

**Professora:** Juliana de Santi (DAINF - Sala 09)

**E-mail:** jsanti@utfpr.edu.br

### Lista - revisão

1) Todo equipamento elétrico possui uma potência apresentada em Watts (W). Para calcular o consumo de um equipamento multiplica-se sua potência pelo tempo de funcionamento em horas, ou seja, Wh (Watts hora). Por exemplo, para um chuveiro com potência de 3500W e que funciona durante 2 horas por dia o consumo é dado por  $3500 \times 2 = 7000$  Wh/dia. Para transformar este valor em kW (quilo Watts hora), dividimos o valor por 1000. Para o exemplo anterior o consumo de 7000Wh/dia = 7kW/dia. Para saber o valor a ser pago pelo consumo do equipamento multiplicamos o valor em kWh pelo valor da tarifa vigente. Por exemplo, considerando o valor da tarifa residencial convencional<sup>1</sup> de R\$0,76897, o custo de utilização do chuveiro no mês é R\$161,48. Considere os seguintes eletrodomésticos e suas respectivas potências<sup>2</sup>:

- 1) Forno elétrico: 9120W
- 2) Cafeteira: 1000W
- 3) Geladeira: 130W
- 4) Lava louça: 1500W
- 5) Liquidificador: 300W
- 6) Microondas: 1200W
- 7) TV: 110W
- 8) Exaustor: 170W

Escreva um simulador que calcula e imprime o consumo (em kW) e o valor mensal a ser pago (considerando a tarifa básica) por um cliente de uma operadora de energia elétrica. Seu programa deve permitir que o usuário escolha (com base em menu) os eletrodomésticos utilizados diariamente e os respectivos tempos (em horas) de uso diário. O comando de saída do programa é a entrada do valor '0' (zero) no menu. Este programa deve conter as seguintes funções (estes protótipos/assinaturas devem ser utilizados):

```
void mostraMenu ();  
float retornaPotencia (int opcaoLida);  
float leTempoUsoHoras ();  
float calcula_consumo_mensal_por_equipamento_kw (float potencia, float tempo);  
float calcula_custo_mensal_por_equipamento (float consumoPorEquipamento, float tarifa);
```

---

<sup>1</sup>[www.copel.com](http://www.copel.com)

<sup>2</sup><http://www.copel.com/hpcopel/simulador/>

Escreva a função `main` com as devidas chamadas.

2) Um número é dito forte (*strong number*) quando a soma do fatorial de cada um de seus dígitos é igual ao número. Por exemplo: 145 é forte já que  $1! + 4! + 5! = 145$ . Escreva a função *strong* que recebe como parâmetro um número inteiro  $k$  e retornar 1 (um) se o número é forte; 0 (zero) caso contrário. Escreva uma função que dado um número inteiro como parâmetro retorna 1 se o número é forte; 0 caso contrário.

Note que o fatorial deve ser calculado por uma função específica para isso.

Escreva um programa (função `main`) que leia um número inteiro  $n$  e imprima todos os números fortes entre 1 e  $n$ .

3) Considere um vetor com os seguintes valores iniciais:

1, 2, 3, 4, 5, 6, 7, 8, 9, 10. Rotacioná-lo à esquerda 3 vezes leva às seguintes configurações:

Primeira rotação à esquerda:

2, 3, 4, 5, 6, 7, 8, 9, 10, 1

Segunda rotação à esquerda:

3, 4, 5, 6, 7, 8, 9, 10, 1, 2

Terceira rotação à esquerda:

4, 5, 6, 7, 8, 9, 10, 1, 2, 3

Escreva a função *rotacionaEsquerda* que recebe como parâmetros um vetor de inteiros e o tamanho do vetor *size*. *void* é o tipo retornado por esta função. Sua função deve rotacionar à esquerda 1 vez o vetor recebido.

Escreva também as funções com os seguintes protótipos/assinaturas:

```
void preencheVetor(int vet[], int size);
```

```
void printVetor (int vet[], int size);
```

Escreva a função principal responsável por preencher um vetor de tamanho *size* ( $size > 5$ , definido pelo usuário) e garantir  $n$  rotações ( $n \geq 3$ , definido pelo usuário). Após cada rotação realizada o vetor resultante deve ser impresso.

4) A primeira fase do campeonato de futebol de Soccer City é composto por 32 rodadas (32 jogos) disputadas entre 16 times. Na tabela do campeonato, a cada vitória uma equipe conquista 3 pontos; a cada empate 1 ponto; e, em caso de derrota, a pontuação é zero. A pontuação para cada uma das 32 rodadas (para cada uma das equipes) deve ser gerada de forma aleatória. Terminada a primeira fase do campeonato (ou seja, após 32 rodadas), seu programa deve (necessariamente) ordenar a tabela de forma que a equipe com a maior

pontuação esteja na primeira posição da tabela, a segunda equipe com maior pontuação esteja na segunda posição de tabela, e assim sucessivamente. Na sequência do campeonato, as 4 equipes com a melhor pontuação disputam, em partida única (confronto direto), as semi-finais do campeonato da seguinte forma: 1ºX4º; 2ºX3º. O vencedor de cada um destes confrontos (são dois confrontos) é determinado através de sorteio (geração dos pontos 0, 1 ou 3). Em caso de empate (ou seja, o valor sorteado é 1 a equipe com a melhor pontuação na primeira fase passa para a fase final. A fase final é, também, disputada em partida única (confronto direto) e em caso de empate a equipe com a melhor pontuação na primeira fase tem a vantagem do empate. No final seu programa deve: 1) imprimir a pontuação do campeão e sua classificação na primeira fase; 2) imprimir a pontuação do vice-campeão e sua classificação na primeira fase; 3) imprimir a pontuação dos 2 rebaixados (ou seja, as duas equipes com a menor pontuação na primeira fase). Seu programa deve conter:

- uma função que simula a primeira fase do campeonato;
- uma função que ordena a tabela de classificação das equipes;
- uma função que simula os confrontos diretos e retorna o vencedor do confronto em questão.

Obs: seu programa **não deve usar vetor auxiliar** para resolver a questão.

**5) (Palavras ordenadas - Baseado em UFBA-2019)** Palavras como “belo”, “fino”, “cruz” e “flor” possuem a interessante propriedade de que suas letras aparecem todas em ordem alfabética. Por exemplo, em “flor”, f vem antes de l no alfabeto, l vem antes de o, e o vem antes de r. Dizemos que palavras com essa propriedade são ordenadas.

Escreva uma função que recebe como parâmetro uma palavra (string) e retorna 1 caso ela seja ordenada; caso contrário, retorna 0. Esta função deve ser chamada tantas vezes quanto o usuário desejar: a leitura será finalizada quando a palavra “stop” for digitada. Escreva a função *critérioParada* que recebe uma string e retorna 1, caso a leitura deva ser finalizada; caso contrário, retorna 0;

Cada palavra é composta apenas por letras minúsculas ou maiúsculas, sem acentos ou pontuação. Suponha que não há palavras com mais de 42 caracteres.

Notas: A biblioteca *string.h* não deve ser utilizada neste exercício.

**6)** Faça um programa que leia um número inteiro e calcule o menor número de notas (dinheiro) no qual o valor pode ser decomposto. Os possíveis valores de notas são: 100, 50, 20, 10, 5, 2, 1. O programa deve imprimir na saída o valor lido e a quantidade e valor de cada nota utilizada. Exemplo: para R\$23,00 temos: 1 nota de R\$20,00, 1 nota de R\$2,00, e 1 nota de R\$1,00. Considere somente valores inteiros.

7) Cidades podem ser mapeadas por regiões (por exemplo, Curitiba é mapeada em 10 regionais<sup>3</sup>) e informações podem ser capturadas e tratadas em função destas regiões.

Você é o responsável por escrever o programa para mapear e gerar estatísticas dos crimes nas regiões de uma cidade. Para tal, são registrados 2 vezes por semana o número de crimes em cada região, sendo que estes 2 valores são registrados separadamente.

O mapa de crimes é dividido em  $n \times n$  regiões (por exemplo, se  $n = 3$ , então são 9 regiões a serem consideradas), sendo que para cada região, a cada registro da semana, é atribuído um número inteiro indicando a quantidade acumulada de crimes para àquela região/registro. A quantidade total de crimes em cada região na semana é dada pela soma dos crimes dos registros da semana. Assim, o mapa de crimes de entrada é definido por  $2 \times n$  linhas, onde as primeiras  $n$  linhas mapeiam o primeiro registro de crimes da semana para cada região; as  $n$  linhas seguintes, mapeiam o segundo registro de crimes da semana para cada região. Exemplo: mapa de crimes de 3x3 regiões (ou seja, 9 regiões) em 2 registros semanais:

```
10 11 12
23 24 25
16 17 18
30 31 32
43 44 45
56 57 88
```

O mapa de saída deverá conter  $n$  linhas, com  $n$  inteiros em cada linha, indicando a quantidade total acumulada de crimes em cada uma das regiões considerando os 2 registros da semana. Para o exemplo anterior, o mapa de saída seria:

```
40 42 44
46 68 70
72 74 106
```

Escreva a função *geraMapEntrada* que tem como parâmetros  $n$  ( $2 \leq n \leq 20$ ) e o mapa de entrada. O retorno desta função é do tipo *void*.

Escreva a função *geraMapSaida* que tem como parâmetros  $n$ , o mapa de entrada e o mapa de saída. O retorno desta função é do tipo *void*.

Escreva a função *printMap* com o seguinte protótipo/assinatura:

```
void printMap(int lin, int col, int map[][col])
```

Escreva a função *estatisticas* que retorna o total de crimes na cidade, ou seja, em todas as regiões durante uma semana. Esta função tem como parâmetros:  $n$ ; o mapa de saída; um ponteiro *pMaior*, que retorna a maior quantidade de crimes durante uma semana; um ponteiro *pMenor*, que retorna a menor quantidade de crimes durante uma semana; em passagem por referência, *pMedia* retorna a média de crimes da semana. Obs: os ponteiros devem, necessariamente, ser utilizados dentro da função, ou seja, variáveis auxiliares não devem ser usadas e depois seus valores atribuídos aos ponteiros.

---

<sup>3</sup><https://www.curitiba.pr.gov.br/conteudo/o-que-sao-regionais/80>

Escreva a função principal que faz as devidas chamadas para as funções escritas e imprime: o mapa de entrada, o mapa de saída, e as estatísticas geradas.

8) Maria irá abrir um pequeno negócio onde serão servidos, doces, salgados e bebidas quentes (café, café com leite, capuccino e chocolate quente). Para determinar o valor a ser cobrado por cada produto ela precisa, primeiramente, determinar o preço de custo deste produtos. Para isso, são considerados as unidades/medidas de ingredientes de cada produto:

		café		leite		chocolate		açúcar
-----								
Café		2		0		0		1
Café com leite		2		2		0		1
Capuccino		2		3		2		2
Chocolate quente		0		6		4		2
-----								

e o custo (preço base) destes ingredientes:

		Preço base (R\$)
-----		
café		1,8
leite		1,7
chocolate		2,3
açúcar		0,5

Levando em consideração as duas tabelas (matrizes) apresentadas, escreva um programa que calcula e imprime o preço de custo de cada bebida servida por Maria.