

Trabalho sobre Métodos de Pesquisa e Ordenação e Linguagem de Programação Orientada a Objetos em Java – Primeira Etapa

Valor

Segunda Etapa: 5,0 pontos na C3

Objetivos

- Avaliar competências e conhecimentos desenvolvidos nas disciplinas de Pesquisa e Ordenação e Linguagem de Programação Orientada a Objetos, por meio de um trabalho experimental.
- Oportunizar o desenvolvimento e escrita de uma atividade nos moldes de um trabalho acadêmico, bem como aperfeiçoar a capacidade de criar, planejar, trabalhar e decidir em grupo de forma cooperativa.
- Aplicar conceitos de POO e de LPOO no desenvolvimento de um programa.
- Aprimorar-se na programação Java.

Obs.: O grupos em que **nenhum dos componentes faz a disciplina LPOO** não precisam cumprir os requisitos de Orientação a Objetos, embora, se cumprirem, a programação será menos exaustiva.

Segunda Etapa

- Implementar uma árvore ABB, uma árvore AVL e um Hashing com vetor Encadeado para armazenar as compras que estão nos arquivos dados pela professora na etapa 1. Essa compras deverão ser armazenadas por CPF, porém o grupo deve pensar em uma estratégia para armazenar compras que tenham o mesmo CPF.
- Data limite para entrega: **28/11** (domingo).
- **O grupo perde 1 ponto por cada aula de atraso na entrega, limitado a duas aulas.**
- Qualquer componente do grupo deve ser capaz de apresentar/defender todas as ideias contidas no trabalho. É fundamental que o trabalho seja feito de forma cooperativa, e não, simplesmente, que cada componente cuide de uma parte do mesmo para depois juntar e formar o todo. As apresentações será realizadas após a entrega da segunda parte.
- O trabalho deve ser postado no AVA, o código, a parte escrita e os arquivos gerados.

Parte Escrita

A parte escrita deverá ser feita de forma bastante sucinta e deverá conter:

- Uma introdução, descrevendo brevemente cada um dos métodos de Pesquisa implementados (ABB, AVL e Hashing).
- O corpo que deve começar descrevendo a máquina em que o programa for rodado e os arquivos utilizados para teste.
- Fazer quadros de comparação entre o tempo de processamento de cada método implementado, para cada arquivo gerado.

- Conclusões geradas a partir da análise dos quadros comparativos de tempo e verificar se elas são compatíveis ou não com a teoria. Se não for compatível, investigue as causas.
- Se for necessário fazer pesquisas em outros materiais, listar os títulos que auxiliaram o desenvolvimento do trabalho, segundo as normas da ABNT

Critérios avaliativos

- Organização do documento impresso e formatação segundo as normas de trabalho acadêmico.
- Utilização adequada de estruturas de dados e métodos de ordenação.
- Programa correto.
- Legibilidade do código e identificação.
- Entrevista individual e/ou coletiva.
- Trabalho em equipe e de forma cooperativa.
- Pontualidade na entrega (a cada dia de atraso, o trabalho valerá 1,0 ponto a menos).

Texto nas normas da FAESA	0,5
Descrição dos métodos e conclusões	0,5
Implementação dos métodos	3,0
Apresentação (essa nota será dada em novembro, depois da apresentação)	1,0
TOTAL	5,0

Problema a ser implementado

Já foram disponibilizados 15 arquivos do tipo texto, contendo 500, 1.000, 5.000, 10.000 e 50.000 registros, dispostos de forma aleatória, ordenada e invertida. Será ainda disponibilizado um arquivo (compra.txt) com 400 CPFs que serão utilizados para a pesquisa, um em cada linha.

O trabalho consiste em:

- 1) Comece a contar o tempo.
- 2) Carregue o arquivo com o arquivo de 500 elementos aleatórios em uma árvore ABB pelo CPF. Você deve prever CPFs iguais e pensar um meio de armazenar todos os CPFs iguais.
- 3) Balanceie a árvore.
- 4) Carregue o arquivo compra.txt em um ArrayList.
- 5) Para cada CPF do arquivo compra.txt, faça uma pesquisa na ABB. Ao final, deve gerar um outro arquivo onde, para cada cpf que for encontrado será gravado os dados de todas as compras daquele CPF. Some todos os valores das compras daquele CPF e mostre no

final o total gasto por aquele cliente. Se o cpf não for encontrado, deve-se gravar uma mensagem NÃO HÁ NENHUMA COMPRA COM O CPF (colocar o cpf).

Veja o exemplo abaixo:

CPF 11122233344 NOME Cinthia Caliri
data: 15/11/2002 valor: R\$ 3215.68
data: 18/01/2010 valor: R\$ 1230.00
Total: 4445.68

CPF 12345612344:
NÃO HÁ NENHUMA COMPRA COM O CPF 12345612344

6) Repita 4 vezes o processo de 2 a 5. **Você deve rodar o processo 5 vezes no total.** Os arquivos gerados podem ser regravados, ou seja, no final você terá apenas um arquivo para cada tamanho, tipo e método.

7) Termine de contar o tempo, faça uma média e armazene este resultado.

8) **Faça os itens de 1 a 7 para cada um dos tamanhos (500, 1000, 5000, 10000 e 50000), para cada tipo de arquivo (aleatório, ordenado e invertido) e para cada método (ABB, AVL, Hashing).** Observe que o balanceamento só será feito com a ABB. Ao todo, o programa rodará 45 vezes e criará 45 arquivos.

9) Compare os tempos de todos os algoritmos em cada tamanho e tipo de arquivo e gere conclusões.

Obs.:

1) Para computar o tempo, utilize o método `System.currentTimeMillis()` que retorna o tempo da máquina em milissegundos, ou `System.nanoTime()` que retorna o tempo da máquina em nanossegundos.