

## Algoritmos e Programação: Fundamentos

**PROVA DO GRAU B**

Resolver as questões com consulta **local** somente ao material apresentado em aula, ao repositório de exemplos da turma e aos códigos-fonte dos exercícios resolvidos pelo próprio aluno. Nos primeiros 15 minutos da prova será permitido baixar este material. Não é permitido consultar outros colegas, a Internet, e-mail, drivers virtuais na nuvem, redes sociais ou telefone celular. Cada questão deve ter seu próprio arquivo-fonte, identificado com o número da questão (1.c, 2.c, etc) em um diretório chamado **ProvaGrauA-NomeSobrenome**. O arquivo contendo o diretório compactado (**.zip**) deve ser enviado pelo Moodle até as 22h15. Os nomes das variáveis devem ser autoexplicativos.

1. **(0.5 pt)** Implemente uma função **sortear** que receba o intervalo de valores inteiros ***início*** e ***fim*** como parâmetro, sorteie e retorne um número dentro do intervalo (considerando intervalo fechado [início, fim]).

Exemplo de uso:

```
int nroSorteado = sortear(1,10);
```

2. **(4.0 pts)** Implemente um programa que, dado uma matriz 5x5 preenchida com valores entre 0 e 4, faça a contagem de quantos valores de cada um dos números possíveis esta matriz contém.

Exemplo de matriz:

0	4	3	2	1
1	0	2	4	0
3	2	4	4	0
1	2	3	4	0
0	0	0	1	2

Retornaria a seguinte contagem:

```
Quantidade de valores 0: 8  
Quantidade de valores 1: 4  
Quantidade de valores 2: 5  
Quantidade de valores 3: 3  
Quantidade de valores 4: 5
```

3. **(4.0 pts) Correção de provas objetivas:** faça um programa que permita cadastrar um vetor com o gabarito de uma prova objetiva contendo 10 questões, cujos resultados podem ser de 'a' a 'f'. Depois disso, permita ao usuário a opção de entrar com as respostas dadas por um estudante.

**Exemplo de gabarito:**

a	f	c	d	d	a	e	e	b	a
---	---	---	---	---	---	---	---	---	---

O programa deve:

- Contabilizar o número de respostas certas e dar a pontuação final, de 0 a 10.
- Durante o processo de correção, imprimir na tela quais questões foram acertadas e quais o estudante errou. Neste caso, deve-se imprimir a resposta correta.

**Exemplo:**

Respostas do estudante:

a	f	a	d	f	a	e	e	c	a
---	---	---	---	---	---	---	---	---	---

Saída do programa:

```

Questão 1: resposta correta
Questão 2: resposta correta
Questão 3: resposta incorreta. A resposta correta é c.
Questão 4: resposta correta
Questão 5: resposta incorreta. A resposta correta é d.
Questão 6: resposta correta
Questão 7: resposta correta
Questão 8: resposta correta
Questão 9: resposta incorreta. A resposta correta é b.
Questão 10: resposta correta
A pontuação do estudante é 7/10.

```

4. **(1.5 pts) Programa "Cebolinha":** ler uma *string* (array de char) e converter todos os r's em l's.

**Exemplo:**

Entrada:

O rato roeu a roupa do rei de Roma

Saída:

O lato loeu a loupa do lei de Loma

**OBS:** para ler uma string com espaços, podemos usar o comando `fgets()`:

```
char myString[100];  
printf("Digite uma palavra ou frase: ");  
fgets(myString, sizeof(myString), stdin);  
printf("String digitada: %s", myString);
```

**BOA PROVA!** 😊

“A ciência é feita de erros, mas são erros que é bom cometer, porque nos levam pouco a pouco à verdade.” — Jules Verne, *Viagem ao Centro da Terra*