

Atividade Prática 2
Valor: 30% da 2ª Avaliação
Entrega: 03/11/2019 (Individual)

1. DESCRIÇÃO

O problema da interseção de conjuntos consiste em encontrar elementos que pertençam simultaneamente a dois conjuntos A e B. Há diversas formas de resolvê-lo, sendo a mais simples provavelmente aquela em que, para cada um dos elementos de um dos conjuntos, percorre-se possivelmente todo o outro conjunto à medida que seus elementos vão sendo comparados.

Você deverá IMPLEMENTAR um programa que resolver o problema de interseção de conjuntos utilizando lista encadeada, árvore binária, árvore rubro-negra e hashing.

Se programa deverá carregar as chaves que serão lidas via arquivo. O arquivo que tiver menor quantidade de chaves será sempre armazenado em uma lista encadeada (Conjunto A). Já o arquivo com maior quantidade de chaves (Conjunto B) deverá ser carregado em:

1. Lista encadeada
2. Árvore binária
3. Árvore AVL
4. Árvore Rubro-Negra.

Implemente as seguintes operações:

1. Buscar os elementos de A que estão em B
2. Inserir em B, os elementos de A que não estão em B
3. Remover os elementos de A que estão em B

Discuta no relatório os resultados a complexidade e o custo em tempo de execução para realizar as operações utilizando cada uma das 4 estruturas de dados utilizadas.

2. Descrição sobre os arquivos de dados

Serão utilizados arquivos de tamanho pequeno, média e grande.

3. Análise dos resultados

A análise deve ser feita sobre o número de comparações, atribuições e tempo de execução dos algoritmos. Procure organizar inteligentemente os dados coletados em tabelas, e também construa gráficos a partir dos dados. Então, disserte sobre os dados nas tabelas e gráficos. Grande parte da avaliação será dedicada a análise dos resultados, ou seja, sobre o que você dissertar.

4. Entrega

- Código fonte do programa em C/C++ , Java ou Python (bem indentado e comentado).
- Relatório do trabalho
- Upload no SIGAA.

O Relatório deve apresentar:

1. Implementação: descrição sobre a implementação do programa. Devem ser detalhadas as estruturas de dados utilizadas (de preferência com diagramas ilustrativos), o funcionamento das principais funções e procedimentos utilizados, o formato de entrada e saída de dados, bem como decisões tomadas relativas aos casos e detalhes de especificação que porventura estejam omissos no enunciado.
2. Análise de complexidade: apresentar o estudo de complexidade das principais funções implementadas usando a notação O.
3. Testes: apresentação dos testes realizados.
4. Conclusão: comentários gerais sobre o trabalho e as principais dificuldades encontradas em sua implementação
5. Referências: referências utilizadas no desenvolvimento do trabalho.