

INF1005 - PROGRAMAÇÃO 1

LISTA DE EXERCÍCIOS 16

1. 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	

2. A eleição para presidente da Ilha de Prog1 terminou ontem à noite, sem nenhum incidente. Tanto os eleitores que moram na ilha quanto os que moram no exterior já votaram em um dos candidatos à presidência (máximo 30 candidatos). Também foi possível que o eleitor anulasse seu voto digitando na urna eletrônica o nome de um candidato inválido.

As informações sobre a eleição estão distribuídas em 2 arquivos:

- **CANDIDATOS.TXT**: arquivo texto com os nomes e partidos dos candidatos à presidência (uma informação por linha):
 - Nome do candidato (máximo de 20 caracteres válidos);
 - Partido do candidato (máximo de 20 caracteres válidos).
- **VOTOS.TXT**: arquivo texto com os votos dos eleitores; uma linha para cada eleitor, contendo:
 - Local da residência do eleitor: valor 0 (se é morador da ilha) ou valor 1 (se mora no exterior);
 - Nome do candidato.

Cada voto de um eleitor que mora na ilha vale integralmente para o seu candidato. Mas o total de votos recebido de eleitores que moram no exterior não pode ultrapassar 10% do total de votos recebidos de eleitores que moram na ilha. Ou seja, se um candidato recebeu 100 votos locais e 35 votos do exterior, o total de votos válidos desse candidato é 110 (100 + 10).

Faça um programa que gere um relatório (arquivo **RELATORIO.TXT**) baseado nas informações dos arquivos fornecidos de acordo com a saída esperada (vide seção RESULTADO ESPERADO). Para facilitar, assumo que não haverá empate no resultado.

O seu programa deverá utilizar as seguintes funções feitas por você:

- a) **função le_dados_candidatos()**: recebe dois vetores de strings como parâmetro e preenche os vetores com os nomes e partidos lidos do arquivo CANDIDATOS.TXT. A função também deve retornar a quantidade de candidatos lidos;
- b) **função le_votos()**: recebe a matriz que armazena a quantidades de votos como parâmetro e a preenche com as informações de votos presentes no arquivo VOTOS.TXT;
- c) **função gera_relatorio()**: recebe, como parâmetros, todas as informações necessárias para gerar o relatório conforme seção RESULTADO ESPERADO;
- d) **função descobre_eleito()**: recebe a matriz com as quantidades de votos como parâmetro e retorna o número da linha da matriz referente ao candidato eleito.

ARQUIVO CANDIDATOS.TXT:

<http://www.inf.puc-rio.br/~elima/prog1/Candidatos.txt>

ARQUIVO VOTOS.TXT:

<http://www.inf.puc-rio.br/~elima/prog1/Votos.txt>

RESULTADO ESPERADO (ARQUIVO RELATORIO.TXT):

Nome	Partido	Votos locais	Votos exterior	Votos exterior computados	Total geral
CLARABELA	DISNEY	1	0	0	1
GASTAO	CARTOON	7	0	0	7
MARGARIDA	DISCOVERY	1	3	0	1
MICKEY MOUSE	DISNEY	14	3	1	15
PATETA	DISNEY	2	0	0	2
JOAO BAFO DE ONCA	DISCOVERY	0	0	0	0
ZEZINHO	DISNEY	1	0	0	1
LUIZINHO	FOX	1	0	0	1
HUGUINHO	DISCOVERY	1	0	0	1
PLUTO	CARTOON	0	0	0	0
PROFESSOR PARDAL	FOX	9	4	0	9
MINNIE	DISNEY	8	0	0	8
ZE CARIOCA	DISCOVERY	87	8	8	95
TECO	CARTOON	0	1	0	0
TICO	CARTOON	0	1	0	0
PATO DONALD	FOX	5	2	0	5
PENINHA	DISNEY	4	0	0	4
TIO PATINHAS	DISCOVERY	1	0	0	1
VOVO DONALDA	CARTOON	15	0	0	15

*** PRESIDENTE ELEITO: ZE CARIOCA DO PARTIDO DISCOVERY ***