**UNIVERSIDADE FEDERAL DE JUIZ DE FORA**

**Estrutura de Dados II**

Relatório Referente ao trabalho 2:

Busca de Tweet

- Árvores Splay

- Árvores AVL

- Árvores B

- Árvores Vermelha/Preta

Bruno Carvalho

Diogo Destefano

Pedro Bellotti

Rafael Terra

11 de Novembro de 2017

**Índice**

1. Introdução......................................................................................…..03
2. Dados Brutos..................................................…….......…...................04
3. Análise dos Resultados..........................……..…….…...................…..05
4. Dados Sobre o Desenvolvimento...........…..........................................06

**1 – Introdução**

Desenvolvido em linguagem C++, a segunda parte do projeto tem como objetivo analisar e comparar os diferentes algoritmos de manipulação e organização de dados utilizando-se tipos de Árvores distintos, a fim de se concluir quais métodos se apresentam mais eficientes em determinados contextos.

Variáveis como número de entradas, gasto em memória, tempo de execução, número de comparações e cópias de registro realizadas foram determinantes para definir e comparar os resultados dentre os algoritmos testados.

O projeto foi desenvolvido visando consumir uma quantidade reduzida de memória, visto que por vezes, os algoritmos foram testados com até 1.000.000 entradas. A cada iteração de testes (5 iterações para cada conjunto de dados de tamanho variando entre 1000 e 1.000.000) os resultados foram salvos e as estruturas foram desalocadas, para que o gasto de memória seja controlado. A cada iteração, o conjunto de dados foi *randomizado*, a fim de se obter um resultado mais confiável (complexidade de randomização: O(n)).

O software utiliza o console para controle de execução. Ao executar o código, o menu é impresso em tela, e ao escolher uma opção de criação de Árvore, o software realiza os testes Inserção, Busca e Remoção desta árvore, salvando os resultados nos arquivos TXT respectivos a cada tipo de teste.

**2 – Dados Brutos**

-Análise dos Algoritmos

As tabelas a seguir contém os dados obtidos na Inserção dos TweetID`s em cada tipo diferente de Árvore (B, Splay e Vermelho-Preto). Métricas como Tempo Gasto, Número de Comparações e Cópias de Registo foram utilizadas. Lembrando que tais valores foram obtidos através da média entre 5 execuções para cada N (onde N é o tamanho do conjunto de dados).

Os arquivos de saída com os resultados dos testes estão anexados ao trabalho.







**3 – Análises dos Resultados**

**3.1. Árvore Vermelho-Preta**

Dentre os algoritmos analisados, a Árvore Vermelho-Preta foi o método que obteve os piores resultados. Partindo do tempo de execução, no pior caso (1.000.000), houve uma demora de aproximadamente 44 segundos na execução, enquanto nos outros 2 métodos testados, tal teste utilizou menos de 10 segundos. O número de comparações foi significativamente maior que os outros métodos, porém o número de cópias de registro foi menor que as Árvores B e Splay no pior caso(1.000.000), e maior nos demais conjuntos de dados (entre 1000 e 500.000).

Acreditamos que a grande diferença no tempo de execução se deve ao fato de que a cada inserção, a Árvore deve ser reajustada, as cores de cada nó são verificadas e mudanças pertinentes são feitas, a fim de se adequar aos diversos casos característicos deste tipo de Árvore.

**3.2. Árvore B**

Comparado a Árvore Vermelho-Preta, a Árvore B mostrou melhoras significativas no tempo de execução e em número de comparações, porém, não houve grande diferença no número de cópias de Registro. O número de comparações e tempo de execução obtiveram um crescimento linear entre os casos analisados, porém o número de cópias de registro obteve aumento cada vez menor a medida que o conjunto de dados aumentou.

**3.1. Árvore Splay**

A Árvore Splay foi o algoritmo que obteve os melhores resultados gerais. Houve uma grande melhora em tempo de execução, sendo este, o algoritmo mais rápido dentre os algoritmos testados. Quanto ao número de comparações e cópias de registro, o algoritmo obteve os melhores resultados em métricas menores que 500.000 dados. Como tais números cresceram significativamente em cada aumento de dados, o algoritmo obteve baixa eficiência com o conjunto de 1.000.000 quando comparado aos outros métodos, obtendo o número de cópias de registro mais elevados em alguns casos.

**4 – Dados Sobre o Desenvolvimento**

**4.1 – Hardware e Software Utilizado**

O projeto foi executado em um computador com Core i7 de 3.5GHz, com 16GB de memória RAM, em um sistema operacional Windows 10 Pro de 64bits utilizando a IDE Visual Studio (em linguagem C++). Para a gestão e controle do projeto entre o grupo, a plataforma GitHub foi utilizada.

**4.2 – Divisão de tarefas entre o Grupo**

Bruno Carvalho: Árvore Vermelho e Preta, funções gerais do main (como criação de testes de inserção, busca e remoção, bem como armazenamento de resultados em arquivos TXT). Aplicação dos testes e extração dos dados de saída. Auxílio na produção do relatório.

Diogo Destefano: Árvore B.

Rafael Terra: Árvore AVL.

Pedro Bellotti: Adaptações de funções no main, Árvore Splay, criação de tabelas de resultados, auxílio na produção o relatório.