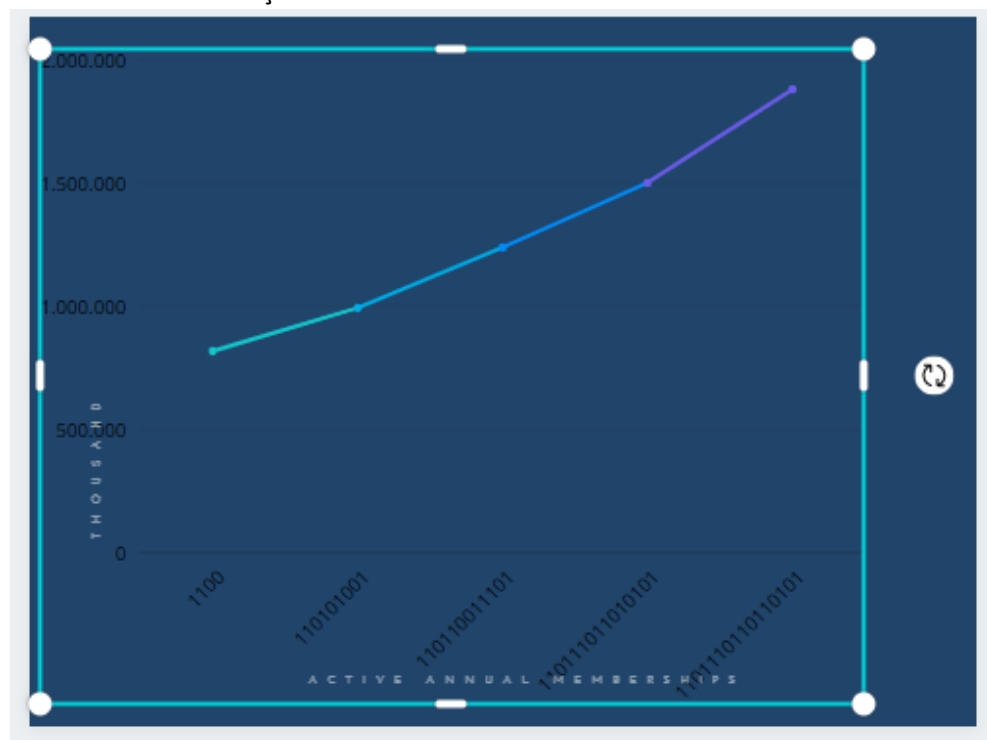


Alunos: Igor Dourado, Pedro Castro, Mathias Chavez e Joana Figueredo

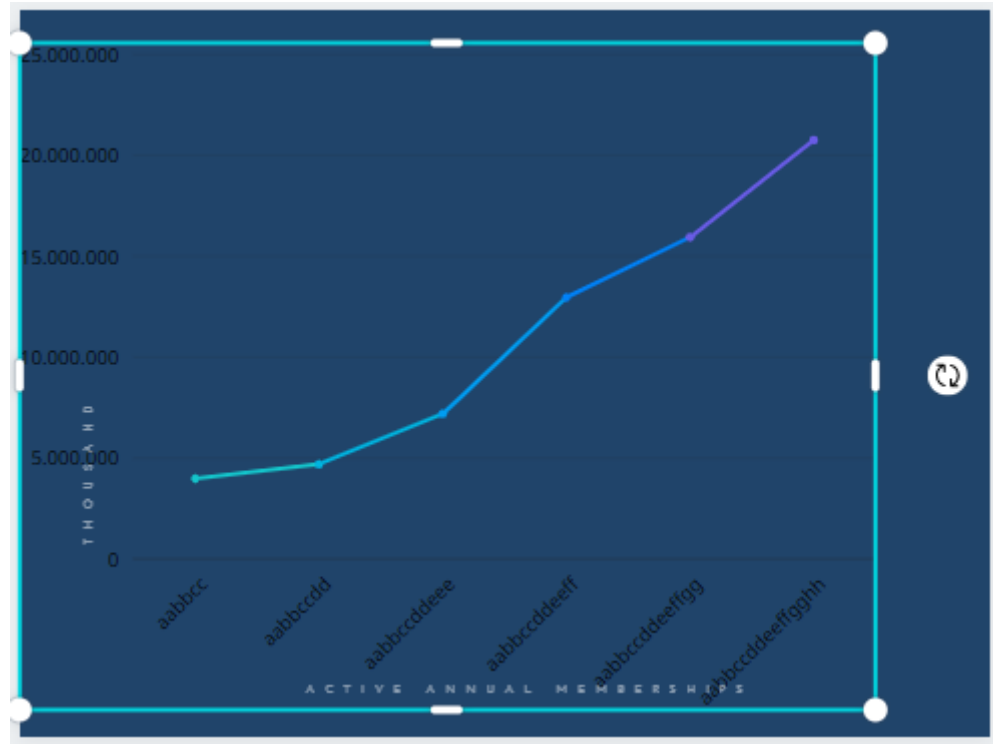
- 1- Não, pois as informações apresentadas no artigo são muito rasas e supérfluas, visto que não busca explicar o motivo de o algoritmo ter aquela complexidade, apenas expõe a informação e fala que basta apenas o leitor ver o pseudocódigo para constatar isso, ou seja, não levando em conta os leitores leigos, e também não explicita se ele está usando big-O, big- Θ ou big- Ω em sua análise, também mostrando informações incorretas durante sua análise, visto que na verdade esse algoritmo tem uma complexidade de **tempo $O(n \log n)$** em seu total, e a depender da construção da fila de prioridade (se ela for implementada com um vetor ordenado), podemos ter também complexidade $O(n^2)$
- 2-

Gráfico da decodificação



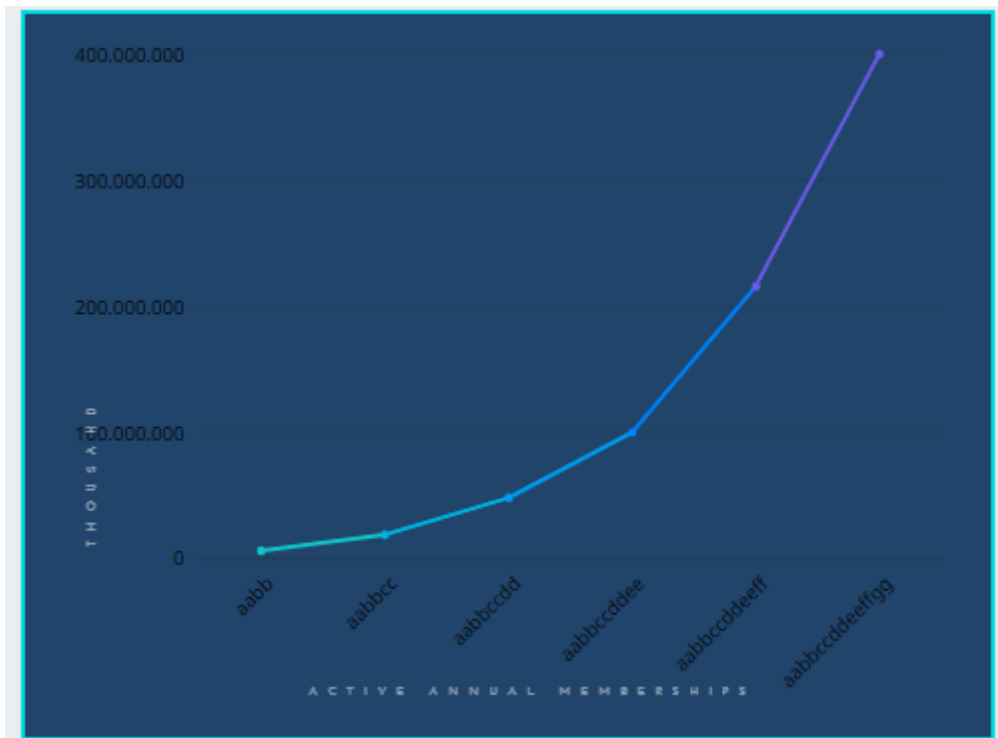
Notamos uma tendência de aumento linear, ou seja, em tempo linear, quando a questão é a decodificação, ou seja, o tempo para a execução dessa parte do código vai aumentar de acordo com o tamanho do código a ser decodificado, isso chega a ser citado no artigo, então confere com o que foi dito no artigo, é de tempo linear a execução desse método.

Gráfico de codificação



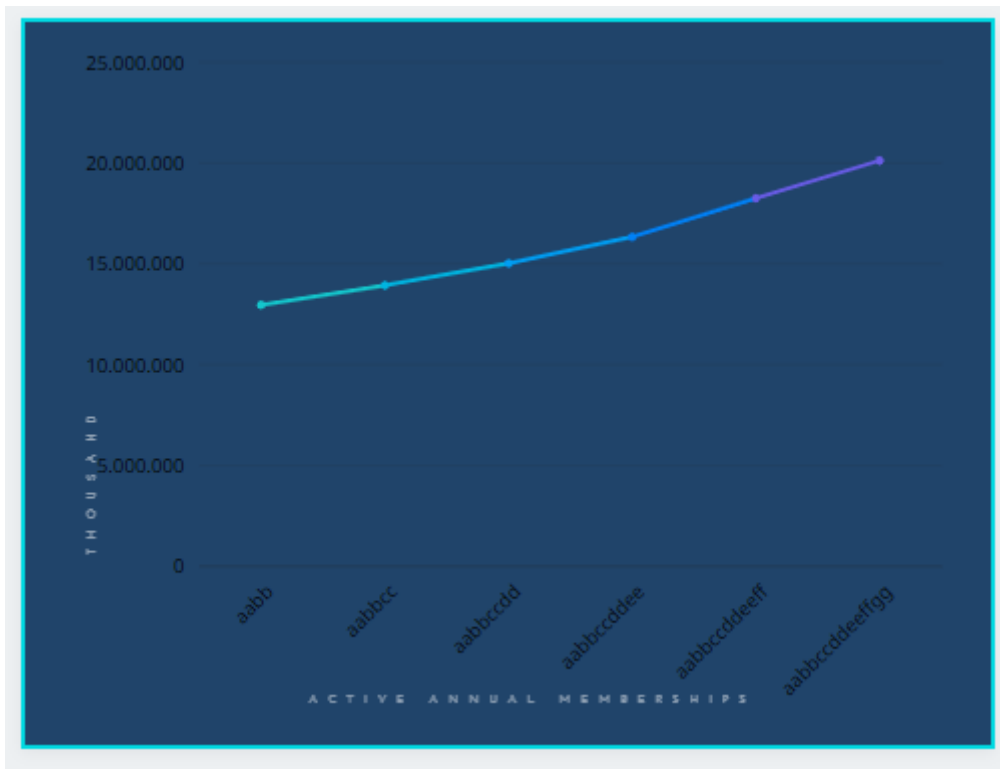
Aqui nós aumentamos de 2 em 2 letras diferentes por interação para manter uma lógica no gráfico, e assim como era de se esperar, como é o caminho inverso, notamos que seguiu a mesma lógica do gráfico de codificação, ou seja, uma linearidade, quanto mais aumentamos, ele proporcionalmente leva mais tempo a processar, ou seja, aqui também encontramos uma complexidade de tempo linear.

Gráfico de cálculo de frequência



Para o cálculo da frequência notamos que ele segue a lógica do código, em que utilizamos 3 for não encadeados, ou seja, ele tem uma tendência ao gráfico de complexidade cúbica, como podemos ver, em que ele vai aumentando exponencialmente conforme adicionamos 2 letras por entradas, ou seja, segue a ideia teórica, porém no artigo não há citações sobre a complexidade dessa parte do código, então não podemos dizer se ela está de acordo ao artigo, mas está, sim, de acordo com a teoria, com 3 for no código.

Gráfico da montagem da árvore de Codificação de Huffman



Notamos que ele leva um tempo relativamente considerável para poder fazer a montagem da árvore, porém, conforme aumentamos a entrada, ele não muda muito esse tempo, há só uma pequena mudança, que podemos considerar linear ou até logarítmica, porém ainda assim com mais característica da curva linear, então podemos considerar a sua complexidade linear, e novamente essa parte não tem no artigo, então não encontramos correspondência.