Data structures and Algorithms 1

Luke Burgess - 1703091

String Searching Algorithms

- Relevant to Ethical Hacking
  - Use in cryptography
  - Search through log files
  - Find usernames/passwords etc.

- Varied, interesting methods of achieving this goal, chosen for my coursework;
  - Boyer Moore algorithm
  - Rabin Karp algorithm

Boyer Moore Algorithm

- String searching algorithm developed in 1977 by Robert S Boyer and J Strother Moore
- Advanced method compared to brute force algorithm
- The end of the search term or pattern is compared to the texts and if a match is not made, it allows the algorithm to make skips
- As a result, not every character needs to be compared, which reduces the amount of processing required.

Boyer Moore Performance

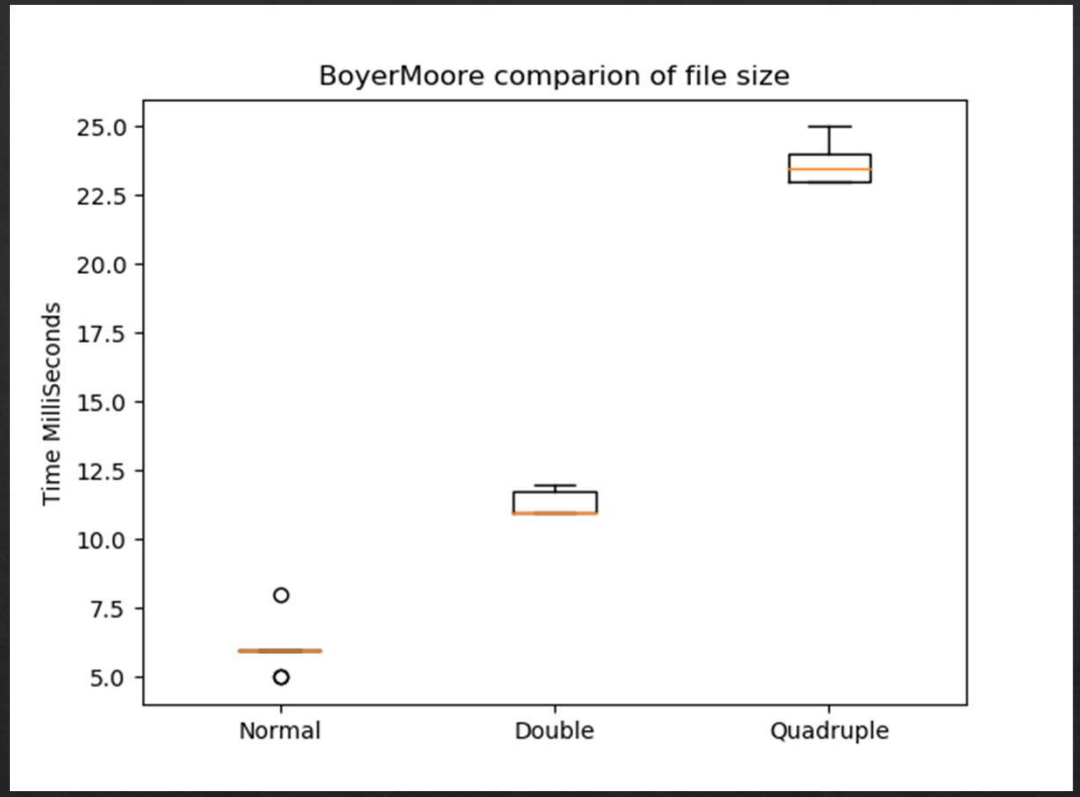
Best Case:

O(N/M) – In each comparison, a match can be ruled out and the length of the pattern can be skipped.

Worst Case:

O(NM) - Unable to skip characters and every single character is compared each time.

where N is the length of the texts, and M is the search term or pattern length.



The graphs produced from testing the Boyer Moore algorithm on sample texts appear to show O(N) notation performance as the time taken increases in a linear fashion with the file size.
Occurrences found for Quadruple size doc: 2880

Rabin Karp Algorithm

- Another string searching algorithm, which was developed in 1987 by Richard M. Karp and Michael O. Rabin.
- This algorithm uses a hash function on strings to create a hash value. The algorithm then checks this hash value against the hash value of the pattern.
- This is efficient however due to the possibility of two different strings having the same hash value, may not be as accurate.
- Used Las Vegas method to provide further feedback on the result of the algorithm.

Rabin Karp Performance

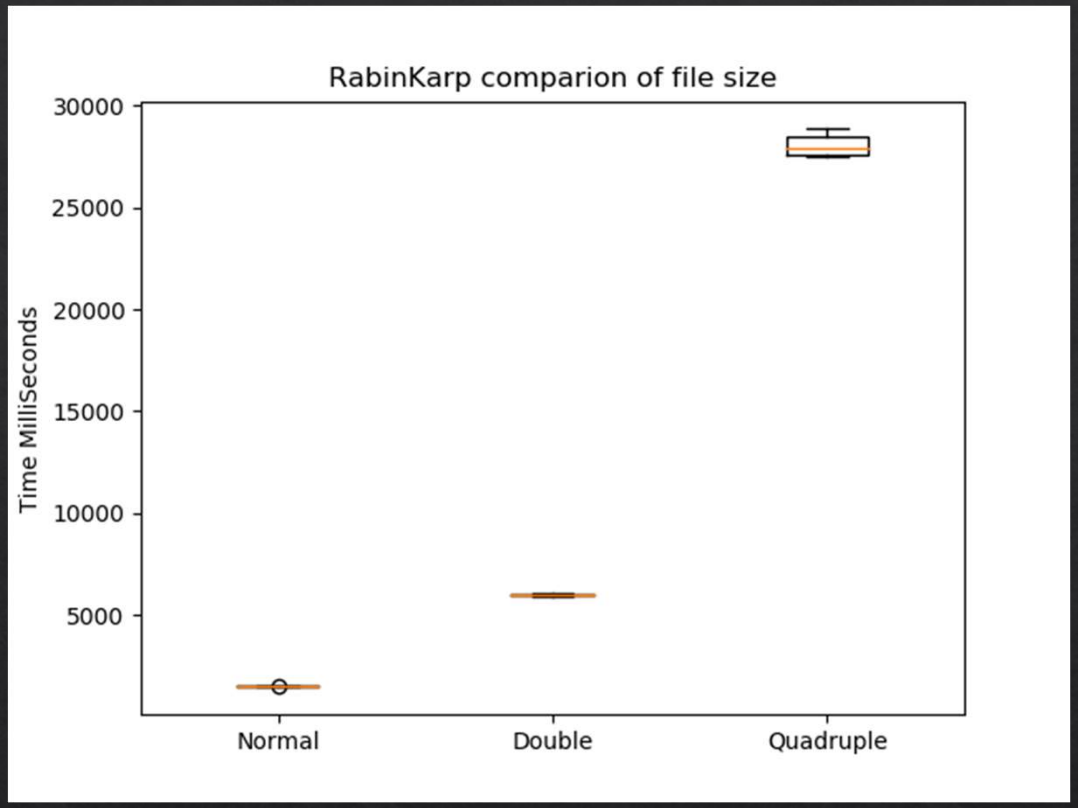
Best Case:

O(N+M) – The average and best case for Rabin Karp.

Worst Case:

O(NM) – This is achieved if all strings and the pattern have the same hash value.

where N is the length of the texts, and M is the search term or pattern length.



The graphs produced from testing the Las Vegas Rabin Karp algorithm appear to show the quadruple file size growing exponentially which could indicate an O(2^N) notation however further testing with more variation of file sizes would be required to confirm this.
Occurrences found for Quadruple size doc: 2880

Data Structures

For both algorithms I opted to use Vectors over lists or arrays due to the following reasons:

- Arrays only allow a fixed length, whereas we may have varied sizes throughout the algorithm so deemed unsuitable.
- Vectors just need to store the elements, whereas lists need to store nodes and pointers, which takes up more memory.
- Vectors have spatial locality as they are contiguous in memory.
- With the .at() function used it can go straight to a vector element as opposed to lists in which you would need to iterate through the list to find the correct element.

Conclusions

- Boyer Moore algorithm took considerable less time to complete
- Not only did Rabin Karp take longer, but when data size quadrupled it appeared to exponentially take much longer than anticipated.
- Both Algorithms found the same number of occurrences of the pattern, however Rabin Karp appears more fallible due to a chance of two strings having the same hash value.
- Throughout testing, each algorithm was run 10 times and a median was taken to eliminate any abnormalities and both algorithms seemed consistent in their task completion time.