
BOYER MOORE AGAINST RABIN KARP
Jack Sime

The Problem.

- String Searching is available in most of the devices and applications we use today.
- Providing the best solution to searching in strings allows people to spend less time waiting and more time searching.
- The ability to search large sections of text for specific patterns allows user to find exactly what they need in a quick time.
- I chose to implement the Boyer Moore Horspool Algorithm and the Rabin-Karp Algorithm which both solve this problem but use different methods.

Parameters of testing

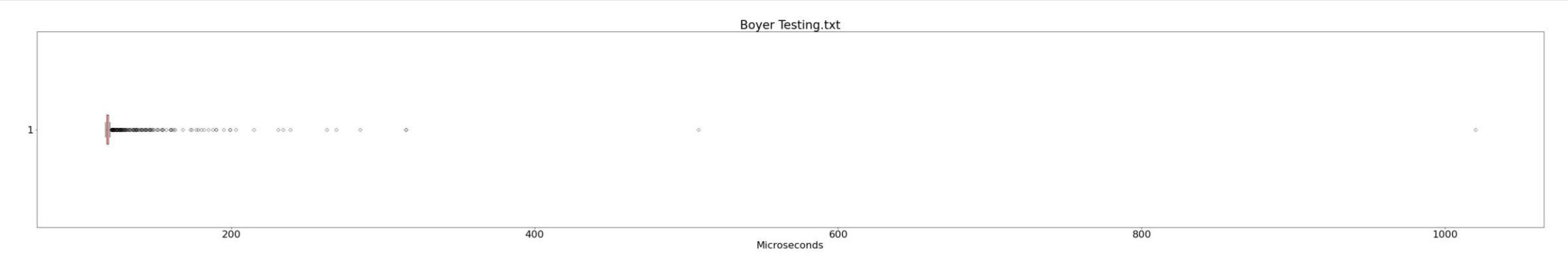
- Both Algorithms were tested on the same system with a 12Mb Cache.
- File jute_book was used for inside the cache testing (174KB)
- File Testing was used for outside cache performance testing. (18.5MB)
  - Testing.txt was compiled from several books:
    The Historians' History of the World in Twenty-Five Volumes, Volume 08 by Williams
    The King James Version of the Bible
    Walden, and On The Duty Of Civil Disobedience by Henry David Thoreau
    A Tale of Two Cities by Charles Dickens
    The Decameron of Giovanni Boccaccio by Giovanni Boccaccio
    The History of the Peloponnesian War by Thucydides
    English Literature by William J. Long
    Villa Eden: The Country-House on the Rhine by Berthold Auerbach
    Campfire and Battlefield by Selden Connor et al.
    Memoirs of General William T. Sherman — Complete by William T. Sherman
- File 11775 is from Human Genome Project, Build 34, Chromosome Number 01 by Human Genome Project - Free Ebook (gutenberg.org) (250mb)
- All tests were ran with minimal background activity and zero user activity while the test was running.
- Sample size of 2001 runs were compiled for each of the 6 results.

Data Structures

- Array:
  - Offers O(1) in terms of accessing which Is useful for moving through all the appearances of the searched for string.
  - Trade-offs are that its O(n) in terms of inserting so will take longer to return all the results from the search

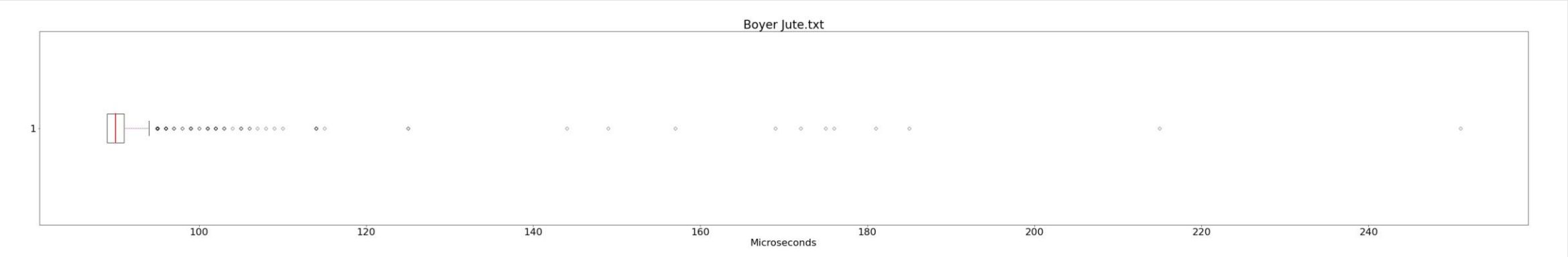
- Lists:
  - Offers O(1) in terms of insertion of results in the Boyer Moore algorithm to be returned to the user after the search is complete.
  - Trade-offs are that its O(n) in terms of accessing so may work better depending on where the pattern occurs in relation to where the user was looking for it.
  - Improves the time of the actual algorithm when searching and adding results.

Boyer-Moore Testing.txt



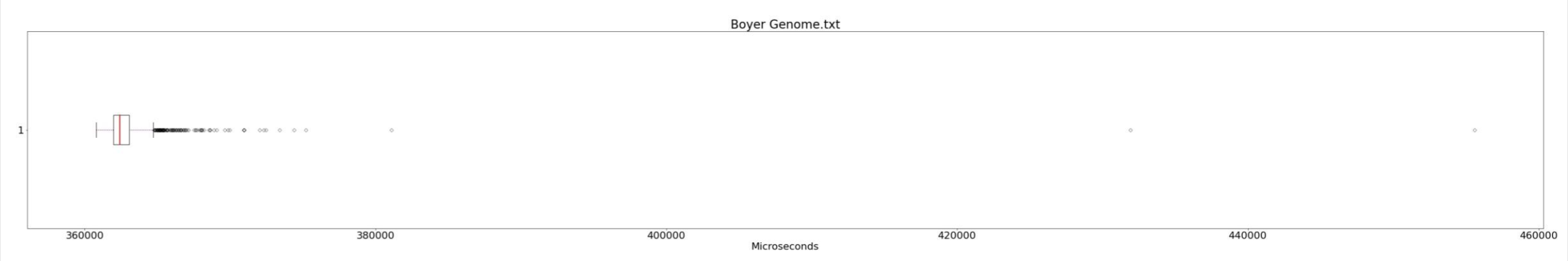
Minimum- 117
Maximum- 1020
Q1-118
Q2(Median)-118
Q3-119
IQR-1
Range-903

Boyer-Moore Jutebook.txt



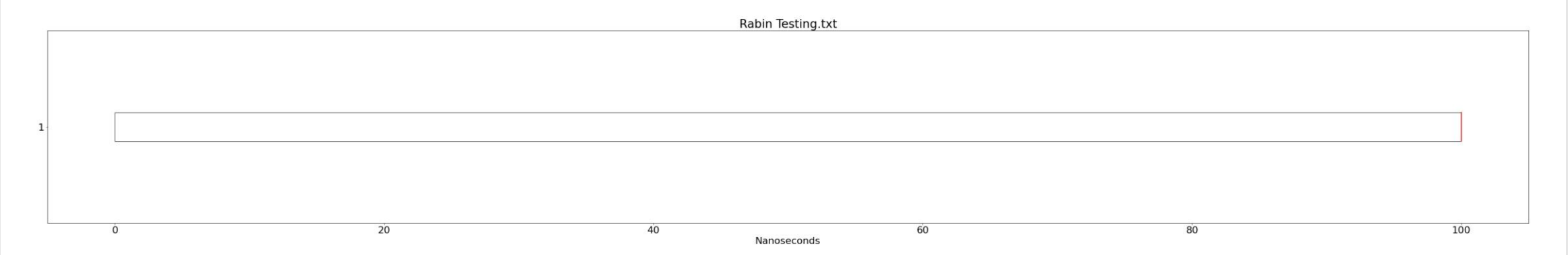
Minimum- 89
Maximum- 251
Q1-89
Q2(Median)-90
Q3-91
IQR-2
Range-162

Boyer-Moore 11775.txt



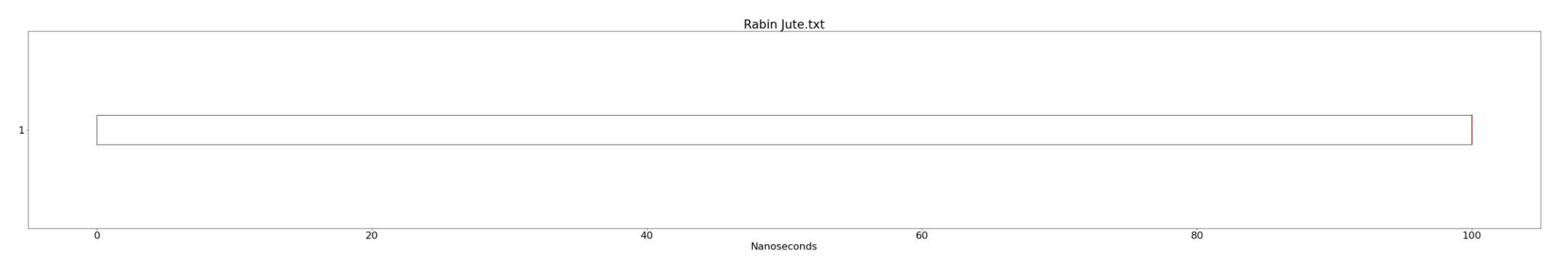
Minimum- 360800
Maximum- 455576
Q1-361984.5
Q2(Median)-362409
Q3-363074.5
IQR-1090
Range-94776

Rabin-Karp Testing.txt



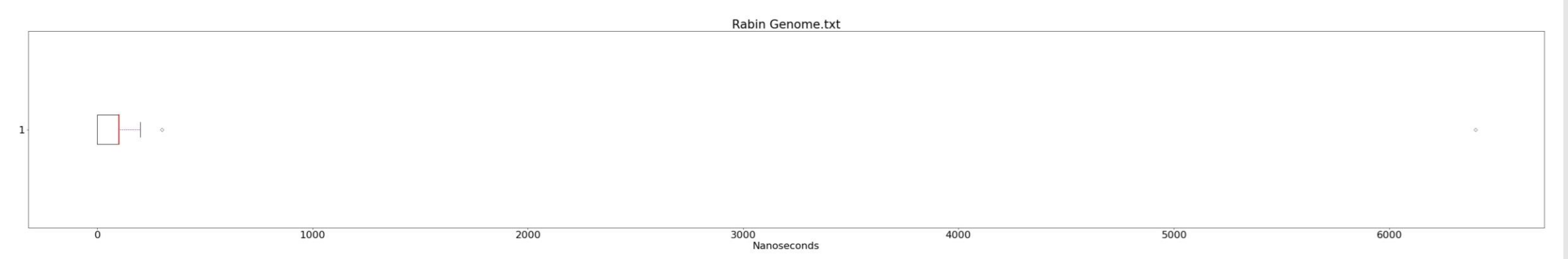
Minimum- 0
Maximum- 100
Q1-0
Q2(Median)-100
Q3-100
IQR-100
Range-100

Rabin-Karp Jutebook.txt



Minimum- 0
Maximum- 100
Q1-0
Q2(Median)-100
Q3-100
IQR-100
Range-100

Rabin-Karp 11775.txt



Minimum- 0
Maximum- 6400
Q1-0
Q2(Median)-100
Q3-100
IQR-100
Range-6400

Comparison

	BM Testing.txt	BM Jutebook.txt	BM 11775.txt	RK Testing.txt	RK Jutebook.txt	RK 11775.Txt
Minimum	117,000	89,000	360,800,000	0	0	0
Maximum	1,020,000	251,000	455,576,000	100	100	6400
Q1	118,000	89,000	361,984,000.5	0	0	0
Q2(Median)	118,000	90,000	362,409,000	100	100	100
Q3	119,000	91,000	363,074,000.5	100	100	100
IQR	1,000	2,000	1,090,000	100	100	100
Range	903,000	162,000	94,776,000	100	100	6400

Comparisons of times are in Nanoseconds

Time Complexity

- All Complexities are in terms of input. M= length of pattern. N= length of text
- Boyer Moore worst case- O(nm)
- Boyer Moore best case- O(n/m)
- File sizes didn't benefit performance significantly, the file composition looks to play a bigger factor on the timings.

- Rabin Karp worst case- O(n)
- Rabin Karp best case- O(n)
- Both file sizes and composition had minimal/zero effect on the timings.

Results

- From testing its noted that Rabin Karp boasted the better results given it was measured in nanoseconds compared to Boyer Moore that was measured in microseconds. (1µs → 1000ns).
- There was no change in timings when file sizes were changed on the Rabin Karp algorithm, hardware limitations prevented the algorithm from working faster.
- There was a difference between Boyer Moore when file sizes were changed but changes were minimal and so could be insignificant.
- During the 11775.txt test, Boyer Moore operated at worst case due to the text that was being searched forcing the algorithm to operate at worst case.
- There was a significant spike during the 11775.txt test for Rabin Karp but due to the minimal appearances of this spike I would deem it as insignificant and caused by external factors.

Thank You for Listening

Any Questions?