

INTRODUÇÃO À PROGRAMAÇÃO

LISTA DE EXERCÍCIOS 14

1. Escreva um programa para ler do arquivo “pauta.txt” a matricula, o nome (no máximo 80 caracteres) e indicação de presenças ou faltas (‘F’, ‘f’, ‘V’, ‘v’) em 10 dias de cada um dos alunos de uma turma (no máximo 40 alunos).

Em seguida o programa deve gravar no arquivo “pauta_final.txt” a matricula, o nome e o total de faltas de cada aluno.

Após ler as informações e gerar o arquivo de pauta final, o programa deve permitir que o usuário faça consultas digitando a matricula do aluno, e se a mesma existir, o programa deve fornecer o nome do aluno e o total de presenças. As consultas terminam quando o usuário digitar uma matricula igual a 0.

Exemplo de conteúdo do arquivo “pauta.txt”:

```
111
Ana Maria
PpppPPpPpP
222
Jose Augusto
PfpfPFPFP
333
Pedro Augusto
FPfpPfffF
444
Vanessa Carla
PPffPpFPPf
555
Joao Vinicius
PPppfFPPfF
```

2. Uma empresa realizou uma pesquisa de satisfação de 7 clientes de operadoras de celular para contabilizar quantos clientes estão satisfeitos com sua própria operadora.

Para isso, na coleta de dados o cliente deu uma nota a todas as 4 operadoras pesquisadas. Estes dados, isto é, as 8 notas atribuídas às operadoras por cada cliente, estão armazenadas no arquivo “Notasdadas.txt”. Portanto, este arquivo tem 7 linhas, cada linha com 8 notas.

O arquivo “entrevistasOper.txt” armazena nome da operadora dos 7 entrevistados, um por linha (na 1ª está o nome da operadora do primeiro entrevistado, na 2ª linha do segundo entrevistado, etc..). O nome das operadoras têm no máximo 30 caracteres.

O arquivo “codoperadoras.txt” armazena o nome da operadora e seu código (número entre 1 e 8). Há quatro operadoras de celular sendo pesquisadas. O nome da operadora está em uma linha e o código desta operadora na linha seguinte.

Faça um programa que mostre na tela a quantidade de pessoas satisfeitas por operadora. Uma pessoa está satisfeita quando atribui uma nota a sua operadora maior ou igual à nota média da operadora. A nota média de uma operadora é calculada pelo somatório das notas recebidas por ela dividida pelo número de entrevistados

Exemplo de conteúdo dos arquivos:

“entrevistasOper.txt”		“Notasdadas.txt”					“codoperadoras.txt”	
		<i>Operadoras</i>					<i>Operadora</i>	<i>Cód</i>
<i>N</i>	<i>Operadora</i>	<i>N</i>	<i>1</i>	<i>2</i>	<i>3</i>	<i>4</i>		
1	Vivo	1	3.0	3.0	3.0	10.0	Vivo	1
2	Oi	2	2.0	3.0	1.0	10.0	Oi	4
3	Oi	3	1.0	6.0	7.0	1.0	Claro	2
4	Claro	4	6.0	9.0	8.0	9.0	Tim	3
5	Vivo	5	9.0	8.0	2.0	7.0		
6	Tim	6	8.0	8.0	8.0	8.0		
7	Vivo	7	7.0	3.0	3.0	3.0		

Para estes exemplos, o total de clientes satisfeitos por operadora é:

Vivo	Claro	Tim	Oi
2	1	1	1

3. Uma prova de corrida de Fórmula IPRJ é realizada da seguinte forma:
- No máximo, 18 equipes participam da prova;
 - O roteiro é dividido em 6 trechos e cada trecho é cronometrado (6 tempos parciais por equipe);
 - Cada trecho terá uma equipe campeã: será a que obtiver o menor tempo de percurso naquele trecho;
 - O tempo total de uma equipe é a soma dos tempos dos 6 trechos.

Faça um programa que, após contabilizar os tempos de todas as equipes, imprima para cada trecho:

- O nome da equipe campeã;
- O ano de criação da equipe;
- O tempo no trecho (em segundos);
- O tempo total da equipe no formato horas:minutos:segundos.

As informações sobre a corrida estão armazenadas nos arquivos EQUIPES.TXT e TEMPOS.TXT, descritos a seguir:

- arquivo texto EQUIPES.TXT, contendo as seguintes informações (uma por linha):
 - ano de criação da equipe (inteiro);
 - nome da equipe (máximo de 15 caracteres válidos);
- arquivo TEMPOS.TXT, contendo as seguintes informações (uma por linha):
 - quantidade de tempos armazenada no arquivo (inteiro);
 - nome da equipe (máximo de 15 caracteres válidos);
 - tempos (em segundos) de cada trecho da prova (inteiros) em ordem: tempo do 1º trecho, tempo do 2º trecho, tempo do 3º trecho, e assim sucessivamente.

Observações:

- A equipe que possuir menos de 6 tempos armazenados no arquivo TEMPOS.TXT, receberá a seguinte punição: para cada tempo não armazenado no arquivo, deverá ser considerado o tempo de 9999 segundos;
- Assuma que nenhuma equipe terá mais de 6 tempos no arquivo TEMPOS.TXT

O seu programa deve implementar e utilizar as seguintes funções feitas por você:

- função `busca()`: implementa o algoritmo de busca sequencial;
- função `le_arq_equipes()`: faz a leitura dos dados do arquivo EQUIPES.TXT, preenchendo os vetores recebidos como parâmetros; retorna a quantidade de registros lidos;

- função `le_arq_tempos()`: uma de suas responsabilidades é fazer a leitura dos dados do arquivo TEMPOS.TXT, preenchendo os parâmetros recebidos;
- função `converte()`: recebe um inteiro representando um valor em segundos e o devolve separado em horas, minutos e segundos.

Exemplo de resultado esperado na tela:

```
Equipe campea do trecho 1: OLHO-DE-BOI
Ano da criacao: 1930
Tempo no trecho atual: 101    Tempo total da equipe: 0:44:20
```

```
Equipe campea do trecho 2: OLHO-DE-BOI
Ano da criacao: 1930
Tempo no trecho atual: 101    Tempo total da equipe: 0:44:20
```

```
Equipe campea do trecho 3: SAFIRA
Ano da criacao: 1960
Tempo no trecho atual: 419    Tempo total da equipe: 0:50:8
```

```
Equipe campea do trecho 4: OLHO-DE-BOI
Ano da criacao: 1930
Tempo no trecho atual: 512    Tempo total da equipe: 0:44:20
```

```
Equipe campea do trecho 5: JADE
Ano da criacao: 1910
Tempo no trecho atual: 612    Tempo total da equipe: 0:51:16
```

```
Equipe campea do trecho 6: AGATA
Ano da criacao: 1890
Tempo no trecho atual: 911    Tempo total da equipe: 0:51:19
```

4. Uma instituição oferece cursos em 3 turnos: M - Manhã, T - Tarde, N - Noite e os alunos inscritos em disciplinas EaD, comuns aos cursos, devem fazer a prova presencial, realizada em julho, no turno de seu curso. O comitê de organização da prova e alocação das salas de aulas, precisa saber quantos alunos realizarão as provas em cada um dos turnos.

O arquivo **disciplinas.txt** armazena a quantidade de alunos nas (no máximo 15) diferentes disciplinas EaD, do seguinte modo:

- Disciplina (string de, no máximo, 10 caracteres válidos);
- Quantidade de alunos inscritos na disciplina em cada um dos 3 turnos (3 inteiros)

Exemplo do arquivo disciplinas.txt:

```
LPO001
400 200 1000
```

Significado: A disciplina LPO001 tem 400 alunos inscritos no turno manhã, 200 alunos inscritos no turno tarde e 1000 alunos inscritos no turno noite.

No início do mês de junho, os alunos inscritos em alguma disciplina EaD podem solicitar a troca do turno para a realização da prova.

O arquivo texto **solicitacoes.txt** armazena, em pares de linhas, a disciplina, a matrícula do aluno, o turno origem e o turno destino:

- Disciplina (string de, no máximo, 10 caracteres válidos);
- Matrícula do aluno (inteiro) turno de origem (M ou T ou N) turno destino(M ou T ou N)

Exemplo do arquivo solicitacoes.txt:

```
LPO001
102030 M T
```

Significado: O aluno de matrícula 102030, matriculado no turno Manhã, deseja realizar a prova de LPO001 no turno Tarde.

Faça um programa **MODULARIZADO** que, utilizando as informações armazenadas nos 2 arquivos, gere o arquivo **ATUAL.TXT** com a quantidade de alunos que devem realizar as provas em cada um dos turnos em cada um das disciplinas.

Obrigatoriamente, o seu programa deve utilizar as seguintes funções feitas por você:

- a) **função busca():** implementa o algoritmo de busca sequencial;
- b) **função le_arq_disciplinas():** realiza a leitura dos dados do arquivo disciplinas.txt, preenchendo os parâmetros recebidos;
- c) **função atualiza_situacao:** uma de suas responsabilidades é fazer a leitura dos dados do arquivo solicitacoes.txt, efetivando as operações realizadas;
- d) **função gera_arq_atual():** cria o arquivo **ATUAL.TXT** com as informações atualizadas;
- e) **função descobre_pos_turnos():** a função recebe o caractere que representa o turno origem e o caractere que representa o turno destino e dois endereços de memória, onde a função irá armazenar o número correspondente ao turno origem e o correspondente ao turno destino, isto é, 0 se for manhã, 1, se for tarde e 2 se for noite.

EXEMPLO DE ARQUIVOS E RESULTADO ESPERADO:

disciplinas.txt	solicitacoes.txt	Atual.txt
LPO001	LPO001	LPO001 99 199 302
100 200 300	102030 M T	LPO002 200 100 300
LPO002	EMP001	EMP001 0 3 3
200 100 300	102030 M T	EMP002 4 4 4
EMP001	LPO001	
1 2 3	203010 T M	
EMP002	EMP002	
3 4 5	203010 N M	
	LPO002	
	102323 N T	
	LPO001	
	103443 M N	
	LPO002	
	102124 T N	
	LPO001	
	102124 T N	