

AL08 - Algoritmos - Lista de Exercícios 08 - TIII - Repetição

Nível fácil e moderado

- 01) Escreva um programa que dado um valor numérico digitado pelo usuário (armazenado em uma variável inteira), imprima cada um dos seus dígitos por extenso. Exemplo: 4571 Resultado: quatro, cinco, sete, um.
- 02) Faça um programa que receba do usuário uma string. O programa imprime a string sem suas vogais. (retire as vogais e imprima).
- 03) Faça um programa que leia um vetor de caracteres, gere e imprima um outro vetor onde as vogais, do primeiro vetor, sejam substituídas pelo carácter '*'.
- 04) Desenvolva um algoritmo que leia duas palavras e verifique se uma palavra é anagrama da outra, ou seja, se as duas palavras podem ser escritas com as mesmas letras. Desconsidere números e caracteres especiais.
- 05) Desenvolva um algoritmo que leia uma frase e encontre a maior e a menor palavra, considerando o número de caracteres, nessa frase. Imprima as palavras encontradas.
- 06) Desenvolva um algoritmo que leia uma palavra e retire todos os caracteres repetidos da palavra.
- 07) Crie uma função que recebe um vetor de números inteiros e retorna um novo vetor sem elementos duplicados. Faça o programa que possibilite o uso da função.

Nível moderado

- 08) Faça um algoritmo e o respectivo programa em JS para ler um conjunto de números inteiros e colocá-los em ordem crescente.
- 09) Faça um algoritmo e o respectivo programa em JS que leia uma palavra (máximo de 50 letras) e some 10 no valor ASCII de cada caractere da palavra. Imprima a string resultante. (Criptografia)
- 10) Faça um algoritmo e o respectivo programa em JS que leia a string resultante do exercício 09 e subtraia 10 no valor ASCII de cada caractere da palavra. (descriptografia).
- 11) Desenvolva um algoritmo capaz de fazer a "criptografia" (codificação de dados visando a privacidade de acesso às informações) de uma palavra. Essa criptografia será feita através da codificação dos dados através de um processo de substituição de letras.

Essa codificação é chamada Cifra de César. Uma letra deve ser lida e utilizada como chave. A chave representa o número de rotações no alfabeto que deve ser feito antes de encontrar a nova letra após a criptografia. A letra 'A' representa uma rotação de uma letra para a esquerda, 'B' uma rotação de duas letras e assim sucessivamente. Faça isso para todas as letras das palavras, ignore caracteres que não sejam letras. Lembre-se que 'z' + 1 é igual a 'a'.

Exemplo:

Entrada: D

Claudio

Saída: Gpeyhms

Dicas:

Em JS, let str = "Hello";

```
let charCode = str.charCodeAt(1); // Obtém o código ASCII do
segundo caractere
```

```
console.log(charCode); // Exibe 101, que é o código ASCII de 'e'
```

```
charCode++; //vai para a letra seguinte
```

```
console.log(charCode);
```

```
let char = String.fromCharCode(charCode);
```

```
console.log(char); // Exibe 'e'
```

Somar 1 ao código de uma letra, implica em transformar esta no caractere seguinte.

Pré requisitos

<https://pt.wikipedia.org/wiki/ASCII>

https://pt.wikipedia.org/wiki/Cifra_de_C%C3%A9sar

12) Escreva um algoritmo que receba 2 strings (A e B) e retorne uma terceira string (C) formada pelos caracteres de A e B intercalados. Ex.: Se A='Quarta' e B='Segunda', a resposta deve ser 'QSueagrutnada'.

13) Escreva um programa que, a partir de um nome informado pelo usuário, exiba suas iniciais. As iniciais são formadas pela primeira letra de cada nome, sendo que todas deverão aparecer em maiúsculas na saída do programa. Note que os conectores e, do, da, dos, das, de, di, du não são considerados nomes e, portanto, não devem ser considerados para a obtenção das iniciais. As iniciais devem ser impressas em maiúsculas, ainda que o nome seja entrado todo em minúsculas. Exemplos: Maria das Graças Pimenta => MGP João Carlos dos Santos => JCS José da Silva => JS Pedro Pereira Teixeira => PPT.