Polytechnic University of the Philippines

College of Engineering and Architecture

Sta. Mesa Campus, Manila

Data Structures and Algorithms

CMPE 201

Group Laboratory Exercise

Search Algorithms

Submitted by:

Group 1

Batin, Katrina Ricci

Budoy, Christian Kenneth

Co, Angel

De Leon, Rose Denise

Hernandez, Dan Efraim

Lobusta, Vanica

Ronio, Ryza Mae

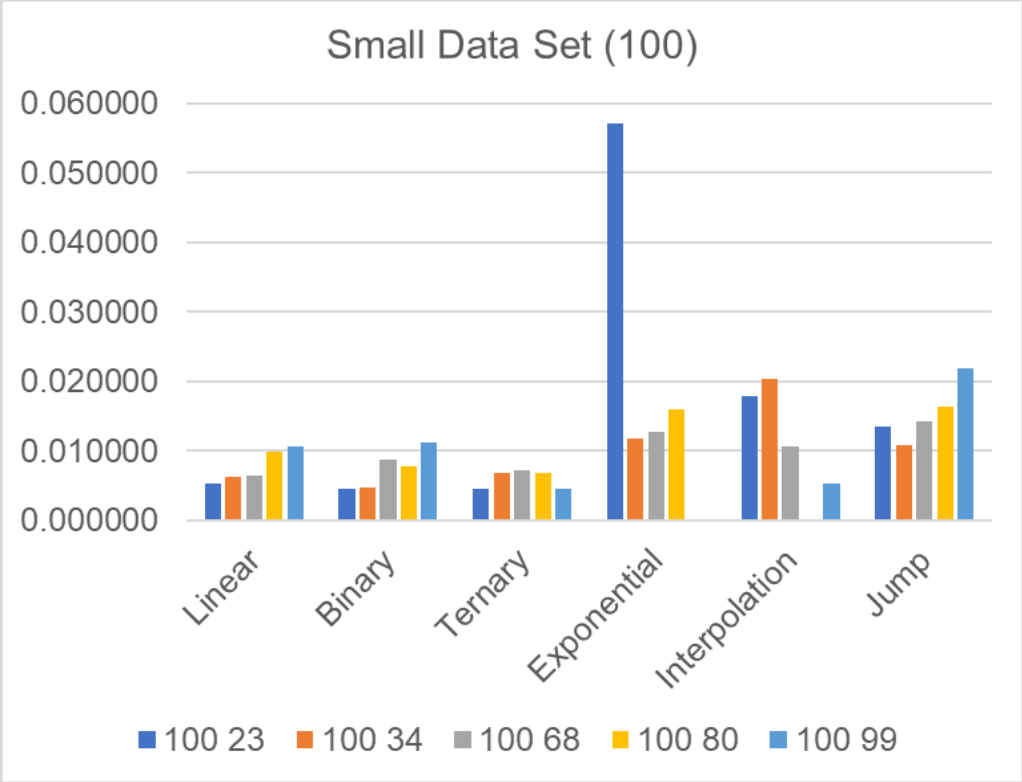
Ybanez, Jiro

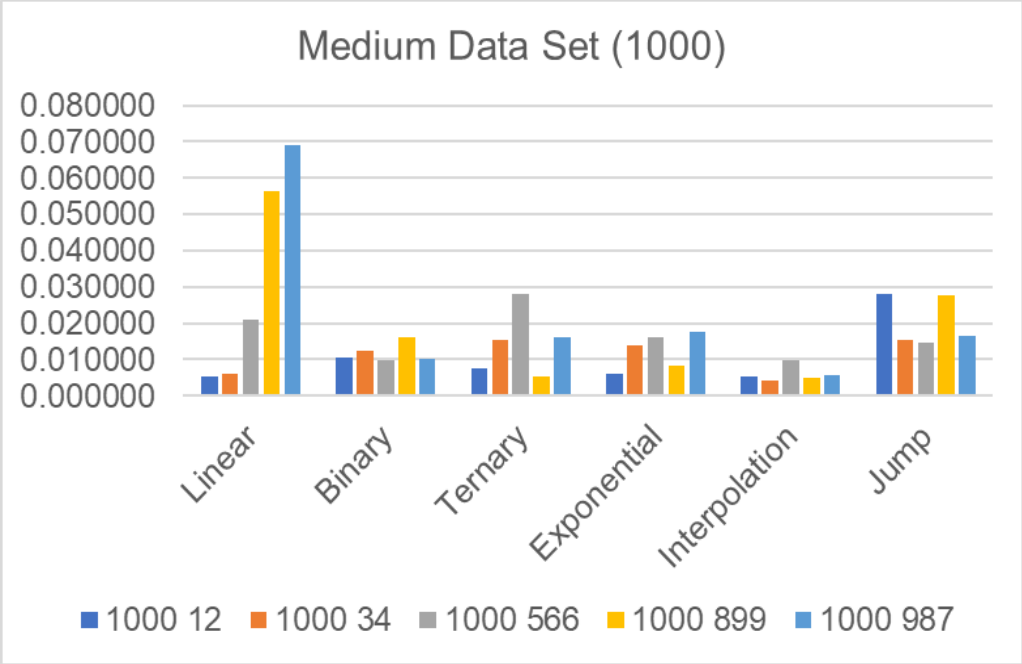
Submitted to:

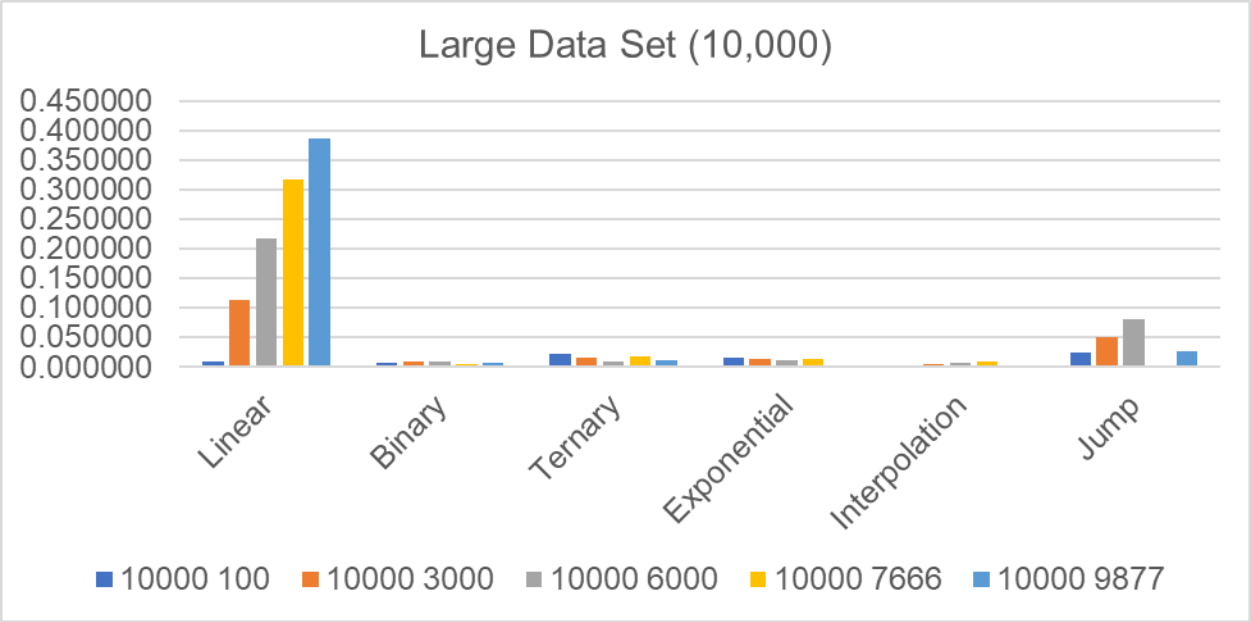
Prof. Godofredo T. Avena

Search Algorithms Analysis Worksheet

		Linear	Binary	Ternary	Exponential	Interpolation	Jump
Target Set	Search data	Time in Milliseconds					
100	23	0.005300	0.004599	0.004499	0.057000	0.017899	0.013500
	34	0.006299	0.004699	0.006900	0.011799	0.020299	0.010899
	68	0.006399	0.008699	0.007299	0.012800	0.010599	0.014300
	80	0.009900	0.007800	0.006899	0.015900	0.009499	0.016400
	99	0.010599	0.011299	0.004599	0.013099	0.005400	0.021800
		Linear	Binary	Ternary	Exponential	Interpolation	Jump
Target Set	Search data	Time in Milliseconds					
1000	12	0.005300	0.010699	0.007699	0.006100	0.005300	0.028200
	34	0.006100	0.012299	0.015400	0.013890	0.004300	0.015199
	566	0.021100	0.009700	0.028100	0.016299	0.009799	0.014600
	899	0.056400	0.015999	0.005299	0.008399	0.004900	0.027800
	987	0.069199	0.010099	0.015999	0.017700	0.005699	0.016600
		Linear	Binary	Ternary	Exponential	Interpolation	Jump
Target Set	Search data	Time in Milliseconds					
10000	100	0.009600	0.006999	0.022600	0.016199	0.004499	0.023600
	3000	0.112499	0.008099	0.014600	0.014200	0.005799	0.050299
	6000	0.218200	0.008000	0.009300	0.011000	0.005999	0.079900
	7666	0.316999	0.005799	0.017599	0.012900	0.008299	0.017499
	9877	0.386199	0.006700	0.011900	0.011800	0.003499	0.026500


Small Data Set (100)
0.060000
0.050000
0.040000
0.030000
0.020000
0.010000
0.000000
Linear
Binary
Ternary
Exponential
Interpolation
Jump
100 23
100 34
100 68
100 80
100 99


Medium Data Set (1000)
0.080000
0.070000
0.060000
0.050000
0.040000
0.030000
0.020000
0.010000
0.000000
Linear
Binary
Ternary
Exponential
Interpolation
Jump
1000 12
1000 34
1000 566
1000 899
1000 987



Analyze the results and answer the following questions:

a. Which search algorithm performed the best overall?

Based on their overall performance, both "Interpolation" and "Linear" algorithms tends to perform well across different target sets and search data sizes. However, the "Linear" algorithm also shows good performance in certain cases. In short, the Interpolation search algorithm performed the best overall. It has the shortest execution time for all problem sizes, except for the smallest problem size (100 elements), where it is tied with the Binary search algorithm.

b. Did any search algorithms perform better on specific data sets?

Yes. By looking at the tables, we can see that some search algorithms performed better on specific data sets. For example, on the data set with 1000 elements, the jump search algorithm performed the best, while on the data set with 100 elements, the binary search algorithm performed the best. However, it is important to note that the performance of each search algorithm can vary depending on the specific data set and the target set being searched for.

c. How did the size of the data set affect the performance of the search algorithms?

Looking at the tables, we can see that the size of the data set had a significant impact on the performance of the search algorithms. In general, as the size of the data set increased, the time it took to search for a target set also increased for all search algorithms. However, some search algorithms were affected more than others. For example, the linear search algorithm had a much larger increase in search time as the data set size increased compared to the binary search algorithm. Additionally, the jump search algorithm performed significantly better than other search algorithms on larger data sets. Overall, the size of the data set had a significant impact on the performance of the search algorithms, and it is important to consider the size of the data set when selecting a search algorithm..

d. Write a brief conclusion summarizing your findings

The size of the data set had a significant impact on the performance of the search algorithms, with larger data sets generally taking longer to search. Some search algorithms performed better than others on specific data sets. **Linear** search, known for its simplicity is expected to **perform well on small datasets** of up to 100 elements but may exhibit noticeable inefficiencies as the dataset grows due to its linear time complexity. On the other hand, **binary** search with its logarithmic time complexity, remains **consistently efficient** scaling well even for datasets of up to 10,000 elements. **Exponential** search, same to binary search but potentially with a slightly higher constant factor while **maintaining good performance** across small, medium, and large datasets. **Interpolation** search, particularly **effective for uniformly distributed data** and is anticipated to perform well across varying dataset sizes. **Jump** search, efficient for sorted datasets and is expected to remain **effective on both medium and large datasets**. Lastly, **ternary** search with its logarithmic time complexity is projected to perform well on small and medium datasets, continuing to demonstrate efficiency as the dataset size increases to 10,000 elements.

It is important to consider the size of the data set and the specific target set being searched for when selecting a search algorithm.