

1ª AVALIAÇÃO

| | | |
|--|------------------------|----------------|
| DISCIPLINA: PROG. MULTIPLATAFORMA | ANO/SÉRIE: 3º | TURMA: |
| PROFESSOR(A): LEONARDO CABRAL DA ROCHA SOARES | | |
| NOME: | | |
| DATA: | VALOR: 8 PONTOS | TIPO: A |
| | PONTO EXTRA: | NOTA: |

INSTRUÇÕES

Roteiro de prova prática

- É proibido o empréstimo de material durante a prova;
 - É proibido o uso de qualquer aparelho eletrônico durante a prova;
 - Ao aluno que descumprir as instruções, bem como utilizar ou tentar utilizar meios fraudulentos, será atribuído nota zero.
- BOA PROVA!!!**

Descrição textual do objetivo: Uma empresa especializada em Recuperação da Informação Web precisa de um software para gerenciar ocorrências de termos em documentos. A coleção de documentos foi rotulada com um código inteiro incremental C , onde $1 \leq C < 100$. Os termos que desejamos controlar a ocorrência são **Brutus** e **Caesar** (chamados aqui de Tokens). O Programa deverá permitir o cadastro rápido e eficiente da ocorrência destes Tokens em um dos documentos e emitir relatórios listando em quais documentos ocorrem os termos (A pesquisa para geração dos documentos deve ser feita com o operador lógico AND – quando for o caso).

Interação Homem-Maquina: O programa deverá possuir uma tela de login (controle de acesso), um menu de controle e a interface para cadastro e busca de documentos. A interface para cadastro e recuperação das ocorrências deverá respeitar o layout estipulado pela **[Índice Invertido]**. A tela de login e o menu de controle podem ser desenhados livremente, respeitando-se a correta utilização das classes java. Todas as ações da tela **[menu de controle]** devem ser disponibilizadas através da **barra de menu**.

Paradigma de desenvolvimento: O programa deve ser implementado segundo o paradigma da **Programação Orientada a Objetos**.

Complexidade: Excetuando-se tarefas de ordenação de vetores (ou matrizes) admite-se como aceitável uma complexidade assintótica de **$O(n)$** para tempo e espaço. Para rotinas de ordenação aceita-se **$O(n^2)$** .

Critérios de avaliação:

- * **Desenvolvimento de interface gráfica – 2pts**
- * **Navegação entre telas – 2pts – Inclui-se aqui o controle de acesso limitado por usuário e senha**
- * **Implementação do objetivo principal – 4pts – Desde que respeitados o paradigma de desenvolvimento e a complexidade assintótica, qualquer implementação que atenda corretamente ao objetivo será aceita.**

Penalidades: Erros em tempo de execução, durante a correção, não tratados com mensagens amigáveis, penalizam a nota em -1 (cada).

Prazos e condições: O tempo para o desenvolvimento da aplicação (prova) será de 3.5 horas aulas – retirando-se destas o tempo necessário para explicação inicial por parte do docente. Os 25 minutos finais da quarta aula (0.5 hora aula) serão reservados para correção. Nenhum discente poderá fazer alterações no projeto. Atrasos por parte dos discentes não resultarão em acréscimo no tempo.

Não é permitida a interação entre discentes durante o desenvolvimento. Fora do ambiente de desenvolvimento (laboratório da escola) é permitida comunicação e pesquisa – não será permitida a consulta de nenhum tipo de material durante o desenvolvimento.

Indice invertido

Controle de Ocorrências

N.Doc Adicionar

Token Encontrado

Brutus Caesar

Relatórios

Token Brutus Caesar

Listar Documentos

Frequência de termos

Brutus

Caesar

Illustration 1: Indice invertido

Fonte: Criada pelo autor