

Tutoria ED2 - Semana 4

Aluno: Matheus Peixoto Ribeiro Vieira

Matrícula: 22.1.4104

1 -

```
C/C++
void Minimo(TipoRegistro * x, TipoApontador Ap){
    if(Ap->p[0] != NULL)
        Minimo(x, Ap->p[0]);

    else
        x = &(Ap->r[0]);
}
```

2 - Em uma árvore B, quando o item é adicionado, ele vai até a página que deve pertencer e é inserido lá. Todavia, caso a página esteja cheia, a página é dividida em duas e o item do meio passa para a página pai, mudando, assim, a estrutura da árvore de baixo para cima.

Já em uma árvore binária, quando o item é adicionado, ele sempre irá para a parte mais baixa da árvore, assim, o tamanho dela irá aumentar de cima para baixo.

3 - Suponha a necessidade de fazer a pesquisa em um dado que está em um arquivo de tamanho de 20 Gb. Suponha, também, que o seu computador possua somente 8 GB de RAM. Dessa forma, ao abrir o arquivo não será possível carregá-lo completamente de uma só vez, pois o seu tamanho supera o que está disponível em memória principal.

Dessa forma, faz-se necessário dividir o arquivo em pequenas partes que possam ser carregadas, mas ao invés de gerar diversos arquivos diferentes, o mesmo só será lido até um certo item. Dessa forma, o conjunto de itens que foi lido será chamado de página.

Assim, com as diferentes páginas e pequenas leituras ao longo de todo arquivo, o mesmo não necessitará de ser completamente carregado para a procura de um item que esteja nas primeiras linhas. Ademais, também terá a vantagem de que a pesquisa do item pode ser melhorada a partir do uso de diferentes tipos de pesquisa em arquivos que usam a paginação.

4 - Em relação ao custo de memória, a árvore B^* é mais interessante, pois os seus itens serão armazenados somente nos nós folhas, dessa forma, não será necessário utilizar apontadores para os registros dentro de cada página, diminuindo a memória gasta por cada uma. Todavia, como a memória também será menor, é possível que cada página possua uma maior quantidade de itens.