NOTA1 + NOTA2 + NOTA3)/3  
escrever (MEDIA)  
se NOTA1 < MEDIA  
    então escrever (NOTA1,"abaixo da média")  
fim-se  
se NOTA2 < MEDIA  
    então escrever (NOTA2,"abaixo da média")  
fim-se  
se NOTA3 < MEDIA  
    então escrever (NOTA3,"abaixo da média")  
fim-se

fim

o programa só  
vale para três notas

**IMPRATICÁVEL!**

Qual seria o algoritmo para uma  
relação de **1000** notas?  
Associarmos um nome para cada nota?



Calcular a média aritmética das notas de 1000 alunos.  
Exibir a média e as notas que estão abaixo da média

## 3ª Solução (ABORDAGEM MAIS REALÍSTICA)

- Associar o nome NOTA ao CONJUNTO  
ORDENADO de notas
- Como em C o indexador inicia em 0

NOTA = {N0, N1, ... N999}

1ª

2ª

1000ª

Calcular a média aritmética das notas de 1000 alunos.  
Exibir a média e as notas que estão abaixo da média

### 3ª Solução (ABORDAGEM MAIS REALÍSTICA)

NOTA = {N0, N1, ... N999}  
          1ª    2ª        1000ª

- para fazer referência ou selecionar uma nota específica uso um índice
- Exemplo:
  - a 3ª nota é indicada por NOTA[2]
  - a 1000ª nota é indicada por NOTA[999]
  - uma kª nota é indicada por NOTA[k-1]

Calcular a média aritmética das notas de 1000 alunos.  
Exibir a média e as notas que estão abaixo da média

### 3ª Solução (ABORDAGEM MAIS REALÍSTICA)

- Chamada VARIÁVEL INDEXADA
- Junta-se o nome dado ao conjunto ordenado com um índice

#### Exemplo:

- a 3ª nota é indicada por `NOTA[3]`
- a 1000ª nota é indicada por `NOTA[1000]`
- uma  $k^a$  nota é indicada por `NOTA[k]`

NOME

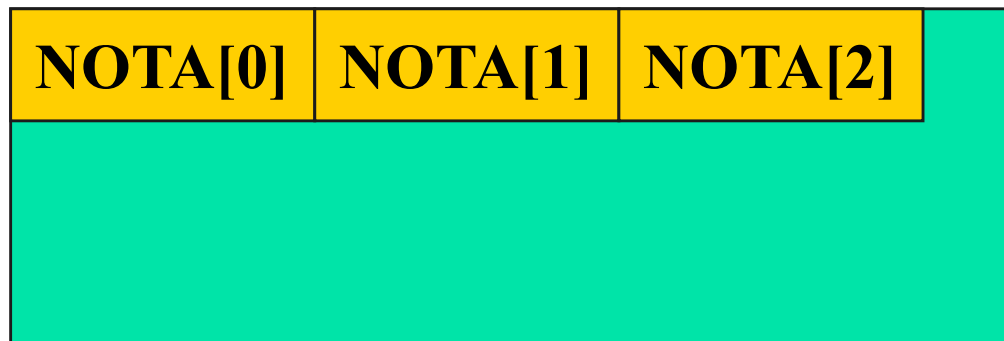
ÍNDICE

NOTA[k]

# VARIÁVEL INDEXADA

---

- Cada **variável indexada** é associada à uma posição de memória, como acontece com variáveis simples.
- Exemplo:



# VETOR

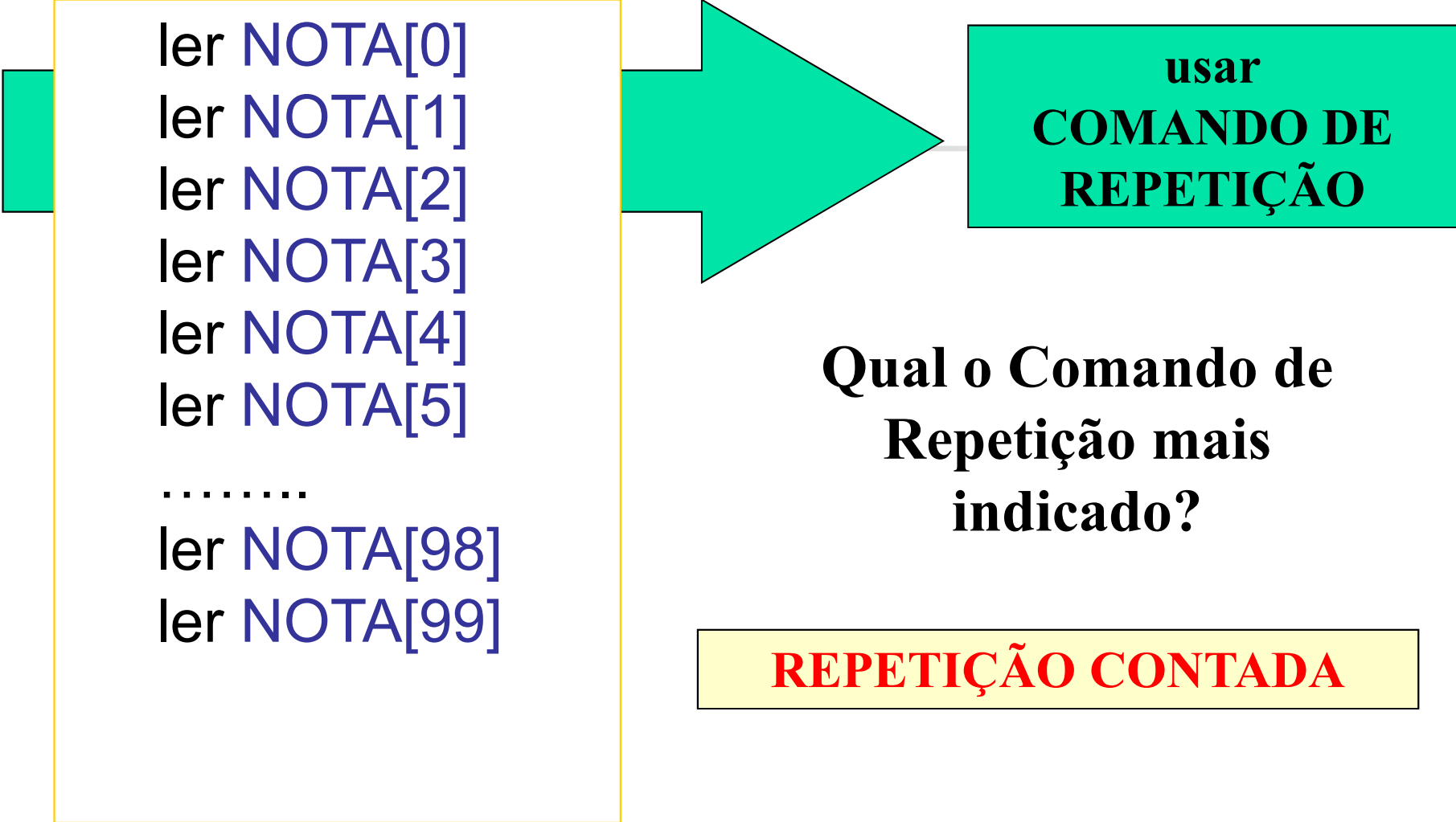
---

- Um **VETOR** é um conjunto ordenado que contém um número fixo de elementos
- Todos os elementos do vetor devem ser do mesmo tipo

# VETOR - Exemplo 1

---

- Ler um conjunto de 100 notas, armazená-las no vetor denominado NOTA e escrever este vetor.
- Esse exemplo também pode ser feito com subrotinas

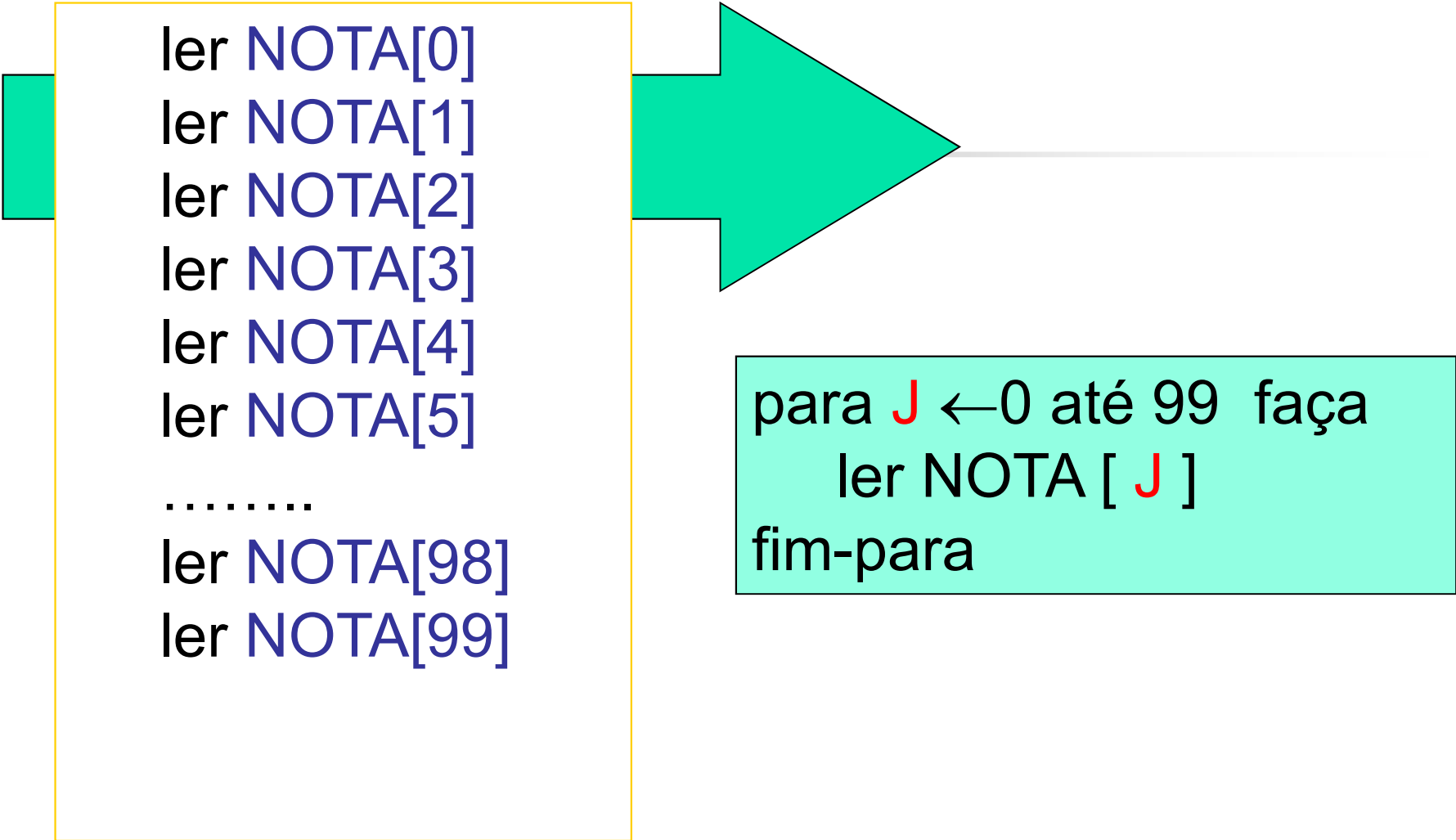


```
ler NOTA[0]
ler NOTA[1]
ler NOTA[2]
ler NOTA[3]
ler NOTA[4]
ler NOTA[5]
.....
ler NOTA[98]
ler NOTA[99]
```

**usar  
COMANDO DE  
REPETIÇÃO**

**Qual o Comando de  
Repetição mais  
indicado?**

**REPETIÇÃO CONTADA**



```
ler NOTA[0]
ler NOTA[1]
ler NOTA[2]
ler NOTA[3]
ler NOTA[4]
ler NOTA[5]
.....
ler NOTA[98]
ler NOTA[99]
```

```
para J ← 0 até 99 faça
    ler NOTA [ J ]
fim-para
```



ler NOTA[0]  
ler NOTA[1]  
ler NOTA[2]  
ler NOTA[3]  
ler NOTA[4]  
ler NOTA[5]  
.....  
ler NOTA[98]  
ler NOTA[99]

para **J** ← 0 até 99 faça  
    ler NOTA [ **J** ]  
fim-para

**mesmo efeito que**

para **I** ← 0 até 99 faça  
    ler NOTA [ **I** ]  
fim-para

**mesmo efeito que**

para **K** ← 0 até 99 faça  
    ler NOTA [ **K** ]  
fim-para

**I, J e K**  
são apenas  
índices que  
assumem um  
valor e que  
junto como  
nome dado ao  
conjunto  
formam a  
variável  
indexada  
NOTA

```
para J ← 0 até 99 faça  
  ler NOTA [ J ]  
fim-para
```

**mesmo efeito que**

```
para I ← 0 até 99 faça  
  ler NOTA [ I ]  
fim-para
```

**mesmo efeito que**

```
para K ← 0 até 99 faça  
  ler NOTA [ K ]  
fim-para
```

```
programa VET1  
declarações  
inicio  
    para J de 0 até 99  
        faça ler NOTA[J]  
    fim para  
    para J de 0 até 99  
        faça escrever NOTA[J]  
    fim para  
fim-programa
```

programa VET1  
declarações  
inicio

para **J** de 0 até 9  
    faça ler NOTA[**J**]  
fim para

para J de 0 até 99  
    faça escrever NOTA[J]  
fim para

fim-programa

Leitura das notas

programa VET1

declarações

inicio

para J de 0 até 99

faça ler NOTA[J]

fim para

para J de 0 até 99

faça escrever NOTA[J]

fim para

fim-programa



Escrita das notas

programa VET1  
declarações  
inicio

para J de 0 até 99  
    faça ler NOTA[J]  
fim para

para J de 0 até 99  
    faça escrever NOTA[J]  
fim para

fim-programa

**Importante!**

Não usar o mesmo laço!  
Isto diminui a legibilidade

## VETOR - Exemplo 2

---

- Ler um conjunto de 100 notas, armazená-las no vetor denominado NOTA, calcular a soma dessas notas, escrever o vetor de notas e a soma das notas.

## VETOR - Exemplo 2

programa VET2

declarações

inicio

para **I** de 0 até 99

faça ler NOTA[**I**]

fim para

**SOMA**  $\leftarrow$  0

para **I** de 0 até 99

faça **SOMA**  $\leftarrow$  **SOMA** + **NOTA**[**I**]

fim para

para **I** de 0 até 99

faça escrever **NOTA**[**I**]

fim para

escrever (**SOMA**)

fim-programa



## VETOR - Exemplo 2

programa VET2

declarações

inicio

para **I** de 0 até 99  
    faça ler NOTA[**I**]  
fim para

Leitura das notas

**SOMA**  $\leftarrow$  0

Soma das notas

para **I** de 0 até 99  
    faça **SOMA**  $\leftarrow$  **SOMA** + NOTA[**I**]  
fim para

para **I** de 0 até 99  
    faça escrever NOTA[**I**]  
fim para

Escrita das notas

escrever (**SOMA**)

fim-programa

## VETOR - Exemplo 3

---

- Ler um conjunto de 100 notas armazenando-as no vetor denominado NOTA. (Esse exemplo também pode ser feito com subrotina)
- Verificar se existe nota 10.0. Se existir, dizer quantas existem

## VETOR - Exemplo 3

programa VET3

declarações

inicio

para I de 0 até 99

faça ler NOTA[I]

fim para

CONTADOR ← 0

para I de 0 até 99

faça se NOTA[I] = 10.0

então CONTADOR ← CONTADOR + 1

fim-se

fim para

escrever (CONTADOR)

fim-programa

### VETOR - Exemplo 3

programa VET3

declarações

inicio

para **I** de 0 até 99  
    faça ler NOTA[**I**]  
fim para

**CONTADOR**  $\leftarrow$  0

para **I** de 0 até 99

faça se **NOTA**[**I**] = 10.0

então **CONTADOR**  $\leftarrow$  **CONTADOR** + 1

fim-se

fim para

escrever (**CONTADOR**)

fim-programa

Leitura das notas

Inicialização  
do contador

Verificação se  
a nota é 10.0

Aumento do contador  
quando a nota é 10.0

## VETOR - Exemplo 4

---

- Ler um conjunto de 100 notas armazenando-as no vetor denominado NOTA. (Esse exemplo também pode ser feito com subrotina)
- Calcular a média, verificar e exibir as notas abaixo da média.



## VETOR - Exemplo 4

programa VETOR13

declarações

inicio

para I de 0 até 99

faça ler NOTA[I]

fim para

SOMA ← 0

para I de 0 até 99

faça SOMA ← SOMA + NOTA[I]

fim para

MEDIA ← SOMA/100

para I de 0 até 99

faça se NOTA[I] < MEDIA

        | então escrever (NOTA[I])

fim-se

fim para

fim-programa

## Exemplo 4

programa VETOR13

declarações

inicio

```
para I de 0 até 99  
  faça ler NOTA[I]  
fim para
```

Leitura das notas

```
SOMA ← 0  
para I de 0 até 99  
  faça SOMA ← SOMA + NOTA[I]  
fim para  
MEDIA ← SOMA/100
```

Cálculo da Média

```
para I de 0 até 99  
  faça se NOTA[I] < MEDIA  
    | então escrever (NOTA[I])  
    fim-se  
fim para
```

Escrita das notas  
abaixo da média

fim-programa

## programa VETOR13

declarações

inicio

para I de 0 até 99

faça ler NOTA[I]

fim para

SOMA ← 0

para I de 0 até 99

faça SOMA ← SOMA + NOTA[I]

fim para

MEDIA ← SOMA/100

para I de 0 até 99

faça se NOTA[I] < MEDIA

| então escrever (NOTA[I])

fim-se

fim para

fim-programa

## VETOR - Exemplo 4

Podem se  
passados para  
C  
diretamente  
Já são  
instruções  
conhecidas.



programa VETOR13

VETOR Exemplo 4

**declarações**

inicio

para I de 0 até 99

faça ler **NOTA[I]**

fim para

SOMA ← 0

para I de 0 até 99

faça SOMA ← SOMA + **NOTA[I]**

fim para

MEDIA ← SOMA/100

para I de 0 até 99

faça se **NOTA[I]** < MEDIA

| então escrever (**NOTA[I]**)

fim-se

fim para

fim-programa

Como são  
declaradas  
as variáveis  
indexadas  
unidimensionais?

# Declaração (C) de Variável Indexada Unidimensional

---

- Deve ser especificado o número máximo de elementos do conjunto
- Deve ser especificado o tipo dos elementos do conjunto
- Exemplo:

```
int X[100];
```

# Declaração (C) de Variável Indexada Unidimensional

- Deve ser especificado o número máximo de elementos do conjunto
- Deve ser especificado o tipo dos elementos do conjunto

**Nome da Variável**

**Número máximo de elementos do conjunto**

**Tipo dos elementos do conjunto**

```
float Nota[100];
```

# VETOR - Exemplo

---

- Ler um conjunto de **N** números inteiros (N é lido e é menor que 100). (Esse exemplo também pode ser feito com subrotina).
- Encontrar e exibir o maior deles. (Esse exemplo também pode ser feito com subrotina)

programa MAIOR

declarações

inicio

ler (**N**)

para **J** de 0 até **N -1**   // for(j=0; j<N; j++)

faça ler NRO[**J**]

fim para

    MAIOR  $\leftarrow$  NRO[0]

para **I** de 1 até **N -1**

faça se NRO[**I**] > MAIOR

            | então MAIOR  $\leftarrow$  NRO[**I**]

fim-se

fim para

    escrever (MAIOR)

fim-programa

# Algoritmo

programa MAIOR  
declarações  
inicio

ler (N)

Lê a quantidade  
de números

para J de 0 até N-1  
  faça ler NRO[J]  
fim para

Lê os números

MAIOR  $\leftarrow$  NRO[0]

Assume que o  
primeiro elemento do conjunto  
é o maior de todos

para I de 1 até N-1  
  faça se NRO[I] > MAIOR  
    | então MAIOR  $\leftarrow$  NRO[I]  
  fim-se

Compara o  
elemento assumido  
como maior com todos  
os outros, colocando em  
MAIOR aquele elemento  
que realmente for o  
maior de todos

fim para

escrever (MAIOR)

fim-programa

# VETOR - Exercício em Classe

---

- Escrever um programa que:
  - a) Leia uma lista **A** de **N** números inteiros (vetor). (Subrotina)
  - b) Leia um elemento inteiro **E** e insira este elemento na lista, em uma posição **P** lida pelo programa. (Subrotina)
  - c) Exiba a nova lista assim obtida (Subrotina)

ESTRUTURAS COMPOSTAS

Variáveis Condicionais

**FIM**

Introdução à Ciência da  
Computação