

Instituto Federal de Educação Ciência e Tecnologia do RN - IFRN

Exercícios

- 001** Escreva um programa que leia 50 números, armazenando-os em um *array* e mostre o décimo quinto elemento do *array* (índice 14). Nada mais deve ser mostrado pelo programa.
- 002** Escreva um programa que leia 72 números inteiros, armazene-os em *array* e mostre o último e o primeiro elemento. Cada número deve ser mostrado em uma linha, onde a primeira linha contém o último elemento e a segunda linha o primeiro elemento.
- 003** Escreva um programa leia 50 nomes e armazene-os em um *array*. O programa deve então ler um número que corresponde a um índice e mostrar o nome contido no índice **anterior** ao índice lido. Se o índice lido não estiver entre os índices válidos do *array* (ou for 0) o programa deve mostrar **Índice Inválido**.
- 004** Escreva um programa que leia 1.000 números inteiros e mostre o menor número. Apenas o número deve ser mostrado, nada mais.

OBS: É proibida a utilização do método sort

- 005** Escreva um programa que leia 80 números reais e armazene-os em um *array*. O programa deve ler então dois números inteiros, que correspondem a índices no *array*, calcular e mostrar a diferença entre o maior e o menor dos números nos índices lidos. O programa deve mostrar a diferença contendo 2 casas decimais. Se um ou os dois índices não forem válidos o programa deve considerar o valor 0.0 para o cálculo da diferença.
- 006** Escreva um programa que leia 50 números e mostre-os na ordem em que foram lidos, indicando ao lado se o mesmo é par ou ímpar. Cada linha deve conter o número, um espaço e a letra P ou I para os números pares e ímpares, respectivamente, como as duas linhas de exemplo abaixo:

```
10 P
13 I
```

Ao final o programa deve mostrar a quantidade de números pares e a quantidade de números ímpares, como na linha abaixo:

```
PARES=28
IMPARES=22
```

Os nomes devem ser em letras maiúsculas e o número deve ter **exatamente** dois algarismos.

- 007** Escreva um programa que leia 1.000 números inteiros, mostre quantos números negativos existem e quais são. Cada valor deve ser mostrado em uma linha. Os números devem ser mostrados na ordem em que foram lidos.
- 008** Escreva um programa que leia 30 nomes, converta-os em maiúsculos e mostre a primeira palavra de cada um, na ordem em que foram lidos. Cada nome deve ser mostrado em uma linha e nada mais deve ser impresso pelo programa.
- 009** Escreva um programa que leia 80 números e armazene-os em um *array*. Depois leia mais um número inteiro, que deve corresponder a um índice do *array*, e troque o último elemento com o elemento do índice lido. Caso o índice não seja válido o programa deve trocar o primeiro com o último elemento. Ao final o programa deve mostrar os elementos do *array* resultante, um elemento por linha, começando do índice 0 e terminando com o o último índice.
- 010** Escreva um programa que leia uma *string* e mostre quantas vogais existem na *string*. O programa só deve mostrar a quantidade de vogais em uma linha, nada mais.
- 011** Escreva um programa que leia 10 nomes e mostre um nome por linha e a quantidade de caracteres do nome. Considere o nome "Pedro Alvares Cabral", a linha deve ser mostrada da seguinte forma:

```
20:Pedro Alvares Cabral
```

onde 20 é a quantidade de caracteres e Pedro Alvares Cabral o texto.

Abaixo segue um exemplo de entrada e saída do programa.

Exemplo de execução	Entrada (entrada.txt)	Saída (saida.txt)
<pre>\$ ruby tamanho_nomes.rb Pedro Alvares Cabral Pero Vaz de Caminha Felipe Camarao Luis da Camara cascudo Joaquim Jose da Silva Xavier Marechal Deodoro da Fonseca Rui Barbosa Getulio Vargas Janio Quadros Luis Inacio Lula da Silva 20:Pedro Alvares Cabral 19:Pero Vaz de Caminha 14:Felipe Camarao 22:Luis da Camara cascudo 28:Joaquim Jose da Silva Xavier 27:Marechal Deodoro da Fonseca 11:Rui Barbosa 14:Getulio Vargas 13:Janio Quadros 25:Luis Inacio Lula da Silva \$</pre>	<pre>Pedro Alvares Cabral Pero Vaz de Caminha Felipe Camarao Luis da Camara cascudo Joaquim Jose da Silva Xavier Marechal Deodoro da Fonseca Rui Barbosa Getulio Vargas Janio Quadros Luis Inacio Lula da Silva</pre>	<pre>20:Pedro Alvares Cabral 19:Pero Vaz de Caminha 14:Felipe Camarao 22:Luis da Camara cascudo 28:Joaquim Jose da Silva Xavier 27:Marechal Deodoro da Fonseca 11:Rui Barbosa 14:Getulio Vargas 13:Janio Quadros 25:Luis Inacio Lula da Silva</pre>

012 Escreva um programa que leia um número n , depois leia n números e mais um número x . O programa deve então mostrar todos os números que são divisores de x , um por linha, na ordem em que foram digitados.

013 Escreva um programa que leia 40 *strings* que contem o número de matrícula, o nome, a nota do primeiro bimestre e a nota do segundo bimestre dos alunos de uma turma. O programa deve então calcular a média e escrever todos os dados com a média. Todas as notas devem ser exibidas com uma casa decimal. A *string* de entrada deve ser digitada da forma a seguir:

MATRICULA:NOME:NOTA1:NOTA2

Exemplo:

20121014080999:Antonio Conselheiro:8.5:8.5

O programa deve calcular a média e escrever uma linha para cada entrada, da seguinte forma:

MATRICULA:NOME:NOTA1:NOTA2:MEDIA

Exemplo:

20121014080999:Antonio Conselheiro:8.5:8.5:8.5

014 Escreva um programa que leia uma *string* qualquer e informe se ela é um palíndromo. Um palíndromo é uma *string* cuja sequencia de caracteres é a mesma se lida da esquerda para a direita ou da direita para a esquerda.

Exemplo:

A cara rajada da jararaca

Essa frase é um palíndromo, pois a sequencia de letras é a mesma nos dois sentidos: acararajadadajararaca

Outro exemplo de palíndromo:

Socorram-me, subi no onibus em Marrocos

O programa deve mostrar a palavra PALINDROMO (sem acento) caso a *string* digitada seja um palíndromo. Nada mais deve ser mostrado pelo programa.

015 Escreva um programa que leia uma *string* e criptografe a mesma através do método de criptografia **zenit polar** (http://pt.wikipedia.org/wiki/Zenit_Polar). O programa deve mostrar apenas o texto criptografado.

Observação: O programa só precisa funcionar corretamente para entradas que não contenham letras acentuadas.

016 Escreva um programa que leia um nome e escreva as iniciais do nome. As iniciais devem ser mostradas em maiúsculas.

Exemplo: para o nome **Ariano Suassuna** o programa de mostrar **AS**. Observe que não há espaços entre as letras e todas são maiúsculas.

017 Escreva um programa que leia um nome e mostre o nome no formato de citação, como no exemplo abaixo:

Para o nome:

Joao Maria Silva

O programa deve mostrar:

SILVA, J.M.

OBS: Elimine as ligações 'de', 'da' e 'e'

018 Escreva um programa que leia uma *string* e depois uma letra. O programa deve mostrar a quantidade de ocorrências da letra na *string*. O programa deve mostrar **apenas** um número inteiro que corresponde a quantidade de vezes que a letra aparece na *string*.

019 Escreva um programa que leia 50 nomes e mostre a primeira e a última inicial de cada, ordenadas, uma por linha.

Exemplo: Para o nome **Machado de Assis** o programa deve mostrar **MA**

020 Escreva um programa que leia um nome e converta as iniciais do nome em maiúsculas, caso elas não estejam, e as outras letras em minúsculas, caso já não estejam. O program deve mostrar o nome com as letras convertidas.

Se o nome digitado for:

Carlos drummond de ANDRADE

o programa deve mostrar:

Carlos Drummond De Andrade