

Lista de Exercícios 01: Variáveis, Expressões, Operadores, Condicionais, Função e Repetição

Questão 1) Suponha que, a partir do ano 2000, uma universidade utilize como matrícula de um aluno um número inteiro no formato AASDDD, no qual:

- os dois primeiros dígitos, representados por AA, são os dois últimos algarismos do ano de ingresso;
- o terceiro dígito, representado por S, vale 1 ou 2, conforme o aluno tenha ingressado no 1o ou 2o semestre; e
- os três últimos dígitos, representados por DDD, correspondem à sua colocação no vestibular da universidade.

Escreva um programa em C que leia a matrícula de um aluno e exiba o ano e o semestre em que ele ingressou nesta universidade e sua colocação no vestibular.

Exemplo de uma execução:

```
Digite a matricula do aluno: 251029
Ano: 25 Semestre: 1 Colocacao: 29
```

Questão 2)

- Faça uma função que retorne a soma dos divisores de um número recebido;
- Faça uma função para exibir todos os pares de números amigáveis entre [num_inicial, num_final]. Um par de números é amigável quando cada um deles é igual à soma dos divisores do outro. Use obrigatoriamente a função do item A.
- Escreva um programa que pergunte ao usuário 2 números (x, y) e torne-os pontos iniciais e finais de um intervalo. A seguir exiba todos os pares de números amigáveis entre x e y. Use obrigatoriamente a função do item B.

Exemplo de uma execução:

```
Informe um intervalo de numeros inteiros [x, y] (onde x < y): 2 20
Par amigavel: (2, 3)
Par amigavel: (3, 4)
Par amigavel: (4, 7)
Par amigavel: (5, 6)
Par amigavel: (6, 12)
Par amigavel: (7, 8)
Par amigavel: (8, 15)
Par amigavel: (9, 13)
Par amigavel: (10, 18)
Par amigavel: (11, 12)
Par amigavel: (13, 14)
Par amigavel: (17, 18)
Par amigavel: (19, 20)
```

Questão 3) Os motoristas se preocupam com o consumo de combustível de seus automóveis. Um motorista manteve um registro do número de abastecimentos que fez, registrando também o número de quilômetros rodados e de litros obtidos a cada abastecimento. Desenvolva um programa na linguagem C que peça o número de quilômetros dirigidos e a quantidade de litros obtidos a cada abastecimento. O programa deverá calcular e exibir a quantidade de quilômetros rodados por litros usados. Depois de processar toda a informação, o programa deverá calcular e exibir o total combinado de quilômetros por litro para todos os abastecimentos. Seu programa deve terminar quando o usuário fornecer o valor de -1 para quantidade de litros abastecidos.

Exemplo de execução:

```
Entre com a quantidade de litros que abasteceu (ou -1 para terminar): 25.6
Entre com a quantidade de quilometros rodados: 287
Consumo atual eh de: 11.210938 km/l
Entre com a quantidade de litros que abasteceu (ou -1 para terminar): 20.6
Entre com a quantidade de quilometros rodados: 200
Consumo atual eh de: 9.708737 km/l
Entre com a quantidade de litros que abasteceu (ou -1 para terminar): 10
Entre com a quantidade de quilometros rodados: 120
Consumo atual eh de: 12.000000 km/l
Entre com a quantidade de litros que abasteceu (ou -1 para terminar): -1
O consumo geral foi de: 10.800712 km/l
```

Questão 4) Um varejista on-line vende cinco produtos diferentes cujos preços de revenda aparecem na tabela a seguir:

Número do produto	Preço de revenda
1	R\$2,98
2	R\$4,50
3	R\$9,98
4	R\$4,49
5	R\$6,87

Escreva um programa em C que leia uma série de pares de números da seguinte forma:

- Número do produto;
- Quantidade vendida durante um dia.

Seu programa deverá usar uma estrutura **switch** para ajudar a determinar o preço de revenda para cada produto e parar de ler pares quando um código de produto inválido for informado. O programa deverá calcular e exibir o valor de revenda total de todos os produtos vendidos na semana anterior. Seu programa deve terminar a execução quando o usuário fornecer um código do produto menor que 1 ou maior que 5.

Exemplo de execução:

```
Qual o código do produto: 1
Informe a quantidade vendida: 3
Qual o código do produto: 5
Informe a quantidade vendida: 4
Qual o código do produto: 7
Total das vendas: R$ 36.42
```

Questão 5) Crie dois arquivos de cabeçalho e defina neles o protótipo das seguintes funções:

- no arquivo "area.h":
 - float area_quadrado(float l);
 - float area_triangulo(float b, float h);
 - float area_circulo(float r);
- no arquivo "volume.h":
 - float volume_cubo(float l);
 - float volume_cone(float b, float h);
 - float volume_esfera(float r);

Crie, em seguida, dois arquivos que implementam as funções especificadas nos arquivos .h

- no arquivo "area.c":
 - área de um quadrado: l^2 , onde l é o lado do quadrado;
 - área de um triângulo: $\frac{b*h}{2}$, onde b é a base e h a altura; e.
 - área de um círculo: $\pi*r^2$, onde r é o raio do círculo.
- no arquivo "volume.c":
 - volume de um cubo: l^3 , onde l é o lado do cubo;
 - volume de um cone: $\frac{\pi*r^2*h}{3}$, onde r é raio da base do cone e h a altura do cone; e
 - volume de uma esfera: $\frac{4.0*\pi*r^3}{3}$, onde r é o raio da esfera.

Crie um quinto arquivo chamado "principal.c" para incluir a função principal. No seu arquivo de código principal, crie apenas a função `main` que fica em um laço interativo perguntando se o usuário quer calcular a

área ou o volume e de que elemento. As leituras devem acontecer pelo menos 1 vez até que o usuário forneça o valor -1 para sair. Use *do-while*, preferencialmente, e pode escolher entre o uso de *if-else* ou *switch*.