

Strings

1. **Tamanho de strings.** Faça um programa que leia 2 strings e informe o conteúdo delas seguido do seu comprimento. Informe também se as duas strings possuem o mesmo comprimento e são iguais ou diferentes no conteúdo.

```
Compara duas strings
String 1: Brasil Hexa 2006
String 2: Brasil! Hexa 2006!
Tamanho de "Brasil Hexa 2006": 16 caracteres
Tamanho de "Brasil! Hexa 2006!": 18 caracteres
As duas strings são de tamanhos diferentes.
As duas strings possuem conteúdo diferente.
```

2. **Nome ao contrário em maiúsculas.** Faça um programa que permita ao usuário digitar o seu nome e em seguida mostre o nome do usuário de trás para frente utilizando somente letras maiúsculas. Dica: lembre-se que ao informar o nome o usuário pode digitar letras maiúsculas ou minúsculas.

3. **Nome na vertical.** Faça um programa que solicite o nome do usuário e imprima-o na vertical.

```
F
U
L
A
N
O
```

4. **Nome na vertical em escada.** Modifique o programa anterior de forma a mostrar o nome em formato de escada.

```
F
FU
FUL
FULA
FULAN
FULANO
```

5. **Nome na vertical em escada invertida.** Altere o programa anterior de modo que a escada seja invertida.

```
FULANO
FULAN
FULA
FUL
FU
F
```

6. **Data por extenso.** Faça um programa que solicite a data de nascimento (dd/mm/aaaa) do usuário e imprima a data com o nome do mês por extenso.

```
Data de Nascimento: 29/10/1973
Você nasceu em 29 de outubro de 1973.
```

7. **Conta espaços e vogais.** Dado uma string com uma frase informada pelo usuário (incluindo espaços em branco), conte:

- a. quantos espaços em branco existem na frase.
- b. quantas vezes aparecem as vogais a, e, i, o, u.

8. **Palíndromo.** Um palíndromo é uma sequência de caracteres cuja leitura é idêntica se feita da direita para esquerda ou vice-versa. Por exemplo: **OSSO** e **OVO** são palíndromos. Em textos mais complexos os

espaços e pontuação são ignorados. A frase **SUBI NO ONIBUS** é o exemplo de uma frase palíndroma onde os espaços foram ignorados. Faça um programa que leia uma sequência de caracteres, mostre-a e diga se é um palíndromo ou não.

9. **Verificação de CPF.** Desenvolva um programa que solicite a digitação de um número de CPF no formato **xxx.xxx.xxx-xx** e indique se é um número válido ou inválido através da validação dos dígitos verificadores edos caracteres de formatação.
10. **Número por extenso.** Escreva um programa que solicite ao usuário a digitação de um número até 99 e imprima-o na tela por extenso.
11. **Jogo de Forca.** Desenvolva um jogo da forca. O programa terá uma lista de palavras lidas de um arquivo texto e escolherá uma aleatoriamente. O jogador poderá errar 6 vezes antes de ser enforcado.

```
Digite uma letra: A
-> Você errou pela 1ª vez. Tente de novo!

Digite uma letra: O
A palavra é: _ _ _ _ O

Digite uma letra: E
A palavra é: _ E _ _ O

Digite uma letra: S
-> Você errou pela 2ª vez. Tente de novo!
```

12. **Valida e corrige número de telefone.** Faça um programa que leia um número de telefone, e corrija o número no caso deste conter somente 7 dígitos, acrescentando o '3' na frente. O usuário pode informar o número com ou sem o traço separador.

```
Valida e corrige número de telefone
Telefone: 461-0133
Telefone possui 7 dígitos. Vou acrescentar o digito três na frente.
Telefone corrigido sem formatação: 34610133
Telefone corrigido com formatação: 3461-0133
```

13. **Jogo da palavra embaralhada.** Desenvolva um jogo em que o usuário tenha que adivinhar uma palavra que será mostrada com as letras embaralhadas. O programa terá uma lista de palavras lidas de um arquivo texto e escolherá uma aleatoriamente. O jogador terá seis tentativas para adivinhar a palavra. Ao final a palavra deve ser mostrada na tela, informando se o usuário ganhou ou perdeu o jogo.