



LISTA DE EXERCÍCIOS

- Arrays -

Vetores

1. Um baralho tem N cartas, não-repetidas, numeradas entre 1 e N. Faça um programa que leia o valor de N e imprima o resultado do "embaralhamento" das cartas.
2. Gere um vetor de K posições ($0 < K < 1000$) com valores aleatórios entre 0 e 1000. Conte e escreva quantos valores pares e ímpares ele possui. Imprima o maior, o menor valor e a média aritmética.
3. Gere dois vetores, V1 e V2, cada um com K valores aleatórios sorteados no intervalo entre 0 e X ($0 < K < X$). O usuário deve informar o valor de K e X, desde que $K < X$. O programa **NÃO** deverá aceitar números repetidos no mesmo vetor, e nem o mesmo número no mesmo índice dos dois vetores. Imprima os valores sorteados nos dois vetores, e o resultado da multiplicação dos valores dos índices de V1 e V2.
4. Faça um programa C para ler do usuário dois valores inteiros X e Y. Alimente dois vetores de tamanhos X e Y com números inteiros aleatórios, não repetitivos, sorteados no intervalo entre 0 e X+Y (inclusive). Faça a impressão das seguintes informações:
 - a) Vetores X e Y;
 - b) Todos os números exclusivos do vetor X;
 - c) Todos os números exclusivos do vetor Y;
 - d) Os números existentes nos dois vetores;
5. Métodos: `map`, `filter` e `reduce` são bastante comuns em linguagens de alto nível para tratamento de Arrays. Faça um programa que implemente o método...

MAP => Dado um vetor V1, com N números inteiros sorteados aleatoriamente (entre 0 e 20), e um valor inteiro X (sendo N e X informados pelo usuário), crie um novo vetor V2, preenchendo-o com os valores do primeiro vetor elevado à potência X. Imprima V1 e V2.
6. Métodos: `map`, `filter` e `reduce` são bastante comuns em linguagens de alto nível para tratamento de Arrays. Faça um programa que implemente o método...

FILTER => Dado um vetor V1, com N números inteiros sorteados aleatoriamente (entre 0 e 99), e um valor inteiro X (sendo N e X informados pelo usuário), crie um novo vetor V2, preenchendo-o apenas com os valores do primeiro vetor que são divisíveis por X. Imprima V1 e V2.
7. Métodos: `map`, `filter` e `reduce` são bastante comuns em linguagens de alto nível para tratamento de Arrays. Faça um programa que implemente o método...

REDUCE => Dado um vetor V, com N números inteiros sorteados aleatoriamente (entre 0 e 9), imprima a média dos valores sorteados, excluindo-se do cálculo o menor e o maior valor sorteado.
8. Faça um programa que preencha um vetor com N números aleatórios, sorteados no intervalo entre X e Y. O programa deve informar a posição (índice) onde se encontra o menor número sorteado, seguido da posição onde se encontra o segundo menor número sorteado, e assim sucessivamente...
9. Leia do usuário um valor X, tal que X seja um valor PAR maior ou igual a 10. Gere aleatoriamente X números inteiros não-repetitivos e armazene-os em uma estrutura do tipo Array. Imprima os números gerados na sequência em que foram sorteados. Após a impressão, troque no vetor a posição da metade inicial dos números sorteados pela metade final. Repita a impressão do vetor.

10. Faça um programa que sorteie aleatoriamente N números, no intervalo entre X e Y (inclusive). Após o sorteio, o programa deve imprimir os números sorteados originalmente. Após isso, e a cada iteração, o programa deve imprimir novamente a relação, mas agora trocando as posições do MAIOR elemento pelo MENOR elemento.

Uma vez trocadas as posições, estes mesmos número não poderão ser mais trocados. Repita essa operação até quando for possível...

Exemplo de Execução:

N = 8

X = 1

Y = 10

Original: 4 - 6 - 8 - 9 - 3 - 2 - 7 - 8

1ª Iteração: 4 - 6 - 8 - 2 - 3 - 9 - 7 - 8

2ª Iteração: 4 - 6 - 3 - 2 - 8 - 9 - 7 - 8

3ª Iteração: 8 - 6 - 3 - 2 - 8 - 9 - 7 - 4

4ª Iteração: 8 - 7 - 3 - 2 - 8 - 9 - 6 - 4

11. Faça um programa que preencha um vetor com N números inteiros, sorteados no intervalo entre 0 e X. O programa deve imprimir os números sorteados, e ao final, o seguinte resumo:
- 1) Quantos números diferentes foram sorteados;
 - 2) O número mais vezes sorteado;
 - 3) O número menos vezes sorteado;
 - 4) Quantidade de números não-repetitivos sorteados;
12. Faça um programa que leia um número inteiro "K" e verifique se ele é palíndromo. P.Ex.: 57875, 131, 5665, 9, etc...
13. Declare um vetor de K números inteiros e preencha-o com números informados pelo usuário. Imprima o vetor original, e o número de comparações e permutações necessárias para ordenação do vetor utilizando os seguintes algoritmos de ordenação: **Bubble Sort** e **Selection Sort**. Imprima também o vetor após a ordenação.
14. Faça um programa que preencha aleatoriamente um vetor de N números inteiros com valores entre X e Y (inclusive). Após isto, imprima o vetor, e as seguintes informações:
- a) média
 - b) moda
 - c) mediana
 - d) desvio padrão.

Strings

15. Faça um programa que leia várias strings até que o usuário informe uma contendo apenas letras ('A'-'Z' ou 'a'-'z').
16. Escreva um programa em C para ler uma frase. A seguir, imprima a frase com: (a) todas as letras maiúsculas, (b) todas as letras minúsculas, (c) início de cada palavra em maiúsculo.
17. Faça um programa que leia, em formato de string, dois números inteiros excepcionalmente grandes e imprima a soma destes números. P.ex.: "25996478547851225" e "1452565475541" = 25997931113326766
18. Faça um programa que leia, em formato de string, um valor numérico representado na base binária. O programa deve validar se o valor informado pelo usuário realmente é um número binário. Em caso positivo, o programa deve informar o valor correspondente na base decimal.
19. Faça um programa que leia uma string A e outra string B. O programa deve informar a seguinte informação, conforme o caso: "**String A está contida em B**" ou "**String B está contida em A**" ou "**As strings são incompatíveis**".

20. Faça um programa que leia em formato de string, o número de uma conta-corrente que possui 5 dígitos numéricos. (Não devem ser aceitos letras e/ou símbolos que não sejam numéricos). O quinto dígito é um DV (Dígito Verificador), que serve para verificar se a sequência de dígitos informados atendem a uma regra específica que valida o número da conta. A regra para cálculo do DV é a seguinte: 1º dígito multiplicado por 3; 2º dígito por 5; 3º dígito por 7 e o 4º dígito por 9. O quinto dígito (DV) deverá ser o resto da divisão inteira da soma destes produtos por 10. O programa deverá informar se a sequência informada corresponde ou não a um número de conta válido.

21. Faça um programa que calcule quantos segundos já se passaram no dia de hoje.
Obs.: Para obter uma *string* com a data/horário atual utilize as expressões abaixo...

```
time_t horario = time(NULL);  
printf("%s", asctime(localtime(&horario)));
```

22. Faça um programa que leia a data de nascimento do usuário no seguinte formato *string* DD/MM/AAAA. O seu programa deve validar a entrada, tanto no formato indicado, quanto na real existência da data informada. Após isto, obtenha a data atual, e calcule a idade completa do usuário (anos, meses e dias).

23. Faça um programa que leia uma string S, e criptografe-a com o seguinte algoritmo: sorteie um vetor de `strlen(S)` números, com valores aleatórios e não-repetitivos entre 0 e `strlen(s)`. Após o sorteio, embaralhe as letras da mensagem original, de acordo com as posições sorteadas no vetor. Atenção, todos os caracteres têm que ter sua posição inicial alterada!

Exemplo de Execução:

```
String:          AULA DE ED  
Vetor Sorteado:  1369847502  
Texto Cifrado:   EADUD LE A
```

24. Considerando o problema anterior, faça um programa que leia uma String S criptografada, e um vetor de `strlen(s)` números inteiros. Valide a consistência do vetor (não deve ter números repetidos e devem estar no intervalo adequado). Se tudo estiver em conformidade, informe a mensagem decifrada...

Exemplo de Execução:

```
String:          EADUD LE A  
Vetor:           1369847502  
Texto Decifrado: AULA DE ED
```

25. Em uma eleição para representante da turma, existem K candidatos. Faça com que seu programa leia o valor de K e o nome de cada um dos K candidatos.

Após isso, programe uma espécie de urna eletrônica, onde o voto do candidato é representado pelo seu número de sequência de cadastro na urna (p.ex.: 1 - 2 - 3 - ... - K).

O valor 0 corresponde a um voto em BRANCO, e um valor negativo representa um voto NULO.

A votação deve ser encerrada assim que qualquer valor maior do que K seja informado.

Ao final da votação, seu programa deve gerar o seguinte relatório:

- A quantidade total de votos;
- O nome e a quantidade de votos para cada candidato;
- A quantidade de votos nulos;
- A quantidade de votos em branco;
- O nome do vencedor das eleições;
- O percentual do vencedor em relação ao número total de votantes.

26. Escreva um programa em C para ler uma frase qualquer e contar o número de palavras existentes na frase. Considere uma palavra como sendo um conjunto maior que dois caracteres separados por um ou mais espaços em branco.

27. Faça um programa que leia o nome de um autor e imprima-o em formato de citação.

P.Ex.: Adriano Antunes Prates == Prates, Adriano A.

P.Ex.: João Joaquim José Júnior == Júnior, João J.J.

Matrizes

28. Gere aleatoriamente uma matriz $N \times N$ de inteiros sorteados no intervalo entre X e Y (inclusive).
Imprima-a em formato de tabela (linhas e colunas) e informe a localização do **maior** e do **menor** valor da matriz.

29. Gere aleatoriamente uma matriz $N \times N$ de inteiros sorteados no intervalo entre X e Y (inclusive).
Imprima-a em formato de tabela, e os valores que compõem a coluna que resulta na **maior** soma, e os valores da linha que resulta na **menor** soma.

30. Gere aleatoriamente uma matriz $N \times N$ de inteiros sorteados no intervalo entre 0 e N^2 (inclusive).
Todo número sorteado, obrigatoriamente deve ser exclusivo em sua linha e em sua coluna. Imprima a matriz em formato de tabela (linhas/colunas).

31. Gere aleatoriamente uma matriz $N \times N$ de inteiros sorteados no intervalo entre X e Y (inclusive).
Peça ao usuário o índice de duas colunas, e faça a troca das respectivas colunas. Faça o mesmo com duas linhas escolhidas pelo usuário. Imprima a matriz após realizadas as trocas.

32. Faça um programa que leia o Nome e a Idade de N pessoas. Imprima a relação de nomes ordenada pelas idades de forma decrescente.

33. Gere uma cartela de bingo 5×5 , com números aleatórios entre 1 e 75. (Em uma cartela de bingo, não há números repetidos e os números são apresentados em ordem crescente). Imprima a cartela e faça com que o programa sorteie as bolas, uma a uma, e realize a “marcação” da cartela em tempo real.

34. Crie um programa que leia o nome de N alunos. Após isto, para cada aluno cadastrado, imprima o nome e solicite ao usuário os valores das notas da prova 1, prova 2 e prova 3. Ao final, o programa deverá gerar um relatório em formato de lista, contendo NOME, NOTA 1, NOTA 2, NOTA 3, MÉDIA, e se está APROVADO ou REPROVADO.

35. Gere uma matriz 10×100 , com números aleatórios entre 1 e 4, que se refere às respostas de 10 questões de múltipla escolha, referentes a 100 alunos. Leia (a partir do usuário) um vetor de 10 posições contendo o gabarito de respostas corretas. O seu programa deverá comparar as respostas de cada candidato com o gabarito, e emitir a nota final de cada aluno. O programa também deverá informar: qual foi a média das notas, a maior nota, quantos alunos atingiram a maior nota, a menor nota, e quantos alunos atingiram a menor nota.