FICHA 02 – TABELAS

O objectivo central desta ficha é aprender a utilizar tabelas em Java.

Tabelas em Java

Em Java as tabelas ou **arrays** são objectos constituídos por um conjunto de dados todos do mesmo tipo primitivo (int, char, float, double, etc.) ou de uma mesma classe e suas descendentes.

1. Para criar um array faz-se:

int [] numeros = new int [10];

ou:

int [] numeros; // definição
numeros = new int [10]; // criação

2. Os elementos de um array são identificados pelo nome do array e por um índice que indica a sua posição no array. numeros [i] representa o elemento na posição i. O primeiro elemento do array tem o índice 0.

3. A criação de um array e a inicialização dos seus elementos também pode ser feita em simultâneo:

int [] tab = {1, 3};

4. A dimensão máxima de um array é dada pelo campo length. Após a sua criação não é possível alterar o valor deste campo.

5. É possível passar um array como argumento de um método. O que é passado é um handle do array e o seu valor pode vir alterado após o retorno do método.

6. A criação de um array bidimensional é feita de forma semelhante:

double [][] x = new double [5][5];

7. Os elementos do array são identificados pelo nome do array e por dois índices que indicam a sua posição no array. x [i][j] representa o elemento do array x que se localiza na linha i coluna j. Os índices de linha e de coluna começam em zero.

8. Num array bidimensional o campo length indica o número de linhas e x [0].length o número de colunas.

9. A construção de arrays de qualquer dimensão faz-se de forma semelhante.

10. A cópia de arrays é mais eficientemente realizada usando o método:

public static native void arraycopy (Object src, int src_position,
Object dst, int dst_position, int length)

que copia um array src, a partir da posição src_position, para outro dst, a partir da posição dst_position. O número de elementos copiados é dado por length.

Exercícios

1. Merge de tabelas

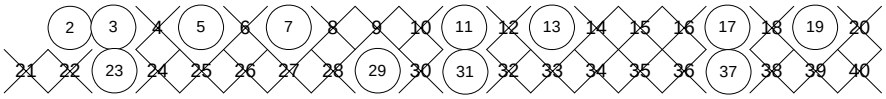
Escreva um programa que leia duas tabelas de n números inteiros cada e determine outra que se obtém das primeiras intercalando os respectivos elementos, mas com os da segunda pela ordem inversa.

2. Multiplicação de matrizes

Escreva um programa que leia duas matrizes rectangulares quaisquer, proceda à sua multiplicação, se for possível, e apresente a matriz resultante. Quando não for possível efectuar a multiplicação deve apresentar uma mensagem elucidativa.

3. Peneira de Eratosthenes

Uma forma simples e eficiente de calcular todos os números primos até um certo valor n é o método da Peneira de Eratosthenes. O processo é simples: escrevem-se todos os valores entre 2 e n (limite máximo). Em seguida faz-se um círculo em volta do 2, marcando como primo e riscam-se todos os seus múltiplos. Continua-se a fazer um círculo em volta do menor inteiro que encontrar, eliminando todos os seus múltiplos. Quando não restarem números sem terem círculos à volta ou traços por cima, os números com círculos à volta representam todos os primos até n. A figura seguinte apresenta o método para n=40.



a) Crie um método que implemente a Peneira de Eratosthenes. Este método deve ter como parâmetro de entrada um número inteiro correspondente ao limite máximo e deve devolver uma tabela contendo todos os números primos encontrados.

b) Desenvolva um programa que, dados x números inteiros inseridos pelo utilizador, identifique todos os números primos (o valor de x deve ser pedido ao utilizador). Na elaboração do programa deve utilizar o método desenvolvido na alínea anterior.

4. Caça aos números repetidos

Pretende-se que desenvolva um programa capaz de simular uma versão simplificada do jogo “Sudoku”. O objectivo do jogo é preencher as células vazias de uma matriz 9x9 com números de 1 a 9 de modo a que em cada linha e em cada coluna não haja números repetidos. Assume-se que as células vazias estão preenchidas com o número 0.

a) Crie um método que preencha e devolva uma matriz 9x9. As células da matriz devem ser preenchidas com números aleatórios entre 0 e 9. Tenha em atenção que no preenchimento da matriz deve garantir que a regra de não repetição de números descrita acima é cumprida. Note que esta regra se aplica apenas aos números entre 1 e 9.

b) Crie um método que, recebendo uma destas tabelas e as coordenadas x, y de uma posição, crie e devolva um vector contendo os números que nesse momento podem

ocupar essa posição da tabela, ou seja todos os que não pertencem à linha e à coluna do elemento em causa.

Exemplo:

Tabela original:

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

Vector resultante da chamada do método relativo à posição vazia marcada com um X na tabela original:

1 7 8 9

5. Quebra-cabeças

Considere que quer resolver um quebra-cabeças que surgiu no jornal. Existe um rectângulo com um determinado número de linhas e um determinado número de colunas preenchido com letras. A figura mostra um exemplo de um quebra-cabeças com 5 linhas e 6 colunas:

E	b	a	u	l	q
L	e	r	r	s	s
u	w	u	q	g	r
a	a	l	l	u	a
p	m	h	u	d	j

Desenvolva um método que procure todas as ocorrências de determinada palavra no quebra-cabeças. A palavra pode ocorrer numa linha ou numa coluna, mas não na diagonal. De cada vez que o método encontrar uma ocorrência da palavra, deve

escrever no monitor o número da linha e da coluna em que a palavra tem início. O método deve receber como argumentos a tabela e a palavra a pesquisar. Considerando o exemplo da figura, se a palavra a pesquisar for lua o método deveria escrever:

A palavra lua surge:

- Ao longo da coluna 0 com início na linha 1

- Ao longo da linha 3 com início na coluna 3

6. Linha (problema para avaliação)

Escreva um programa que gere aleatoriamente as ordenadas de um conjunto de n pontos do plano e as armazene numa tabela unidimensional. Admita que as abcissas dos pontos são dadas pelo índice da tabela*10. Desenvolva um algoritmo que 'alise' a linha formada por todos os pontos de modo a que fiquem todos a pertencer à mesma linha recta. Em cada iteração o alisamento é feito substituindo cada ordenada pela média da do próprio ponto com a do seguinte.