

Dados Estruturados

Vetores e Matrizes

SCC120 - Introdução à Ciência de Computação

São Carlos
Abril de 2010

Array - Definição

- *Vetor* ou *Array* é a forma mais familiar de dados estruturados.
- Um *array* é um conjunto de componentes do mesmo tipo.

LABIC - rafjr

Array - Problema

Dada uma relação de 5 estudantes, imprimir o nome de cada estudante, cuja nota é maior do que a média da classe.



LABIC - rafjr

Array - Solução

Imagine fazer um algoritmo deste tipo para 100 números!!!

1º. Algoritmo

Início

Leia(nome1, nota1, nome2, nota2, nome3, nota3, nome4, nota4, nome5, nota5)

media ← (nota1+nota2+nota3+nota4+nota5) / 5,0

Se nota1 > media **então** escreva (nome1)

Se nota2 > media **então** escreva (nome2)

Se nota3 > media **então** escreva (nome3)

Se nota4 > media **então** escreva (nome4)

Se nota5 > media **então** escreva (nome5)

Fim

LABIC - rafjr

Array - Solução 1

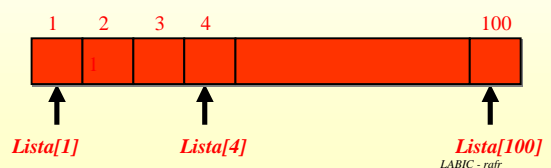
1. Uma variável para cada nome → 100 variáveis
2. Uma variável para cada nota → + 100 variáveis
3. 100 testes



LABIC - rafjr

Array - Definição

- Como estes dados têm uma relação entre si, podemos declará-los com um nome **ÚNICO** para todos os 100 elementos.
- Conjunto de 100 números = **LISTA**



LABIC - rafjr

Array - Definição

- O elemento do vetor tem todas as características de uma variável e pode aparecer em expressões e atribuições.

Lista[2] ← Lista[3] + Lista[20]

- Para somar todos os elementos da Lista:

soma ← 0

para I ← 1 até 100 faça

soma ← soma + Lista[i]

LABIC - rafjr

Array - Características

- As características básicas são:
 - é uma estrutura homogênea, isto é, formada de elementos do mesmo tipo
 - todos os elementos da estrutura são igualmente acessíveis, isto é, o tempo e o tipo de procedimento para acessar qualquer um dos elementos do *Array* são iguais
 - cada elemento componente desta estrutura tem um nome próprio segundo sua posição no conjunto

LABIC - rafjr

Array - Solução 2

2º. Algoritmo



Início

Para i ← 1 até 5 faça

Leia(nome[i], nota[i])

soma ← 0,0

Para i ← 1 até 5 faça

soma ← soma + nota[i]

media ← soma/5

Para i ← 1 até 5 faça

Se nota[i] > media então escrever (nome[i])

Fim

LABIC - rafjr

Array - Declaração de Tipos

- arrays são agrupamentos de dados adjacentes na memória
- declaração:

tipo_dado nome_array[<tamanho>];

define um arranjo de <tamanho> elementos adjacentes na memória do tipo *tipo_dado*

LABIC - rafjr

Array - Problema 2



Para um vetor A com N números, formular um algoritmo que determine o maior e o menor elemento deste vetor. Imprimir o vetor, o maior e o menor elemento.

LABIC - rafjr

Array - Solução

```
#include<stdlib.h>
#include<stdio.h>
#include <conio.h>
int A[10]; int i,Maior, Menor, N;
int main(){
    printf("Digite valor de N: ");
    scanf("%d",&N);
    for(i=0;i<N;i++){
        printf("%d numero:",i);
        scanf("%d",&A[i]);
    }
    Maior=A[0];
    Menor=A[0];
    for(i=1;i<N;i++){
        if (Maior < A[i]) Maior=A[i];
        if (Menor > A[i]) Menor=A[i];
    }
    printf("%d %d", Maior,Menor);
    getch( );
}
```

LABIC - rafjr

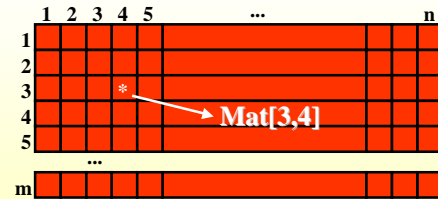
Matrizes - Definição

- Também chamadas conjuntos bidimensionais, contém:
 - um número fixo de elementos;
 - todos são do mesmo tipo;
 - arranjados na forma de tabela de 2 dimensões;

LABIC - rafjr

Matrizes - Definição

- Ex.: Uma matriz chamada **MAT** que tenha m elementos (horizontal) e n elementos (vertical)



LABIC - rafjr

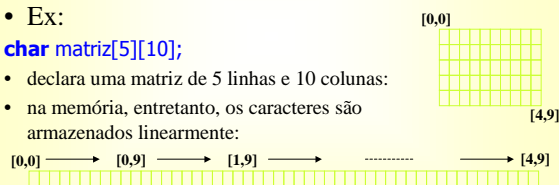
Arrays Multidimensionais

- Arrays* podem ter diversas dimensões, cada uma identificada por um par de colchetes na declaração

- Ex:

```
char matriz[5][10];
```

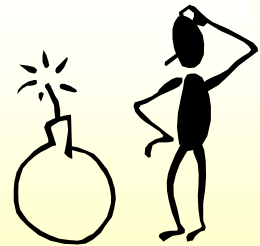
- declara uma matriz de 5 linhas e 10 colunas;
- na memória, entretanto, os caracteres são armazenados linearmente:



LABIC - rafjr

Matrizes - Problema

Dada uma tabela de 4x5 elementos, calcular a soma dos elementos e o maior elemento.



LABIC - rafjr

Matrizes - Solução

```
Int main(){
    int A[4][5];
    Int i, j, Maior, Soma;

    {Leitura dos Dados}
    for (i=0; i<4; i++)
        for (j=0; j<5; j++)
            scanf("%d", &A[i][j]);
```

(continua no próximo slide...)

LABIC - rafjr

```
{continuação...}
{Inicialização de variáveis}
Soma=0;
Maior=A[0][0];

{Cálculo da Soma}
for (i=0; i<4; i++)
    for (j=0; j<5; j++){
        soma=soma + A[i][j];
        if Maior < A[i][j]
            Maior=A[i][j];
    }

{Impressão dos Resultados}
printf("%d %d", Soma, Maior);
}

{Fim do programa}
```

LABIC - rafjr

Exercício 1

Dado um vetor VET, definido por:

tipo VET = vetor[1:100] - inteiros
v : VET

- a) preenchê-lo com o valor inteiro 30;
- b) preenchê-lo com os números inteiros 1,2,3..100;
- c) preencher VET[i] com 1, se i é um quadrado perfeito, e com 0, nos demais casos.

LABIC - rafjr

Exercício 2

Fazer um programa em C para somar dois vetores de mesmo número de elementos.



LABIC - rafjr

Exercício 3

Fazer um programa para calcular a soma de duas matrizes reais de dimensão 3x5.



LABIC - rafjr

Exercício 4

Fazer um programa para gerar a matriz transposta de uma matriz 3x3.



LABIC - rafjr

Exercício 5

Dada uma matriz MAT de 4x5 elementos, fazer um programa para somar os elementos de cada linha gerando o vetor **SOMA**. Em seguida, somar os elementos do vetor **SOMA** na variável **TOTAL**, que deve ser impressa no final.

LABIC - rafjr

Dados Estruturados

Arrays: Vetores e Matrizes

*Material Didático preparado por:
profa. Roseli Romero*

Fim