

UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE - UFRN  
BRENO KLINSMANN  
JOHNNYLEE BRYAN MARQUES DA ROCHA

## Análise Empírica

Natal-RN, 17 de Março de 2016

## 1 - Introdução

Vinhemos a partir deste relatório, explicitar os resultados e estratégias utilizados para a análise empírica de um algoritmo de teste passado na disciplina de Estrutura de Dados Básicas I. O mesmo, analisa e imprime os dados resultados de tempo de execução de 7 funções, sendo elas: Busca binária recursiva e iterativa, Busca sequencial recursiva e iterativa, Busca Ternária recursiva e iterativa e própria busca binária padrão (biblioteca "cstdlib"). Cada função deverá ser executada em 3 casos, estes: Pior caso (quando o elemento buscado não se encontra no vetor),  $\frac{3}{4}$  do vetor (quando o elemento está na quarta parte do vector) e o caso padrão (quando o elemento é encontrado em alguma posição do vetor).

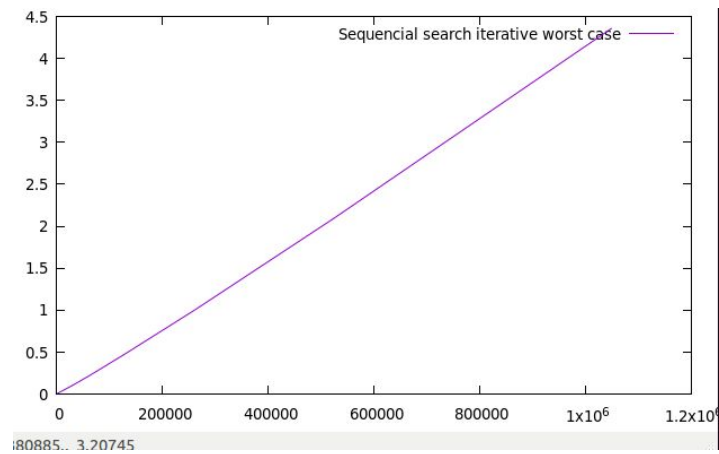
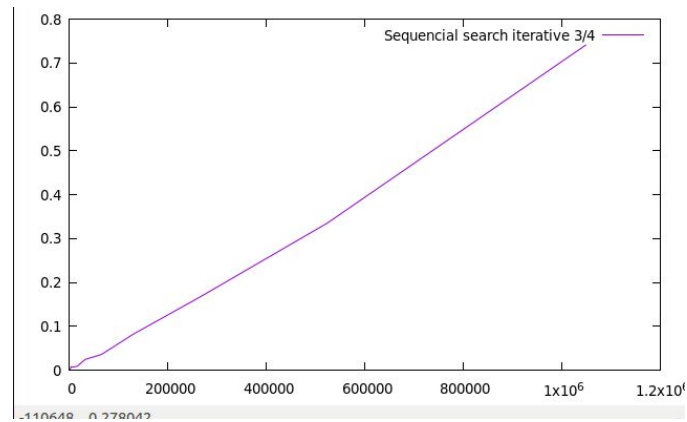
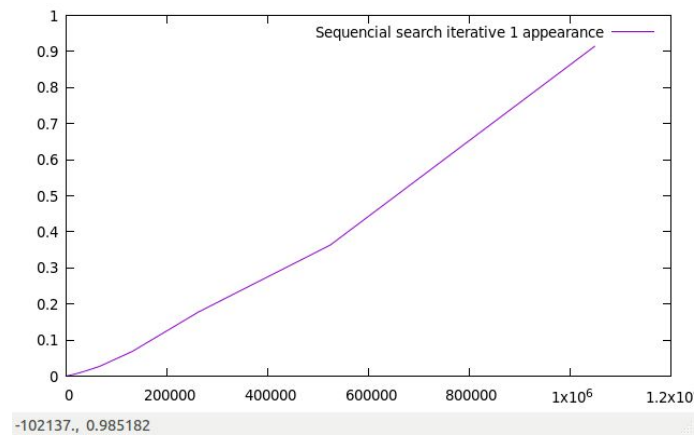
## **2 - Recursos e Estratégias**

Cada algoritmo foi executado 100 vezes para cada caso e o tempo foi calculado para cada, e ao terminar os 100 processos, foi tirada a média dos tempos e retornado para um arquivo “.dat”. E para cada 100 execuções de 3 casos, o tem o seu tamanho aumentado de forma exponencial, sempre começando de  $2^5$ .

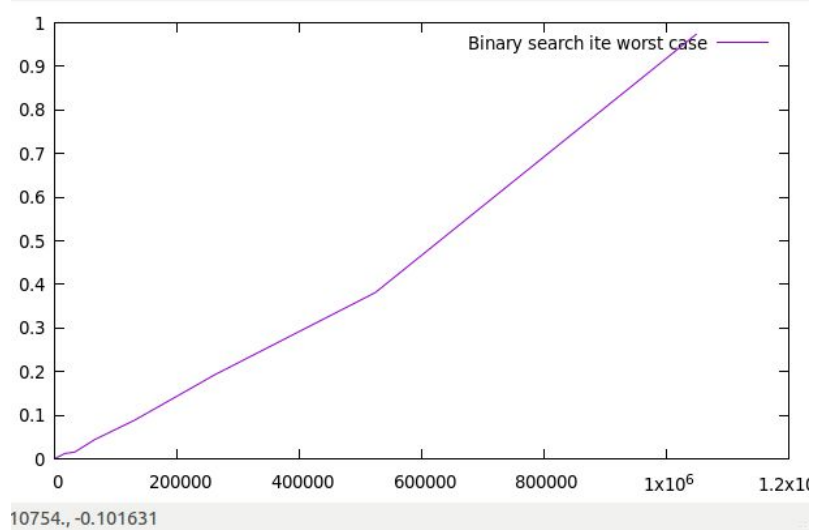
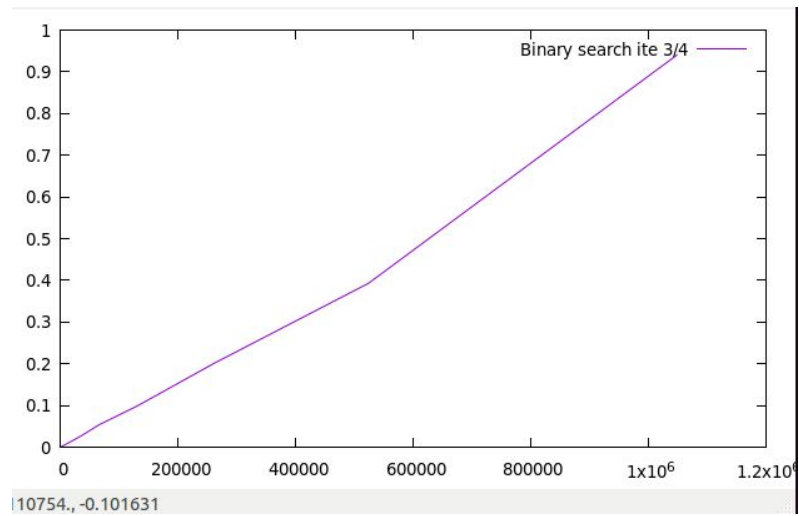
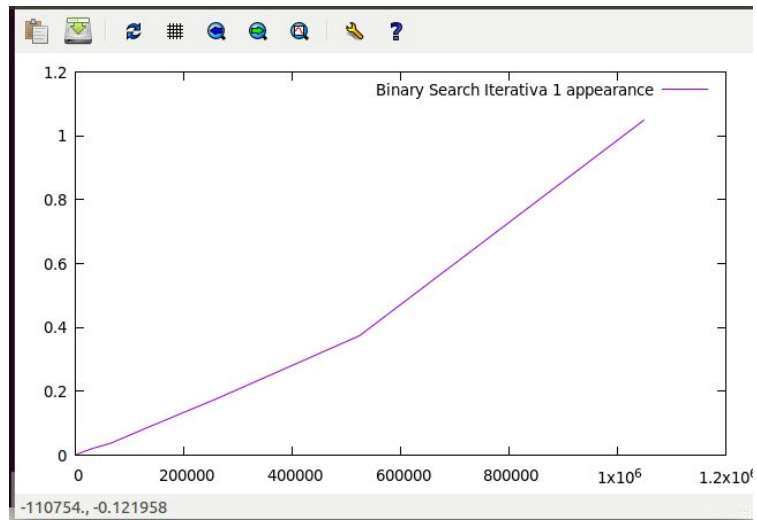
A máquina utilizada para executar o algoritmo, foi desktop com as seguintes configurações: Intel i5 3470 3.2GHz, 8gb ram DDR3 1600mhz, com OS Ubuntu 15.10.

### 3 - Resultados obtidos

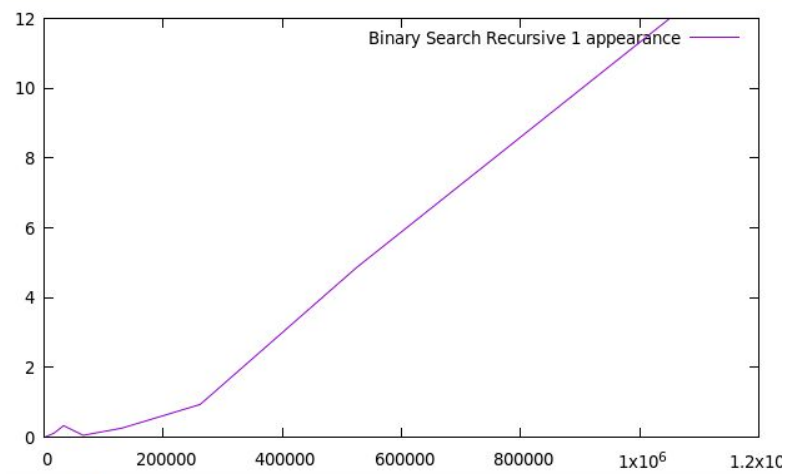
- Busca sequencial Iterativa



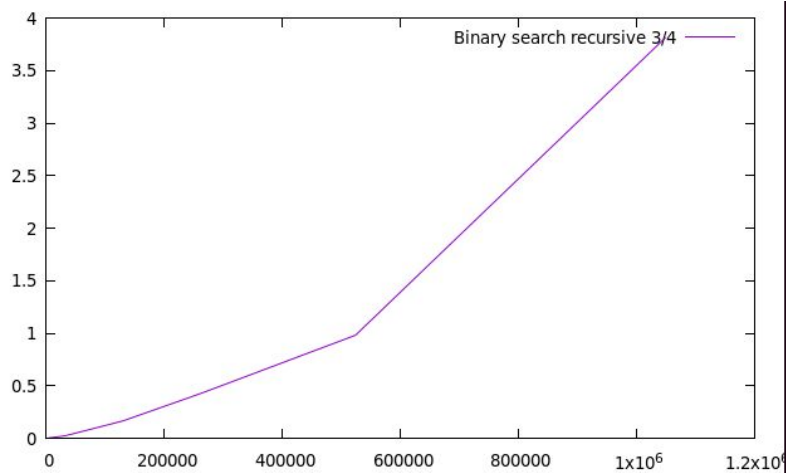
- Busca Binária Iterativa



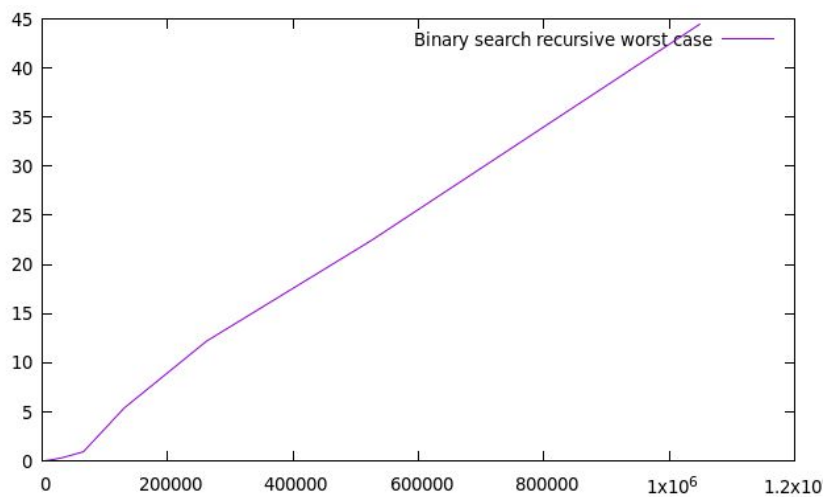
- Busca Binária Recursiva



23787., 12.5766

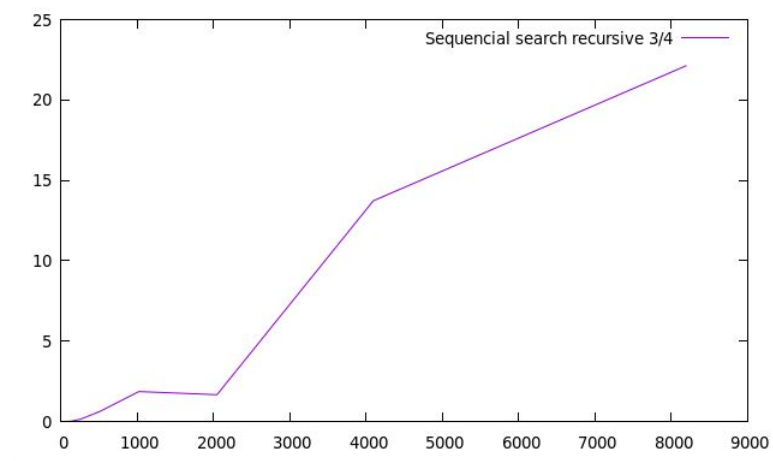
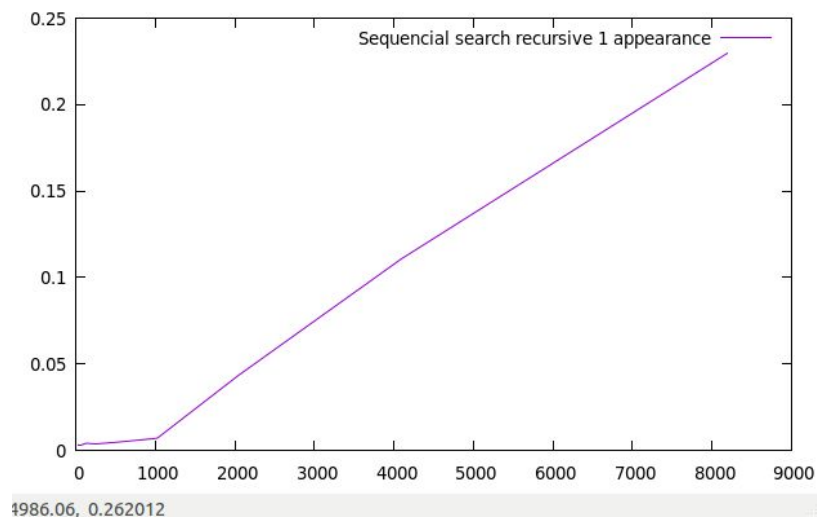
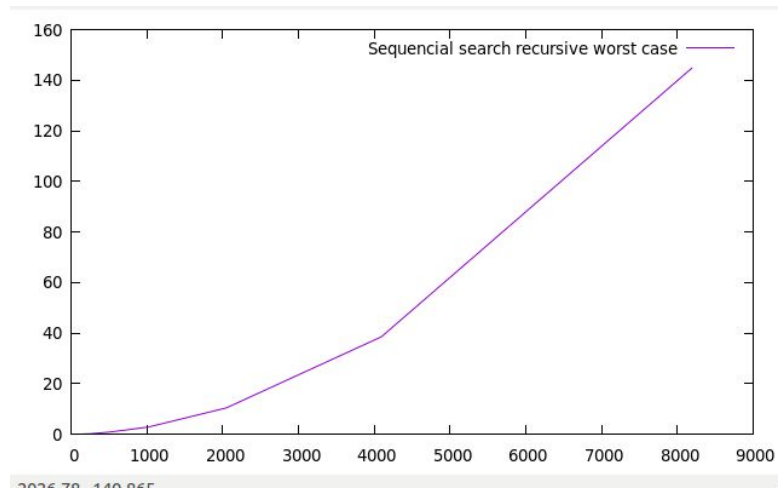


32532., 3.3532

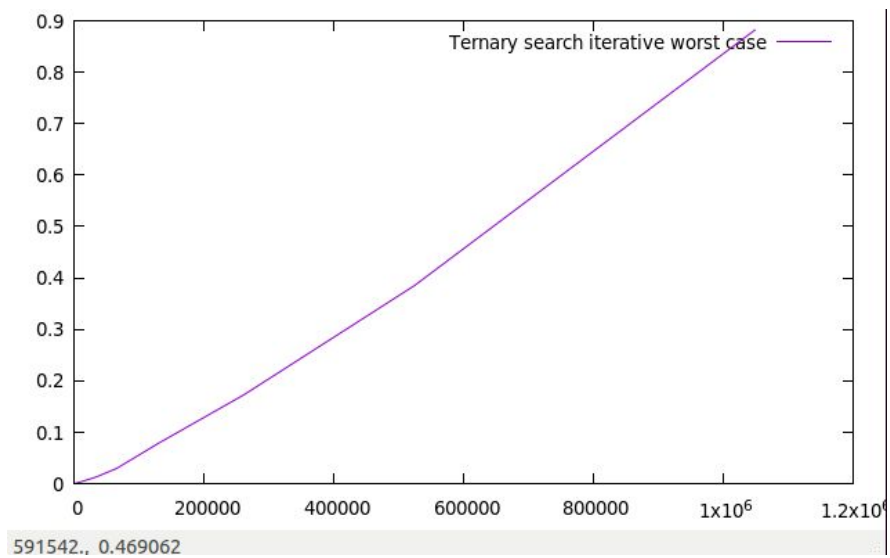
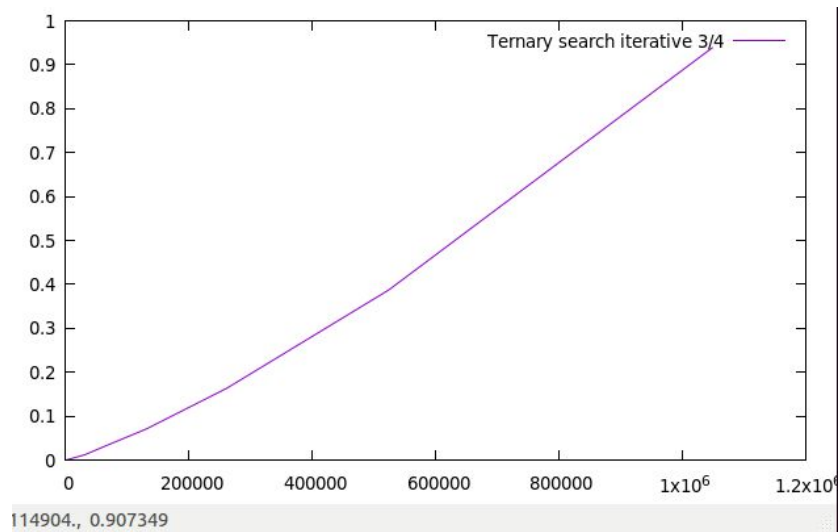
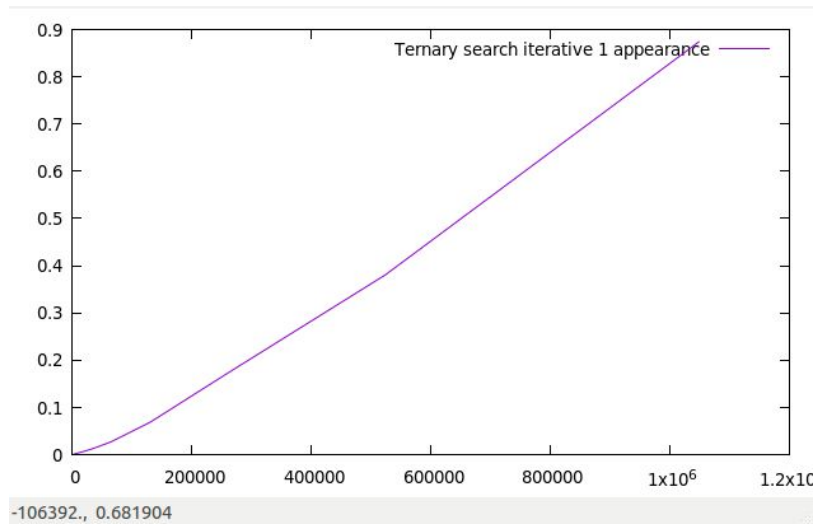


656701., 47.8357

- Busca Sequencial Recursiva

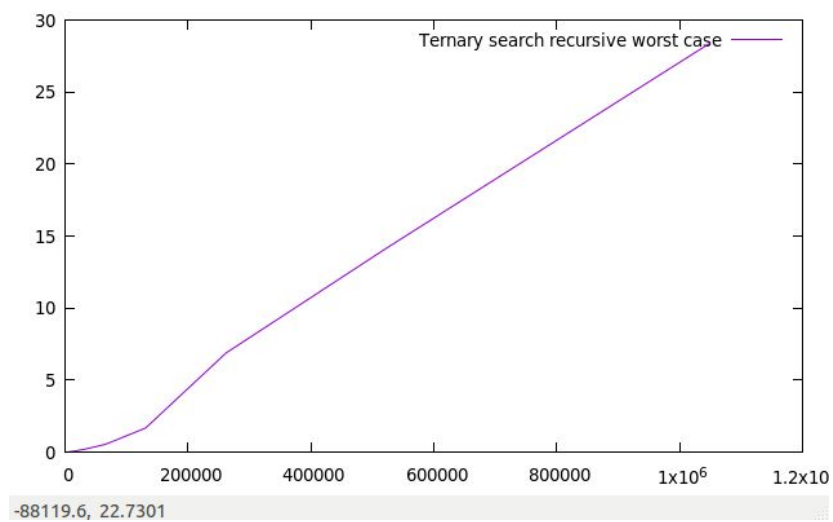
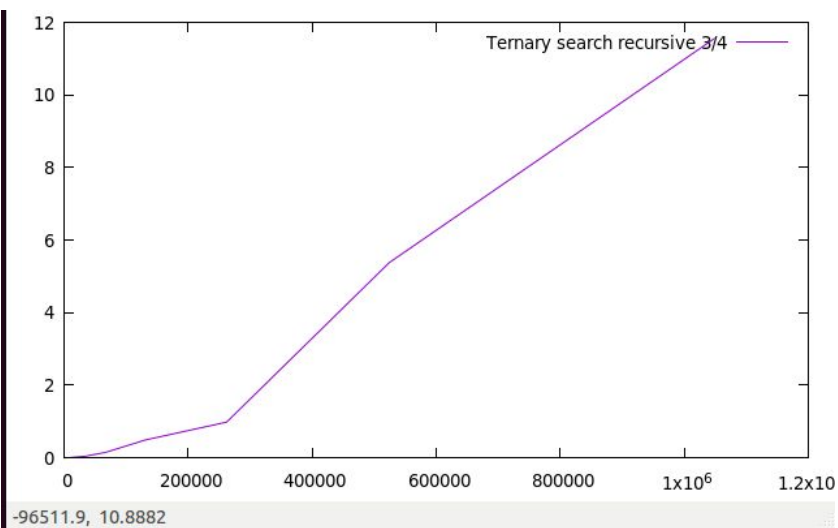
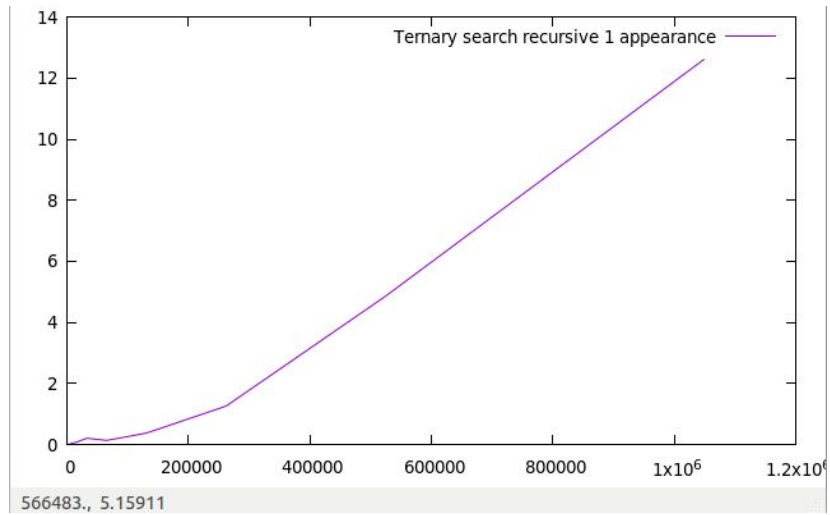


- Busca Ternária Iterativa

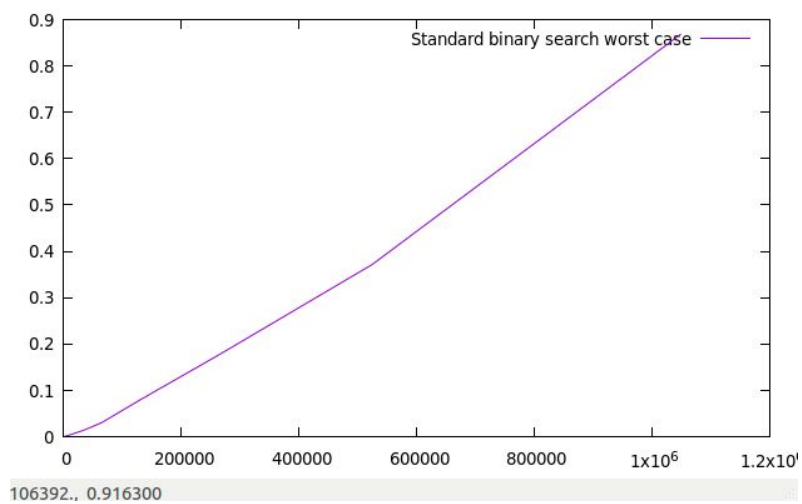
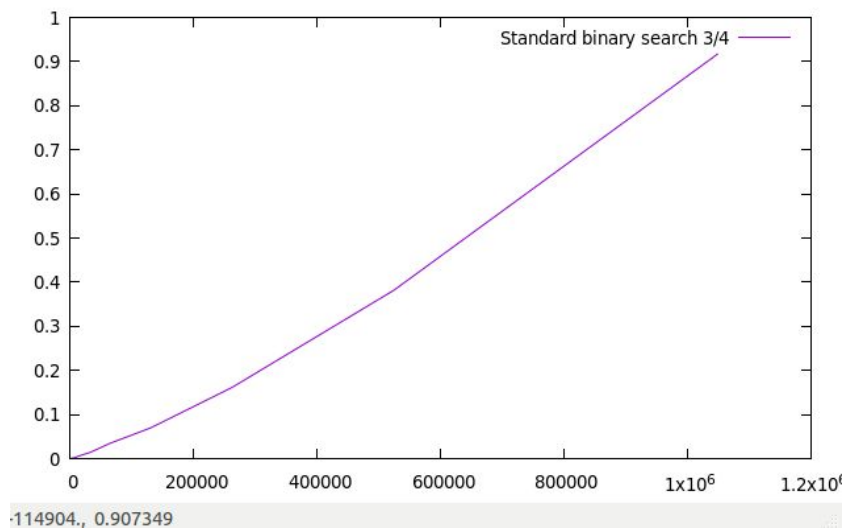
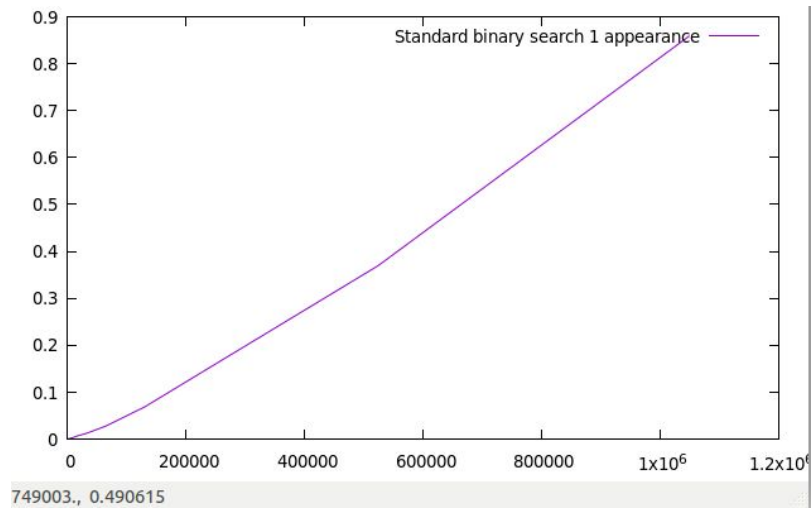




- Busca Ternária Recursiva



- **Standard Binary Search**



#### **4 - Conclusão**

A partir dos gráficos, pode-se concluir que os algoritmos se mostraram estáveis, na maioria dos casos. Porém, também pode-se observar que as buscas binárias se mostraram mais eficazes em relação as outras apresentadas.

Durante as operações, foram obtidos erros de alocação no momento de executar a função Busca sequencial recursiva, pois, está apresenta um limite de empilhamento de funções, equivalente a  $2^{13}$ . Ao ser observado isso, tivemos de limitar as execuções, em função desta.

Ao executar o método de cálculo de tempo, um subvetor é criado a partir do maior, com o tamanho necessário para executar aquele determinado tamanho passado no laço. Função utilizada para isso, utiliza o método "begin()" do vector para pegar o valor inicial e o soma com "n", que neste caso é o tamanho do vector.

Como podemos observar, as funções de busca binária são quase que sempre, as melhores opções quando se trata de buscas em geral.