

Estruturas de dados

Manipular Vetores e matrizes

Comprimento Fixo - Arrays

- Vetores/Matrizes (arrays)
 - Um array é uma estrutura de dados que agrupa dados em linhas e/ou colunas
 - Pode ser unidimensional (vetor) ou multidimensional (matriz);

Vetor de nomes dos alunos

1	2	3	...	49	50
João	Pedro	Carlos	...	José	Maria

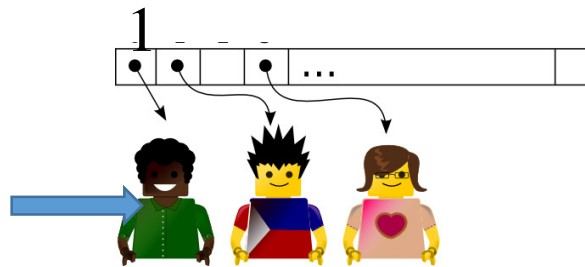
Matriz das notas dos alunos

	1	2	3	4
1	9,5	10	8	7,5
2	10	9	9	5,5
3	9	8,5	9,5	7
...
49	7	10	10	9
50	7	8,5	5,5	4

Estruturas de dados – Vetores

- Vetor (Array unidimensional)
 - Apresentam a capacidade de armazenar vários valores (dados) com uma única referência de nome dado ao vetor;
 - por exemplo: uma coleção de nomes de alunos;
 - Um elemento do array pode ser referenciado com o nome do array e o seu índice entre [];

Ex: aluno[1]



Estruturas de dados – Vetores

- Vetor (Array unidimensional)
 - Um array é uma porção de memória fixa e sequencial dividida em pedaços idênticos
 - Em cada posição do array, podemos guardar por exemplo o nome de um aluno;
 - Exemplo da declaração de um array em JAVA:
 - `int[] idades = {10, 20, 30, 40};` // declara e inicializa um array de inteiros com 4 elementos.

Estruturas de dados – Vetores

Manipulação de Vetores

- Escrever um valor numa posição do array:
- Leia(aluno[1])
 - aluno[1] – Acede ao segundo elemento do array;
 - Necessário usar o nome do array e o índice do elemento que se pretende obter
- Aceder (Escrever) a um elemento do array:
 - Escreva(aluno[2]):
 - aluno[1] <- “José” – preenche/subscreve o nome José ao aluno cuja posição é 2 (índice 1), continuando as restantes posições intactas;

0	1	2	3	...
Ana	José	Carlos	Paula	...

- Na prática o índice de um vetor é um valor inteiro, sendo nas várias Linguagens de Programação normalmente iniciado em 0;

Estruturas de dados – Vetores

Manipulação de Vetores

- Pode ser usado um valor variável para aceder a um array ():

```
i <- 2
```

```
aluno[i] <- "Maria"
```

```
Escreva(aluno[i])
```

0	1	2	3	...
Ana	José	Maria	Paula	...

Estruturas de dados – Vetores

Manipulação de Vetores

- Cálculo com elementos de um array
 - Cada elemento do vetor comporta-se como uma variável qualquer e todas as operações (por exemplo aritméticas - +, ~/, *) sobre variáveis são válidas.
 - Exemplo:
a[0] <- 3
a[1] <- 5
a[4] <- a[1] + a[0]

	0	1	2	3	4	5	6	7	8	9
a	3	5			8					

Vetores (implementação)

Exemplo de implementação em VisualG

- Uso: <nome>: vetor [<n>..<m>] de <tipo_dados>
 - <nome> = nome do vetor;
 - <n> = índice do primeiro elemento;
 - <m> = índice do último elemento;
 - <tipo_dados> = tipo de dados dos elementos;
- Exemplo de declaração e atribuição de valores às posições do vetor:

```
var
    numeros: vetor [0..9] de inteiro
...
inicio
    numeros[0] <- 5
    leia(numeros[6] // utilizador insere 99
    numeros[1] <- 10
    numeros[9] <- 0
    numeros[2] <- 4
    escreva(numeros[6]) // 99
...
fimalgoritmo
```


Atividade

1. Declare um vetor de 5 inteiros;
2. Preencha o vetor com valores (sem usar ciclos);
3. Mostre os valores no ecrã;

Estruturas de dados – Vetores

Percorrer um array (unidimensional)

```
#include <iostream>
#include <stdio.h>
using namespace std;

int main()
{
    setlocale(LC_ALL, "Portuguese");
    int i;
    int notas[4] = {10,12,14,13};

    for ( i = 0; i < 4; i++) {
        cout << notas[i] << "\n";
    }
}
```

Atividade

1. Leia para um array 10 inteiros, calcule a sua média e imprima o resultado;
2. Altere o exercício anterior de modo a imprimir o maior valor recebido;
3. Ordene um vetor de 10 inteiros de acordo com o seguinte algoritmo em “Português Estruturado” (ordenação por inserção):

```
para j ← 2 até comprimento do vetor, faça  
    elemento ← vetor[j]  
    i ← j - 1  
    enquanto i > 0 e vetor[i] > elemento, faça  
        vetor[i + 1] ← vetor[i]  
        i ← i - 1  
    fimenquanto  
    vetor[i + 1] ← elemento  
fimpara
```

Funcionamento:

1. Percorre um vetor sempre a partir do primeiro índice desordenado (inicialmente o 2º elemento);
2. Procura inserir esse elemento na posição correta comparando-o com o seu(s) anterior(es);
3. Troca-os de lugar enquanto ele for menor do que o(s) seu(s) precedente(s);

Atividade

1. Copiar os dados de um VET1 (5 posições) para as primeiras 5 posições de COPIA e VET2 (5 posições) para as outras 5 e imprima o vetor resultante (COPIA).
2. Imprima o vetor "COPIA" na sequencia inversa à recebida.