

- Desenvolva um método que recebe como parâmetro um número inteiro e decomponha os dígitos desse número em um vetor de inteiros, onde cada dígito ocupará uma posição do vetor. Por exemplo, se o método receber o número 1300, deverá retornar o vetor {1,3,0,0}
- QuadradoMágico** (QM). Um QM é uma matriz quadrada (mxn) que é preenchida com números de 1,2,3,4,..., n^2 e cuja a soma dos elementos em cada linha, cada coluna e nas duas diagonais é sempre igual. Por exemplo a matriz abaixo é uma QM.

16	3	2	13
5	10	11	8
9	6	7	12
4	15	14	1

Faça um programa em Java com os seguintes métodos estáticos:

- Método geraMatrizQuadrada(): que ao receber a ordem (n), retorne uma matriz preenchida com elementos de valores entre 1 e n^2
 - Método verificaQM(): recebe uma matriz e verifica se ela é uma QM. Retorna true ou false
 - Método exibeMatriz(): que exibe uma matriz recebida como parâmetro.
- Escrever um programa que gere valores aleatórios para uma matriz de inteiros. Após, exiba para cada linha, o percentual de valores pares e o percentual de valores ímpares. Deve-se pedir que o usuário informe o tamanho da matriz (número de linhas e de colunas). Modularize o programa de forma que seja desenvolvido um método para gerar a matriz, onde a quantidade de linhas e colunas sejam recebidas como parâmetros; outro método para completar a matriz com valores aleatórios e um último método para exibir as porcentagens de valores pares e ímpares para cada linha da matriz.

Exemplo:

Linha 0: 100% de pares e 0% de ímpares	
Linha 1: 0% de pares e 100% de ímpares	

- Desenvolva um método estático em Java que retorne uma matriz de mxn posições com valores inteiros aleatórios, m e n são recebidos como parâmetro. Desenvolva outro método que receba a matriz gerada no método anterior e retorne um vetor com os elementos pares e ímpares separados da matriz. Sugestão: no início do vetor armazene os elementos pares da matriz e nas posições restantes do vetor armazene os elementos da matriz que são ímpares.

5) MEGASENA:

A Mega-Sena é a maior loteria do Brasil. Os sorteios acontecem duas vezes por semana, às quartas-feiras e aos sábados. O princípio do jogo é simples: são sorteados seis números diferentes, compreendidos entre 01 e 60 (inclusive). Os sorteios são realizados em globos duplos (duas gaiolas esféricas girando). Os números são sorteados em dígitos separados, que formam um número decimal de dois dígitos, de 01 a 60. A primeira gaiola tem esferas variando de 0 a 5 para o primeiro dígito, e a outra tem esferas variando de 0 a 9 para o segundo. O número 00 corresponde ao 60. Quando seis pares de números originais são extraídos o sorteio está concluído.

Os apostadores podem apostar de seis a quinze números do total de 60.

Desenvolva um algoritmo em Java que pergunte ao usuário quantos jogos ele irá fazer e qual a quantidade de números que ele irá apostar para estes jogos (todos deverão obedecer a mesma quantidade de números). Após, gere aleatoriamente os números para cada aposta (não é permitido números repetidos em uma mesma aposta), faça o sorteio da Mega-Sena (obedecendo a regra oficial) e por final analise todas as apostas, emitindo o resultado da apuração de cada uma delas.

Para facilitar o desenvolvimento, modularize a solução com os seguintes métodos:

- a) `int[] geraAposta(int num):` retorna um vetor com os números sorteados aleatoriamente. Recebe a quantidade de números a serem sorteados. Não deve permitir números repetidos.
- b) `int[] geraMegaSena():` retorna um vetor com os 6 números sorteados para a Mega-sena.
- c) `int verificaAposta(aposta[], megasena[]):` recebe uma aposta e os números da Mega-sena. Retorna a quantidade de números acertados.
- d) `MostraMegaSena(megasena[]):` apresenta todos os números sorteados de uma Mega-Sena
- e) `MostraAposta(aposta[]):` apresenta os números de uma determinada aposta.

O método principal `main()` deve controlar todo o programa. Dica: utilize uma matriz para armazenar todas as apostas.