

UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE
DEPARTAMENTO DE INFORMÁTICA E MATEMÁTICA APLICADA

Introdução às Técnicas de Programação — IMD0012

◁ Exercícios - parte 4 ▷

Observações:

1. as strings serão especificadas como **string**(x), onde x é a quantidade máxima de caracteres (incluindo \0) (leia com gets ou fgets)
2. as strings que não admitem espaço serão referidas como **palavra**(x) (leia com scanf)
3. assumo que a entrada respeita os limites de tamanho da string/palavra especificada, com isso não há risco de buffer overflow. Mas lembre-se de que gets e scanf não verificam os limites da string
4. assumo que não há caractere *multi-character* ('é', 'à', etc) nas strings
5. lembre-se de que os caracteres podem ser utilizados como inteiros e são valorados de acordo com a tabela ASCII

Questões:

1. ▷ Você está implementando um jogo revolucionário chamado campo minado... 1D. A entrada do programa consiste em uma palavra(20) com “.” e “*”, onde o primeiro representa um espaço vazio e o segundo representa uma bomba. Depois o programa deve ler um inteiro representando um índice na palavra (começando de 0) e escrever na tela “**bum!**” caso nesse índice haja uma bomba ou escrever na tela um inteiro representando quantas bombas há na adjacência do índice em questão.

Exemplo 1:

```
..**..*.*.. 0
0
```

Exemplo 2:

```
..**..*.*.. 1
1
```

Exemplo 3:

```
..**..*.*.. 2
bum!
```

2. ▷ ~~Rafael~~ Um professor de ITP quer saber quantos alunos ficaram com a nota acima da média de todas as notas. Escreva um programa que leia um número inteiro **n** (assuma $1 \leq n \leq 30$) e, em seguida, **n** números reais representando as notas dos **n** alunos. Depois o programa deve escrever na tela quantos alunos tiveram nota acima da média.

Exemplo 1:

```
28 7.00 7.60 4.21 0.28 7.57 3.75 9.25 5.64 9.75 2.93 7.02 7.04 6.31 2.37 6.19
9.47 8.00 4.79 5.44 2.47 1.15 9.42 9.69 5.98 5.64 9.47 8.46 2.04
15
```

3. ▷ Escreva um programa em C que leia uma string(100) e escreva na tela qual(is) letra(s) minúscula(s) **não** aparece(m) na string. Dica: utilize um vetor de tamanho 128 e realize a indexação de cada letra com a própria letra.

Exemplo 1:

```
Tres pratos de trigo para tres tigres tristes
b c f h j k l m n q u v w x y z
```

4. ▷ Escreva um programa em C que leia um número inteiro **n** (assuma $1 \leq n \leq 10$) e, em seguida, **n** números inteiros. O programa deve escrever “**sim**” se esses **n** números são uma permutação dos números entre 1 e **n** e “**não**” caso contrário. Por exemplo, se $n = 5$, então [1 3 4 2 5] e [5 2 3 1 4] são permutações de [1 2 3 4 5], enquanto não são: [1 6 4 3 2] (pois falta o 5) e [1 2 5 3 3] (pois falta o 4).

Exemplo 1:

```
4 1 4 3 7
não
```

Exemplo 2:

```
4 1 4 3 2
sim
```

5. ▷ Cheque esperto é um famoso pirata que sempre pede ajuda aos alunos de ITP para encontrar tesouros (nunca encontrou). Em seu mapa há 10 ilhas numeradas de 0 a 9. Ele não sabe em qual dessas ilha está o tesouro. Cada vez que ele visita uma dessas ilhas, os nativos afirmam que o tesouro está em outra ilha. Ele pediu sua ajuda para que determine a primeira ilha que, partindo da ilha 0, será visitada **uma segunda vez** se for seguindo a sugestão dos nativos. Ele acredita que lá deva estar o tesouro. A entrada consiste na sugestão dos nativos em cada uma das 10 ilhas, desde a ilha 0 até a ilha 9. A saída do seu programa deve ser a primeira ilha que será visitada uma segunda vez.

Exemplo 1:

```
1 2 3 4 5 6 7 8 9 0
0
```

Exemplo 2:

```
5 4 2 8 7 4 2 1 0 2
4
```

6. ▷ O MEC precisa de sua ajuda para automatizar a correção das provas objetivas do ENEM! Escreva um programa que leia um número inteiro **n** representando o número de questões ($1 \leq n \leq 20$). Em seguida leia as **n** respostas do gabarito e, em seguida, as **n** respostas do aluno. Assuma que as respostas estão sempre entre 1 e 5. Depois o programa deve escrever na tela quantas questões o aluno acertou e a string “**acertos**” ou “**acerto**” (para 1 acerto), conforme exemplo abaixo.

Exemplo 2:

```
4
1 2 3 4
1 5 3 5
2 acertos
```

Exemplo 2:

```
7
1 2 3 2 1 5 4
3 3 3 3 3 3 3
1 acerto
```

7. ▷ Escreva um programa que leia um número inteiro **n** representando uma quantidade de meses ($1 \leq n \leq 20$). Em seguida faça a leitura de **n** inteiros, representando a quantidade de dias que choveu em cada um desses meses (do mês 1 até o mês **n**). O programa deve em seguida escrever na tela o número do mês em que mais teve dias de chuva e a respectiva quantidade. Se houver empate entre mais de um mês, escreva-os todos. Assuma que choveu em pelo menos algum dia.

Exemplo 1:

```
4
9 9 9 9
1 2 3 4 9
```

Exemplo 2:

```
5
2 5 3 4 1
2 5
```

8. ▷ Escreva um programa que leia uma palavra(15) e um caractere (na mesma linha ou não). Em seguida deve escrever na tela quantas vezes esse caractere aparece na string.

Exemplo:

```
teste e
2
```

9. ▷ Escreva um programa que leia uma string(100). Em seguida deve escrever na tela a mesma string invertida.

Exemplo 1:

```
O seguro morreu de velho
ohlev ed uerrom oruges O
```

10. ▷ Um dos princípios utilizados para compactação de arquivos é que símbolos que ocorrem com mais frequência podem ser codificados com menos bits enquanto que os que ocorrem com menos frequência podem ser codificados com mais bits. Escreva um programa em C que leia uma string(200) e escreva na tela, em ordem alfabética, quantas vezes aparece cada uma das letras minúsculas e maiúsculas, desde que haja pelo menos uma ocorrência (maiúsculas primeiro).

Exemplo:

Ando devagar porque ja tive pressa

A: 1

a: 4

d: 2

e: 4

g: 1

i: 1

j: 1

n: 1

o: 2

p: 2

q: 1

r: 3

s: 2

t: 1

u: 1

v: 2