

## ESTRUTURAS DE DADOS

## Exercícios de Fixação de Arquivos Texto

- 1. Ao receber um arquivo de texto, via parâmetro, retorne um **novo arquivo** com todas as letras em maiúsculo.
- 2. Vasculhe um arquivo texto qualquer e retorne a quantidade de letras, espaços e linhas existentes no mesmo.
- 3. Ao receber 2 arquivos texto, via parâmetro, compará-los e retornar se são idênticos ou não.
- 4. Recebendo um arquivo texto, via parâmetro, faça com que seja criado um novo arquivo texto, com todas as primeiras letras de cada palavra em maiúsculo.
- 5. Gere um **Arquivo Texto Formatado** que contenha em cada linha: número da aposta (sequencial), data da aposta e os 7 (sete) números apostados obrigatoriamente.

	Apostas.txt								
1	10/08/2012	5	15	8	22	43	53	32	
2	10/08/2012	10	23	43	15	16	9	44	
3	11/08/2012	59	32	7	8	21	29	31	
4	11/08/2012	23	43	15	32	7	8	1	
:	:								

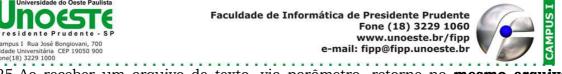
- 6. Implemente uma nova função que ao receber o arquivo "Apostas.txt" e as 5 (cinco) dezenas sorteadas, retorne as apostas que foram premiadas com o terno (3 acertos), quadra (4 acertos) e quina (5 acertos).
- 7. Gere uma espécie de criptografia, que ao receber um arquivo texto, retorne em outro arquivo o texto codificado. Invente sua técnica!
- 8. Faça o processo inverso ao item 6, ou seja, "descriptografe" o arquivo gerado anteriormente.
- 9. Ao receber um arquivo texto, retorne a quantidade de palavras que são iniciadas por uma vogal.
- 10. Faça um programa que escreva em disco uma contagem que vá de 1 até 1000, com um número em cada linha e o seu nome após o número 1000. Após, exiba o arquivo.
- 11. Faça um programa que leia um arquivo texto, lendo linha a linha, e exibindo as linhas numeradas na tela. A ideia é possibilitar a visualização de um arquivo de programa em linguagem C/CPP e mostrar na tela com as linhas numeradas.
- 12. Ao receber um arquivo texto e uma letra, retorne a linha que mais tem essa letra.



## Faculdade de Informática de Presidente Prudente Fone (18) 3229 1060 www.unoeste.br/fipp e-mail: fipp@fipp.unoeste.br

- 13. Ao receber um arquivo texto e uma palavra, retorne a linha que mais tem essa palavra.
- 14. Faça uma função em C que receba um Arquivo Texto (via parâmetro) e retorne um novo com as vogais substituídas por "@".
- 15. Ao receber um arquivo texto, uma 'letra' e um 'caractere especial', todos via parâmetro, retorne um novo arquivo com as devidas substituições dos caracteres iguais a 'letra' pelo 'caractere especial' escolhido.
- 16. Implemente uma função em C que seja capaz de verificar se um arquivo ".cpp" possui inconsistências em relação aos parênteses existentes. Ou seja, para cada parênteses aberto deve ter um fechado.
- 17. Faça um programa que imite um editor de textos. Inicialmente você irá criar um vetor em memória onde será armazenado o texto escrito pelo usuário (texto de 1 até no máximo 50 linhas). O usuário vai escrever um texto, terminando por uma linha onde ele irá escrever apenas a palavra 'FIM'. Salvar o conteúdo deste vetor que está na memória, em um arquivo em disco.
- 18. Implemente uma função que ao receber um arquivo texto em um idioma qualquer e um arquivo de dados contendo o dicionário desse idioma, seja capaz de retornar o arquivo traduzido. Caso não houver a tradução de determinada palavra a mesma deve permanecer.
- 19. Faça um programa do tipo "*grep*". Dado um arquivo texto e uma palavra qualquer, este programa deve procurar esta palavra no arquivo texto e devolver a linha onde esta palavra foi encontrada.
- 20. Faça um programa do tipo "sed ou awk" (Unix). Dado um arquivo texto e duas palavras (original e substituída), substitua todas as ocorrências da palavra original colocando no seu lugar a palavra a ser substituída. O texto substituído deve ser retornado em um novo arquivo texto.
- 21. Faça um programa que ao receber um arquivo texto (extensão PAS) por parâmetro, seja capaz de validar um "programa Pascal" em relação aos comandos "begin" e "end". Ou seja, para cada "begin" deve haver um "end".
- 22. Faça um programa que leia via teclado, dados referentes a alunos, gravando em um **arquivo texto formatado** (fprintf) o nome, nota1, nota2, porcentagem de faltas e calcule a média aritmética simples a partir das médias fornecidas.
- 23. Faça um programa que leia os dados gerados pelo programa anterior e mostre na tela as informações referentes aos alunos em relação à aprovação (média>=6 e frequência>=75%) ou não, e ainda, calculando a média geral da turma ao final.
- 24.Implemente um programa que possibilite a visualização de arquivos texto, independente do tamanho do mesmo, possibilitando ao usuário a utilização das teclas:
  - $\square$  Home  $\rightarrow$  início do arquivo;
  - $\boxtimes$  End  $\rightarrow$  fim do arquivo;
  - ☑ Page Up → uma página acima;
  - ☑ Page Down → uma página abaixo.





- 25. Ao receber um arquivo de texto, via parâmetro, retorne no mesmo arquivo com todas as letras em maiúsculo.
- 26. Recebendo um arquivo texto, atualize-o modificando todas as primeiras letras de cada palavra para maiúsculo.
- 27. Gere uma espécie de criptografia, que ao receber um arquivo texto, retorne-o codificado. Invente sua técnica!
- 28. Faça uma função em C que receba um Arquivo Texto (via parâmetro) e retorne-o com as vogais substituídas por "@".