UNIVERSIDADE FEDERAL RIO GRANDE DO NORTE

INSTITUTO METRÓPOLE DIGITAL

Disciplina: Estrutura de dados Básicas I

Relatório de Análise Empírica de Algorítmos de Busca

Prof.: Jorge Estefano Santana De Souza

Alunos: Kelvin Coelho Loiola

Sara Paloma de Souza

NATAL - Agosto de 2016

1 INTRODUÇÃO

Este relatório tem como objetivo apresentar os resultados da análise empírica realizada com a utilização de quatro algoritmos de busca distintos, utilizando os critérios exigidos para realização dos experimentos.

O desenvolvimento deste relatório teve como base as instruções passadas em sala de aula, juntamente com o material apresentado e disponibilizado pelo professor. Por se trata de uma análise empírica de algoritmos computacionais, tornou-se necessário o desenvolvimento de artifícios para agilizar a coleta destes dados, como registro do tempo computacional e contagem de passos de cada programa executado.

Os algoritmos utilizados foram: busca sequencial iterativa, busca sequencial recursiva, busca binária iterativa e busca binária recursiva.

2 OBJETIVO DA ANÁLISE

O objetivo geral desta análise é realizar uma comparação entre os algoritmos de busca sequencial e binária nas versões iterativas e recursivas.

3 HIPÓTESE

Os algoritmos de busca binária têm um desempenho, na maioria dos casos, superior aos de busca sequencial.

4 MATERIAIS E MÉTODOS

4.1. Informações do Computador Utilizado Relevantes ao Experimento

Processador: Core i7 2630QM	
Número de núcleos	4
Nº de threads	8
Frequência baseada em processador	2 GHz
Frequência turbo max	2.9 GHz
Cache	6 MB L3
Velocidade do barramento	5 GT/s DMI
Memória RAM e ROM	
Memória RAM Total	16 GB
Tipos de memória	DDR3 1333
Canais	Dual channel
Largura de banda máxima da memória	21,3 GB/s

Memória ROM	HD Sata Seagate 500 GB - 7200rpm
Sistema Operacional: Ubuntu 16.04 - 64 bits	

4.2. Algoritmos Utilizados

Utilizamos quatro algoritmos principais como base do experimento, foram estes*:

I. Algoritmo de busca sequencial iterativa;

II. Algoritmo de busca sequencial recursiva;

III. Algoritmo de busca binária iterativa;

IV. Algoritmo de busca binária recursiva;

V. Biblioteca desenvolvida com o nome "complementos.h", onde foram incluídas as demais bibliotecas utilizadas no projeto;

*As funções de cada um dos algoritmos foram transcritas do material disponibilizado pelo professor e posteriormente adaptados para manter um padrão de organização em todos os algoritmos, possibilitando uma apresentação mais adequada do relatório.

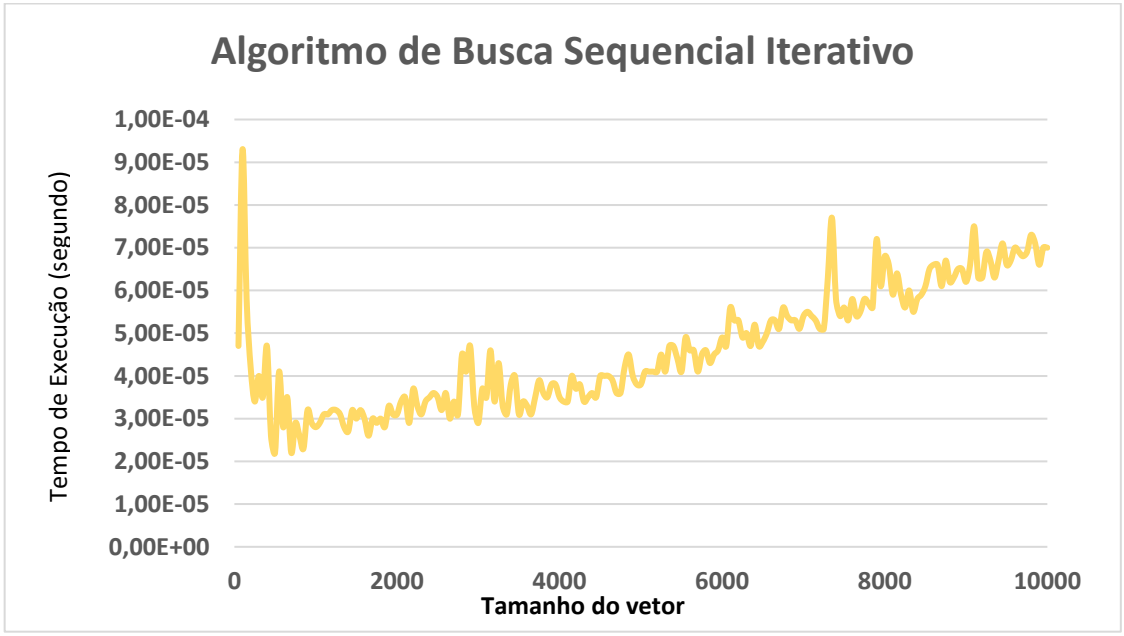
4.3. Métodos e Estratégias de Desenvolvimento

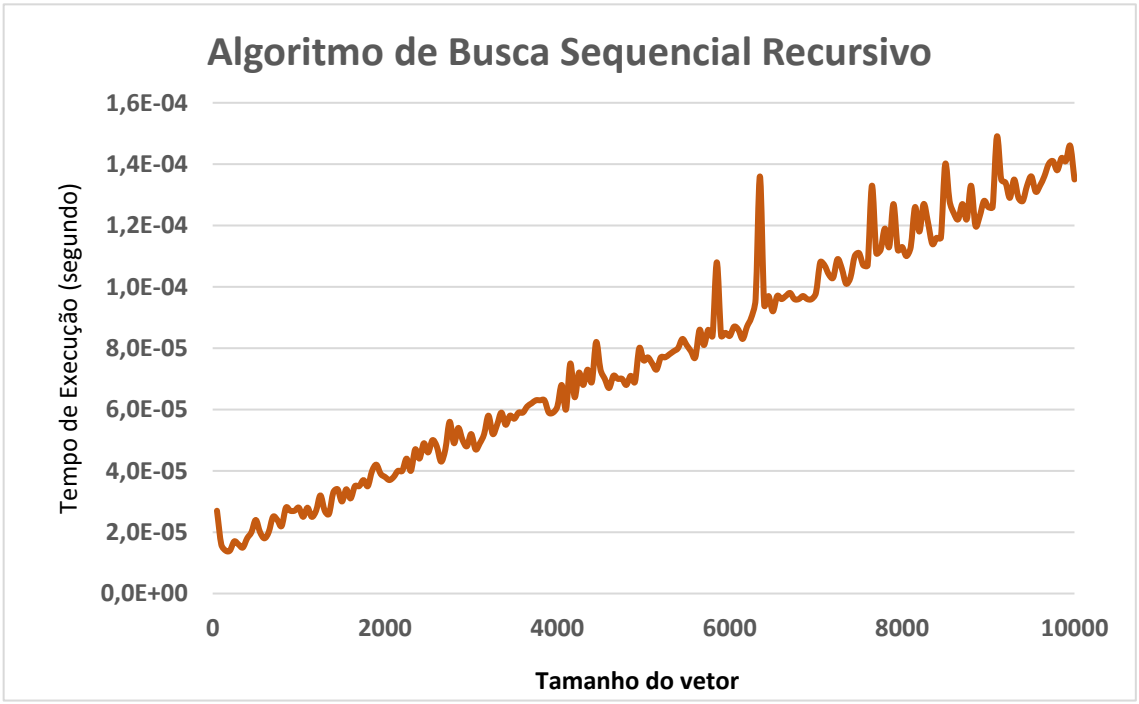
- Logo no início do desenvolvimento, percebemos que o principal problema para realização dos experimentos estava relacionado com a necessidade de executar os programas diversas vezes. Mais precisamente 800 vezes no total. Pois para a análise dos dados, deveríamos utilizar um vetor de inteiros recebendo do usuário o tamanho utilizado para este vetor. Tamanho este iniciando em 50 e acrescentando mais 50 casas a cada execução, até o limite de 10.000 no tamanho deste vetor.
  Para solucionar esse problema, desenvolvemos um script para o terminal do Ubuntu, onde as execuções com suas respetivas entradas foram configuradas.
- O critério de busca utilizado foi o do caso médio. Onde o número buscado estaria localizado na metade do vetor. Para isso, o alvo da busca era sempre obtido pelo $\frac{[tamanho\ do\ vetor]}{2}$;
- Para calcular o tempo, utilizamos a biblioteca <sys/time.h>. E para gerar os arquivos com os dados do tempo de execução e a contagem dos loops, utilizamos a biblioteca <fstream>;
- Para a compilação dos arquivos de modo otimizado, utilizamos um makefile.

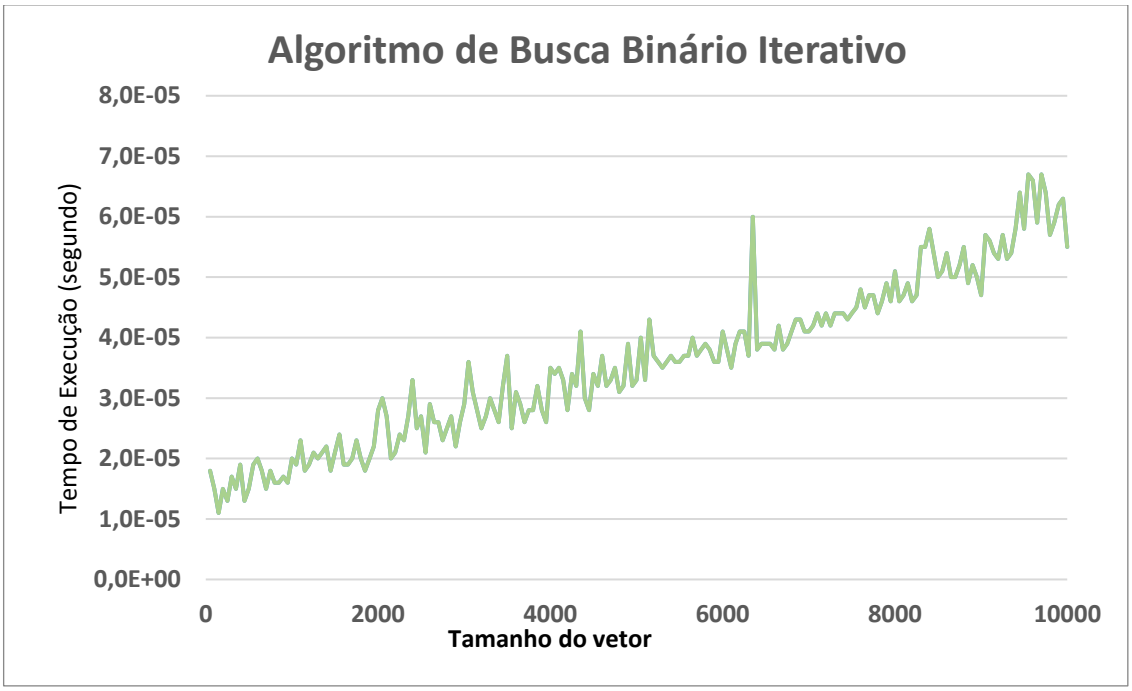
5 DADOS COLETADOS

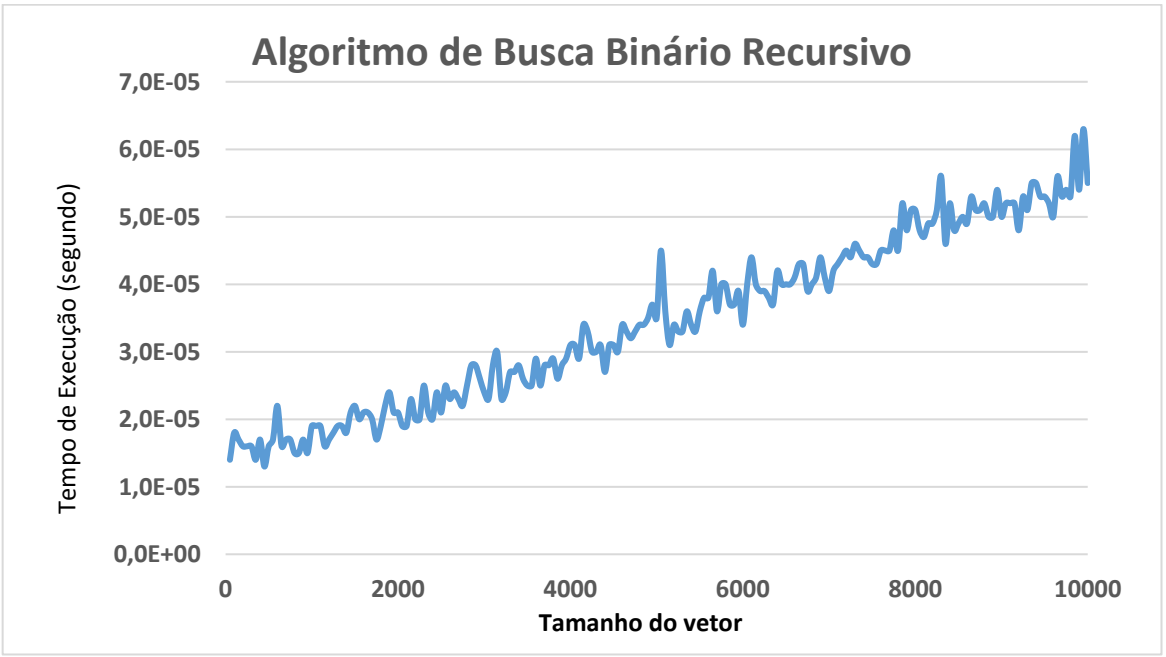
5.1.Tempo de Execução

A partir dos dados coletados, podemos observar que, no geral, o tempo de execução de todos algoritmos de busca tenderam a diminuir ao longo da execução, apesar de apresentar frequente oscilação em todo o processo. O que acreditamos ter ocorrido devido às mudanças no padrão de processamento da máquina, como pode ser visto nos gráficos abaixo:

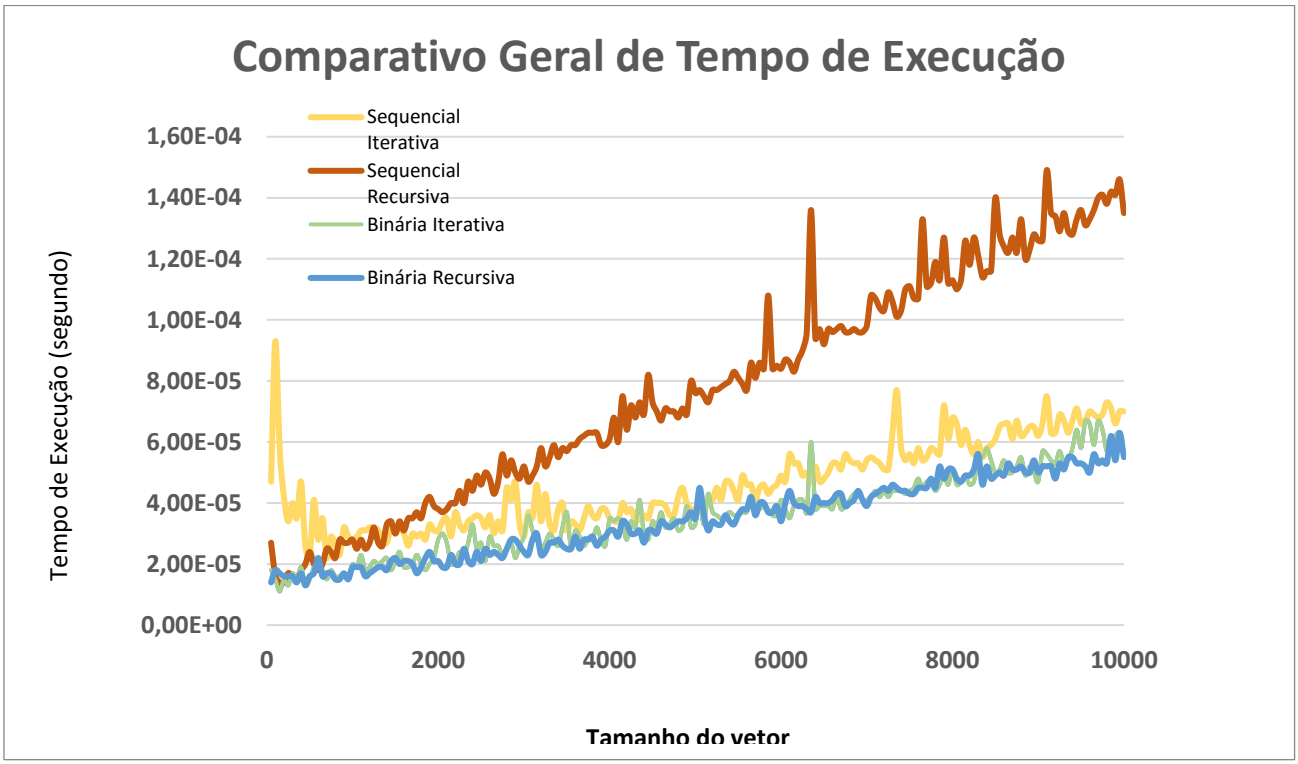





Algoritmo de Busca Binário Iterativo
8,0E-05
7,0E-05
6,0E-05
5,0E-05
4,0E-05
3,0E-05
2,0E-05
1,0E-05
0,0E+00
Tempo de Execução (segundo)
0
2000
4000
6000
8000
10000
Tamanho do vetor

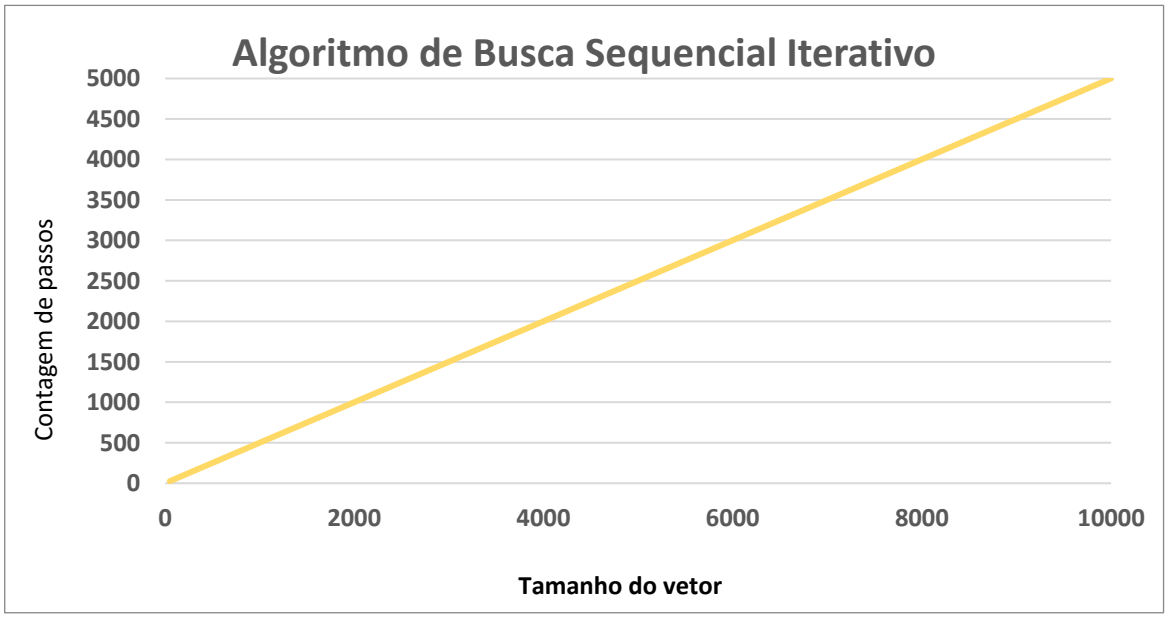

Algoritmo de Busca Binário Recursivo
7,0E-05
6,0E-05
5,0E-05
4,0E-05
3,0E-05
2,0E-05
1,0E-05
0,0E+00
Tempo de Execução (segundo)
0
2000
4000
6000
8000
10000
Tamanho do vetor

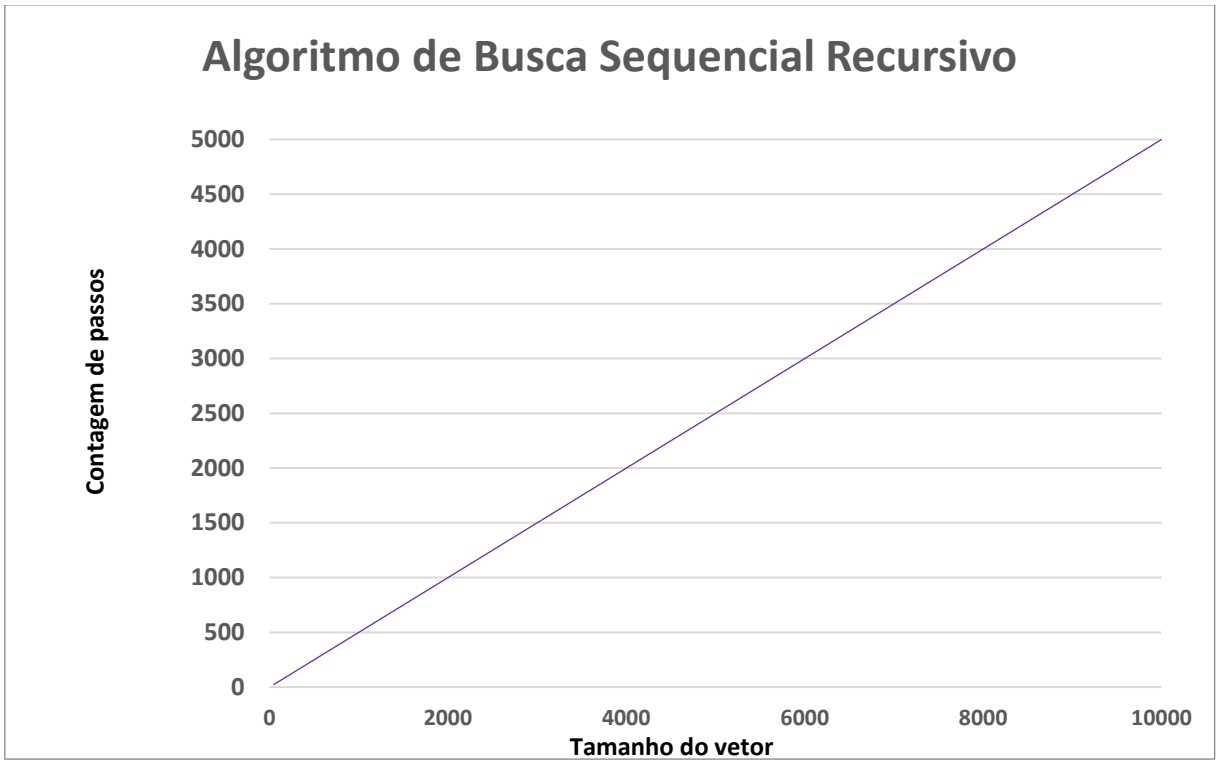
Para melhor visualização, podemos ver no gráfico comparativo geral como o tempo de execução geral ficou equilibrado ao longo do tempo com excessão do algoritmo de busca sequencial iterativo, que teve uma redução no início e depois manteve o equilibrio de crescimento.



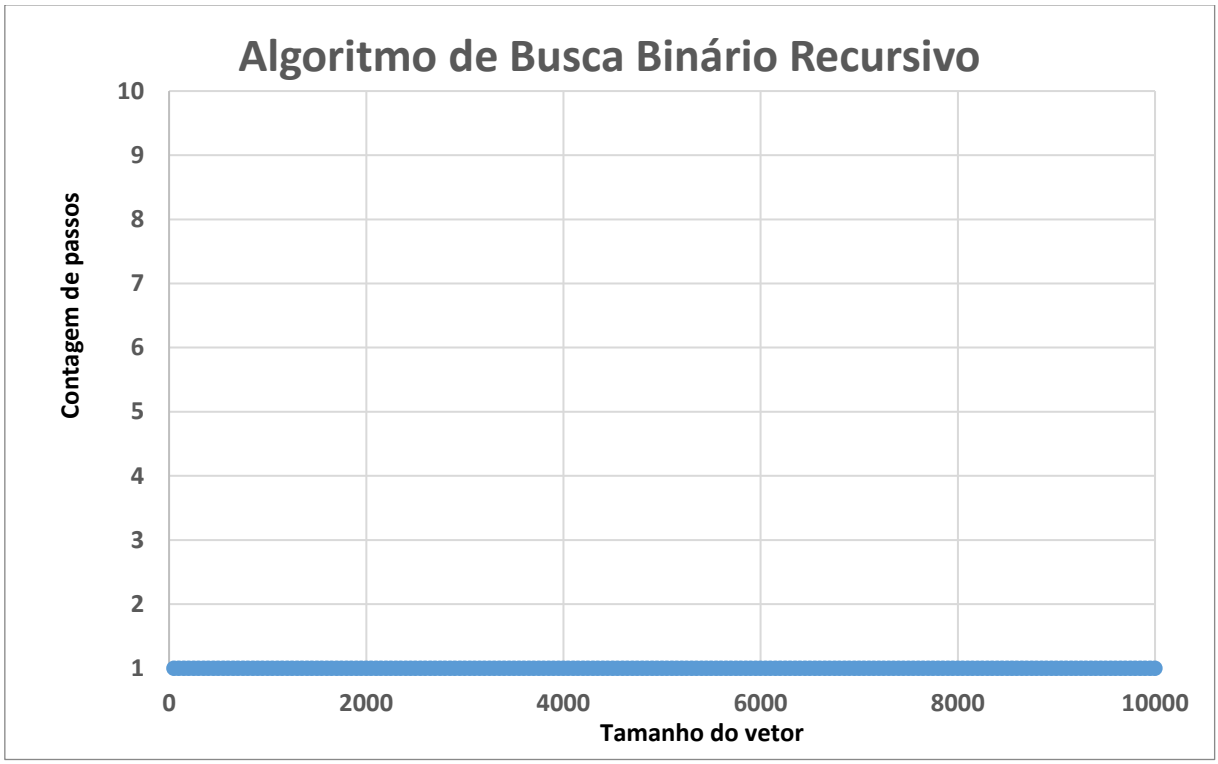
5.2. Contagem de Loops

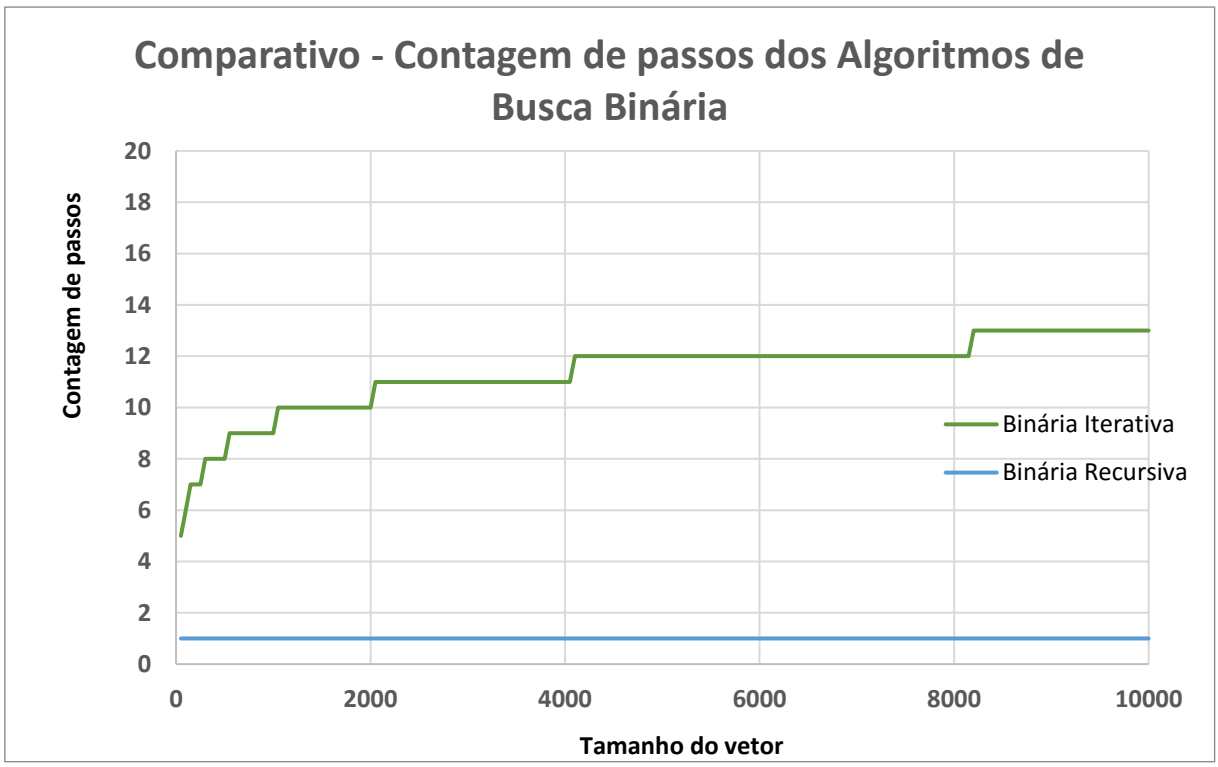
No nosso experimento, os dados de contagem podem ser vistos com maior clareza de diferença entre os diferentes algorítmos. Exceto com relação aos algorítmos de busca sequencial iterativo e recursivo. Que só podemos observar com clareza a diferença quando recorremos à tabela dos dados.




Algoritmo de Busca Sequencial Recursivo
Contagem de passos
5000
4500
4000
3500
3000
2500
2000
1500
1000
500
0
0
2000
4000
6000
8000
10000
Tamanho do vetor


Algoritmo de Busca Binário Iterativo
Contagem de passos
14
12
10
8
6
4
2
0
0
2000
4000
6000
8000
10000
Tamanho do vetor


Algoritmo de Busca Binário Recursivo
Contagem de passos
10
9
8
7
6
5
4
3
2
1
0
2000
4000
6000
8000
10000
Tamanho do vetor


Comparativo - Contagem de passos dos Algoritmos de Busca Binária
Contagem de passos
20
18
16
14
12
10
8
6
4
2
0
0
2000
4000
6000
8000
10000
Tamanho do vetor
Binária Iterativa
Binária Recursiva

Quadro comparativo – Contador dos Algorítmos Sequenciais

Tamanho do Vetor	Iterativo	Recursivo
50	26	24
100	51	49
150	76	74
200	101	99
...	...	...
1350	676	674
1400	701	699
1450	726	724
1500	751	749
1550	776	774
...	...	...
3700	1851	1849
3750	1876	1874
3800	1901	1899
...	...	...
5000	2501	2499
5050	2526	2524
5100	2551	2549
...	...	...
9900	4951	4949
9950	4976	4974
10000	5001	4999

6. Resultados e discussões

No decorrer da análise foi possível verificar que, quando levada em consideração a maior amostra, a busca binária recursiva foi a que teve o melhor desempenho em relação a todos os outros algoritmos, perdendo somente para a busca sequencial iterativa nas primeiras amostras, até a amostra 122. Com isso, também foi permitido observar que a busca binária recursiva foi a que teve o pior desempenho em relação a menor amostra.

Ao comparar a busca sequencial recursiva com a busca binária recursiva, verifica-se que a busca binária é bem superior quando comparada na maior amostra. O mesmo resultado pode ser observado para os algoritmos de busca iterativa, aonde a busca binária é superior à busca sequencial para as maiores amostras. Os dados coletados confirmam a intuição de que os algoritmos de busca binária são mais rápidos e eficientes que os de busca sequencial.