

RELATÓRIO DO PROJETO FINAL

No projeto foram usadas basicamente duas estruturas de dados: uma Lista simplesmente encadeada e uma Árvore Binária de Busca, para que a busca da informação sobre as vendas sejam mais eficiente do que se usar somente uma lista simplesmente encadeada, por exemplo.

As operações básicas em uma árvore de pesquisa binária levam tempo proporcional à altura da árvore. Para uma árvore binária completa com n nós, essas operações são executadas em $\Theta(\log n)$ no pior caso. Se a árvore for uma cadeia linear de n nós, contudo, as mesmas operações levam $\Theta(n)$ no pior caso. A altura esperada de uma árvore de pesquisa binária construída aleatoriamente é $O(\log n)$, de modo que as operações básicas sobre um conjunto dinâmico dessa árvore levam $\Theta(\log n)$ tempo na média.

Para responder a primeira pergunta do projeto, foi utilizada uma Árvore Binária de Busca para armazenar os códigos das filiais sem repetir nenhum. Cada nó da árvore aponta a uma lista de objetos de tipo venda cuja filial coincide com a chave do nó, como mostra o exemplo da figura 2, o qual está baseado nos dados de entrada da figura 1.

```
16, 2017_06, 1, 5.00
8, 2017_12, 2, 100.00
24, 2017_01, 3, 25.00
4, 2016_06, 4, 20.00
12, 2016_09, 5, 30.00
20, 2017_05, 6, 10.00
28, 2015_03, 7, 10.00
2, 2015_10, 8, 45.00
6, 2017_07, 9, 15.00
10, 1999_05, 10, 10.00
14, 2015_06, 11, 5.00
18, 2017_02, 12, 10.00
22, 2017_01, 13, 5.00
26, 2017_10, 14, 100.00
30, 2012_12, 15, 50.00
```

Figura 1. Dados de vendas (exemplo) no arquivo *vendas.txt*

Para responder a terceira pergunta, foi usada uma estrutura similar à anterior, mas agora a Árvore Binária de Busca vai armazenar as datas (no formato resultante do cálculo: $100 * \text{ano} + \text{mes}$) das vendas, também sem repetir nenhuma chave na árvore.

Para responder a segunda pergunta do projeto, foi utilizada uma Árvore Binária de Busca para armazenar os códigos das filiais sem repetir nenhum. Cada nó desta árvore aponta a uma outra árvore binária de busca, a qual armazena as datas das vendas efetuadas na filial correspondente à chave do nó da árvore primária. Finalmente, cada nó na segunda árvore contém um apontador a uma lista de objetos de tipo venda, cujas filiais coincidem com a chave do nó correspondente na árvore primária e cujas datas coincidem com a chave do nó correspondente na árvore secundária, como mostra o exemplo da figura 3.

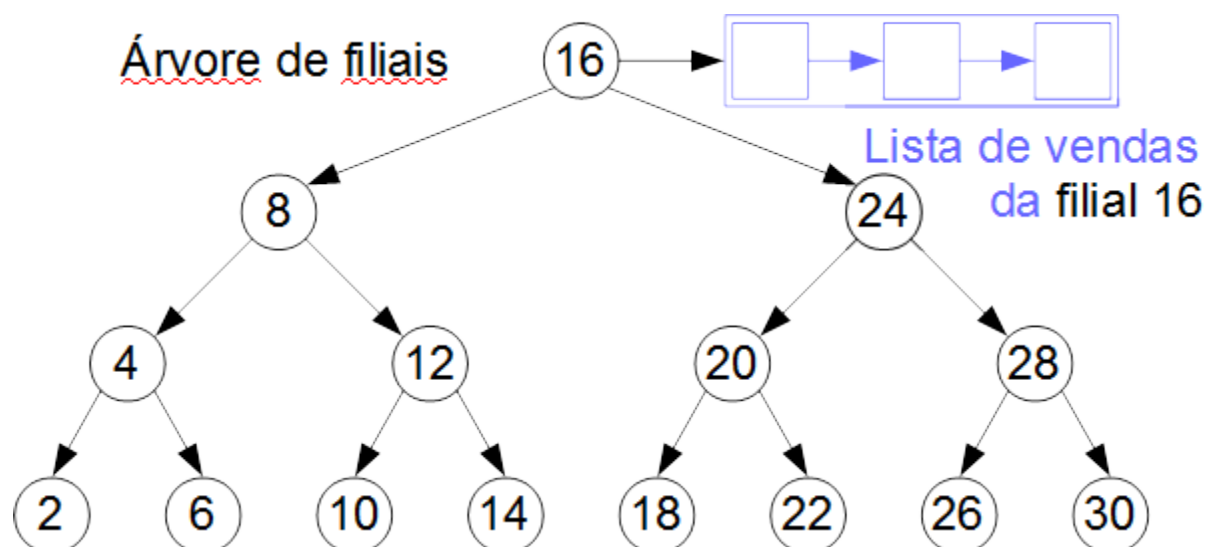


Figura 2. Exemplo da estrutura de dados usada para reponder a primeira pergunta.

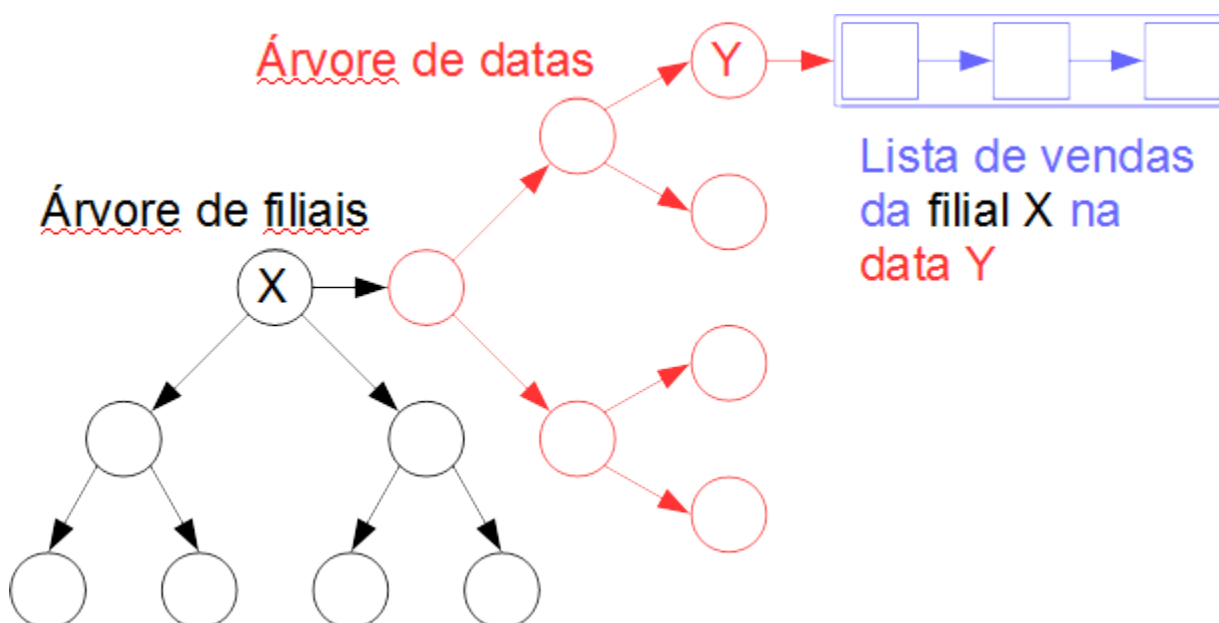


Figura 3. Exemplo da estrutura de dados usada para reponder a segunda pergunta.

Bibliografia

Cormen, T.H., et. al.: Introduction to Algorithms. Third Edition. The MIT Press, Cambridge (2009)