

1. Desenvolva um algoritmo que terá dois vetores, no primeiro vetor serão pedidos dez números, em seguida serão pedidos mais dez números e armazenados no segundo vetor. Após armazenados os vetores deverá ser pedido para o usuário informar um cálculo a ser executado, as opções de cálculo serão:

- a) Somar
- b) Subtrair
- c) Multiplicar
- d) Dividir

Após informado o tipo de cálculo deverá ser realizado o mesmo com a posição de cada vetor.

Exemplo:

Vetor1	Vetor 2	Tipo de Cálculo	Resultado
10	1	Soma	11
20	2		22
30	3		33
40	4		44
50	5		55
60	6		66
70	7		77
80	8		88
90	9		99
100	10		110

2. Crie um algoritmo onde o usuário irá informar dez números distintos, após informados deverão ser exibidos os números informados, só que na ordem contrária.

Exemplo: Números informados -> 1, 7, 3, 9, 5, 6, 11, 8, 19, 3

O algoritmo deverá retornar -> 3, 19, 8, 11, 6, 5, 9, 3, 7, 1

3. Elabore um algoritmo onde deverão ser solicitados cinco números distintos. Após informados os cinco números deverá ser retornado em qual posição do vetor se encontra algum número que seja igual a dez, caso não haja nenhum número dez deverá retornar: Não foi encontrado nenhum número 10.

Exemplo 01	Exemplo 02
5	9
6	8
10	5
3	2
7	2
Resposta que o algoritmo deverá fornecer	
O número 10 está na posição 3 do vetor	Não foi encontrado nenhum número 10

4. Criar um algoritmo para criação de provas. De imediato será pedido o gabarito da prova que será composta por dez questões, as respostas de cada questão poderão ser A, B, C e D, se por ventura for digitado algum termo diferente dessas quatro letras deverá ser pedido novamente a questão.

Assim que o gabarito estiver disponível informe o nome de um aluno, enquanto o nome desse aluno for diferente de SAIR deverá ser pedido o nome e as dez questões respondidas pelo aluno (valide as questões caso não sejam informados A, B, C, D). Assim que obter as dez respostas do aluno informe quantas ele acertou e quantas errou (apenas quantidade).

Quando digitado SAIR deverá informar o nome e a quantidade de acertos de cada aluno, faça com que a ordem seja da maior quantidade de acertos até a menor.

5. Implemente um algoritmo onde o usuário irá informar quinze números inteiros. Após informados os quinze números informe os seguintes dados:
- Soma dos números
 - Média dos números
 - Quantidade de números maiores ou iguais a média
 - Quantidade de números negativos, neutros e positivos
 - Maior número
 - Menor número
 - Números pares
 - Números ímpares

6. Desenvolver uma agenda, nessa agenda deverá ter as seguintes ações:
- Cadastrar Contato
 - Alterar Contato
 - Excluir Contato
 - Pesquisar
 - Sair

Especificações de cada ação da agenda:

- O cadastro deverá ser pedido: Nome, data de nascimento, estado, cidade, endereço, telefone, celular e e-mail. Antes de cadastrar valide se o nome informado já existe na lista de contatos, caso já cadastrado deverá ser pedido outro nome para cadastrar.
 - Pesquisando através do nome do contato faça com que os dados sejam alterados.
 - Através do nome do contato deverá ser excluído.
 - A pesquisa poderá ser feita através de nomes, faça com que se a pessoa digitar um determinado nome apareça os dados do contato.
 - Sair do sistema.
7. Utilizando a função aleatório, faça com que 5 números gerados sejam adicionados ao vetor. Esses números que serão guardados não podem ser repetidos.
8. Crie uma matriz 5X6 onde o usuário irá informar números inteiros. Após informados deverá ser exibido:
- Quantidade de Números Pares.
 - Quantidade de Números Ímpares.
 - Média dos Números Informados.
 - Quantidade de Números Maiores ou Iguais a Média.
 - Exibir a matriz
9. Crie uma matriz 3X2, peça para que o usuário escreva os seis números e exiba nas proporções: 3X2 e 2X3. Chamamos isso de matriz transposta.

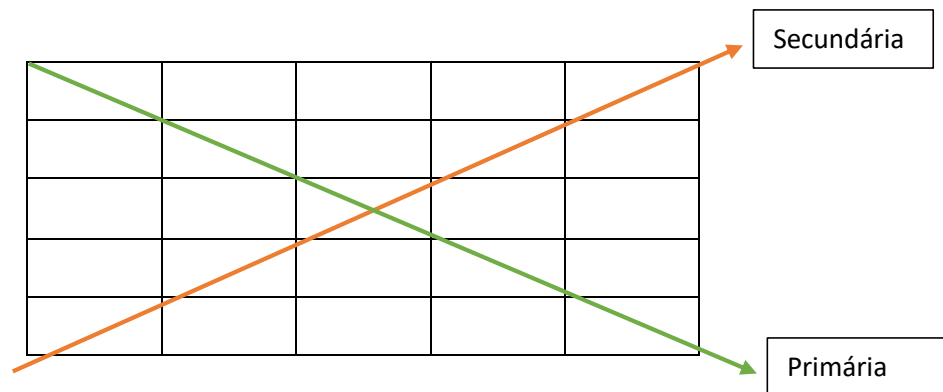
Imagine que o usuário coloque os números 1,7,4,8,9,3. Deverá exibir:

1	7	4
8	9	3

1	8
7	9
4	3

10. Desenvolva um algoritmo onde o usuário irá informar 25 números, essa matriz será 5X5. Após informado os números deverá retornar:

- A soma dos números informados na quarta linha.
- A soma dos números informados na segunda coluna.
- A soma da diagonal principal.
- A soma da diagonal secundária.
- A soma de todos os elementos da matriz.
- Escrever todos os elementos que compõem a matriz.



11. Criar um vetor 20x6 onde o usuário irá informar os seguintes dados:

- Nome
- Sexo
- Altura
- Peso

Nome, sexo, altura e peso ficarão nas colunas de posição 0,1,2 e 3, a coluna de posição 4 deverá ser calculado o IMC ($\text{peso} / \text{altura} * \text{altura}$) e a posição 5 deverá conter a situação de cada indivíduo.

Faça um laço para que sejam cadastrados um determinado número de pessoas (a quantidade pode variar).

Ao término exiba:

- Nome, imc e situação de todos os participantes.
- Quantidade e percentual de homens e mulheres que participaram
- Quantidade de cada situação (quantidade de pesos normais, abaixo, sobrepeso...)

IMC	Situações
Abaixo de 17	Muito abaixo do peso
Entre 17 e 18,49	Abaixo do peso
Entre 18,5 e 24,99	Peso normal
Entre 25 e 29,99	Acima do peso
Entre 30 e 34,99	Obesidade I
Entre 35 e 39,99	Obesidade II (severa)
Acima de 40	Obesidade III (mórbida)

12. Criar um jogo da velha, onde duas pessoas irão jogar.

13. Desenvolver um algoritmo para supermercado, nela deverá conter os seguintes produtos já cadastrados:

PRODUTOS COMERCIALIZADOS		
1	Arroz 5 KG	R\$ 9,50
2	Arroz 1 KG	R\$ 2,75
3	Feijão 3 KG	R\$ 5,60
4	Feijão 1 KG	R\$ 3,10
5	Macarrão 500 GR	R\$ 1,90
6	Farinha de Trigo 2 KG	R\$ 3,21
7	Ketchup 370 GR	R\$ 5,50
8	Maionese 480 GR	R\$ 7,30
9	Gelatina 50 GR	R\$ 0,65
10	Macarrão Instantâneo 120 GR	R\$ 1,19
11	Creme de Leite 150 GR	R\$ 1,89
12	Leite Condensado 150 GR	R\$ 2,88
13	Leite 1 L	R\$ 2,06
14	Chocolate em Pó 500 GR	R\$ 9,67
15	Pepino em Conserva 600 GR	R\$ 11,99
16	Vinagre 550 ML	R\$ 1,94
17	Sal 500 GR	R\$ 3,34
18	Açúcar 550 GR	R\$ 4,89
19	Fermento 150 FR	R\$ 2,19
20	Biscoito de Chocolate 70 GR	R\$ 2,09

O algoritmo com esses produtos pré-definidos deverá perguntar o código do produto a ser comprado e a quantidade, enquanto o código for diferente de zero deverá perguntar o código e a quantidade.

Após digitado zero (ou alguma outra opção para sair do sistema) deve ser exibido: código, nome do produto, quantidade comprada, valor unitário e total de cada produto (quantidade * produto) de cada pedido adicionado. Juntamente com o total a ser pago pelo cliente.

Informe também qual é o total da compra (soma de todos os totais).

Para os próximos exercícios você precisará utilizar a classe **Random**.

```
public static void main(String[] args) {  
  
    //Classe Random  
    Random gerarValor = new Random();  
  
    //Gerar valor inteiro de 0 a 9  
    int numero = gerarValor.nextInt(10);  
  
}
```

14. Implementar uma matriz 5X15 onde deverá ser exibida da seguinte maneira:

```
*****  
*           *  
*      5    *  
*           *  
*****
```

A primeira linha e última linha deverão aparecer apenas asteriscos.

A primeira coluna e a última coluna deverão aparecer apenas asteriscos.

A terceira linha 8 coluna deverá ser exibido um número entre 1 e 9 que deve ser gerado automaticamente.

As demais posições precisam estar vazias.

DICA! No JOptionPane os espaços geralmente não funcionam, para melhor visualização faça no console.

15. Criar uma matriz 10x10 onde o usuário irá informar uma posição. Você pode fazer com que ele peça a linha/coluna ou o número da posição.

Em uma das 100 posições poderá ter:

- j. Três bombas;
- k. Uma bandeira;
- l. 96 espaços em branco.

O jogo finaliza quando o usuário encontrar a bandeira ou se encontrar as três bombas. Quando clicado em um quadrado deverá ser dado dicas para saber onde está a bandeira:

Até duas posições: **Você está perto;**

Entre três e cinco posições: **Está longe;**

Acima de cinco posições: **Está muito longe.**

Valide para que o usuário não possa selecionar mais de uma vez uma posição.

Para cada erro deverá ser feito um contador, ao término se o usuário ganhar ou perder deverá ser informado quantos erros ele obteve. Caso seja informada uma posição que já foi dita não deverá ser contabilizado o erro.

Quando finalizar o jogo dependendo da quantidade de erros dê as seguintes mensagens:

- a) Nenhum erro: Nossa!!! Acertou na mosca.
- b) Até 5 erros: Parabéns! Você é bom nesse jogo.
- c) Até 10 erros: Ótimo, você foi muito bem.
- d) Até 20: Poderia ser melhor, tente novamente.
- e) Acima de 20: Você é muito ruim nesse jogo.

A bandeira e as bombas deverão ser geradas de maneira aleatória, sendo assim utilize a função: aleatório.

16. Criar um jogo da velha onde será jogador x computador.

17. Crie um caça palavras, onde haverá ao menos 10 palavras distintas.

Defina uma quantidade de vezes onde o usuário poderá jogar, enquanto ele não zerar as chances ou encontrar as dez palavras deverá ser perguntado.