Programming Assignment 3

# Overview

The objective of this programming assignment is to demonstrate the differences between the textbook implementation of the QUICKSORT algorithm that leverages the PARTITION method, and a modified version of PARTITION that utilizes median-of-three partitioning to determine a pivot index.

# Part A

The MedianOfThreeMethod method shown below outlines the altered PARTITION method that uses median-of-three partitioning. This method calls upon the CalculateMedianOfThree method, which is responsible for calculating the median-of-three index that is used as the pivot index on a given iteration. Upon calculation of the median-of-three index, the core functionality of the textbook definition of PARTITION is done.

The MedianOfThreeMethod and CalculateMedianOfThree methods can be seen realized in the Partition.py file.





# Part B

## Worst-Case Median-Of-Three Quicksort

The worst-case asymptotic behavior of the QUICKSORT algorithm using median-of-three partitioning occurs when the partition consistently creates highly imbalanced arrays, particularly one array of size and one array of size . This will cause the subproblems to decrease by the smallest amount on each recursive call. In these cases, the recurrence can be defined as:

With this recurrence, we can sum the time complexity incurred on each level to obtain an arithmetic series, which as we know by Equation A.1 and Equation A.2, provides the following asymptotic behavior:

The worst-case asymptotic behavior of the QUICKSORT algorithm using median-of-three partitioning is .

## Benefit Of Median-Of-Three Quicksort

From looking at the current analysis of the median-of-three quicksort, it is not apparently obvious the benefit of the utilizing a median-of-three partition as the partition method in the QUICKSORT algorithm, because they both have the same worst-case asymptotic behavior of . However, because the median-of-three always selects the middle value of three possibles values, we can avoid several cases where the textbook implementation of PARTITION returns a highly imbalanced array, but the median-of-three implementation of PARTITION would not. This means that the median-of-three method for the PARTITION algorithm allows us to bring the average-case asymptotic running time down.

# Part C

Performing a QUICKSORT algorithm using median-of-three partitioning on an input set that is already sorted will cause a worse-case asymptotic behavior of . This is because this case will result in the largely imbalanced partitions mentioned in Part B, that is, one array of size and one array of size

# Part D

See …\Source\Python…

# Part E

Implementations of both algorithms are done by using test runs, which count the number of key points for varying numbers of input values, . These are further represented as graph is Part G.

The worst-case metrics for the implementation of the QUICKSORT algorithm using the textbook implementation of the PARTITION method are shown in the text file:

…\Source\TestRun\_MedianOfThreeMethod\_Output.txt.

The worst-case metrics for the implementation of the QUICKSORT algorithm using the textbook implementation of the PARTITION method are shown in the following text file: …\Source\TestRun\_MedianOfThreeMethod\_Output.txt

# Part F

See …\Source\Python…

# Part G

Figure 1 highlights the comparison of asymptotic worst-case, , to the expiramental behavior of the textbook implementation of the QUICKSORT algorithm.

A graph of a graph with blue dots and a red line

AI-generated content may be incorrect.

Figure 1

Figure 2 highlights the comparison of asymptotic worst-case, , to the expiramental behavior of the implementation of the QUICKSORT algorithm.

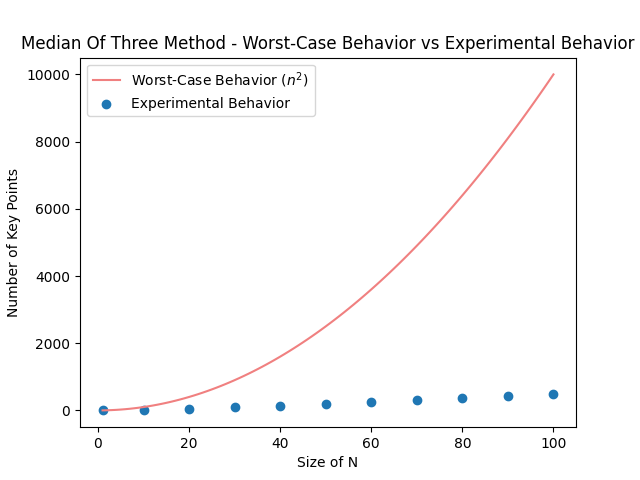


Figure 2

Figure 3 highlights the comparison of average-case, , to the expiramental behavior of the median-of-three implementation of the QUICKSORT algorithm.

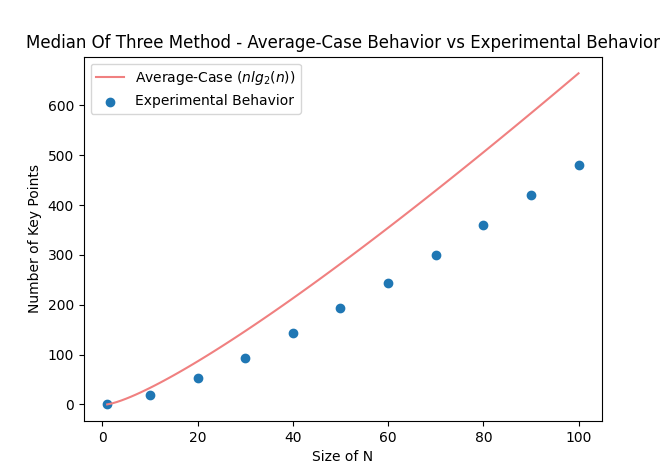


Figure 3

# References

*Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2022). Introduction to Algorithms (Adobe Digital Editions). MIT Press. Section 7.2 and 7.3.*

*Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2022). Introduction to Algorithms (Adobe Digital Editions). MIT Press. Page 1141.*