Burak Yetiştiren

Section-2

21802608

CS201

HW2

Computer Specifications

Model: MSI GF75 Thin 8SC-206XTR

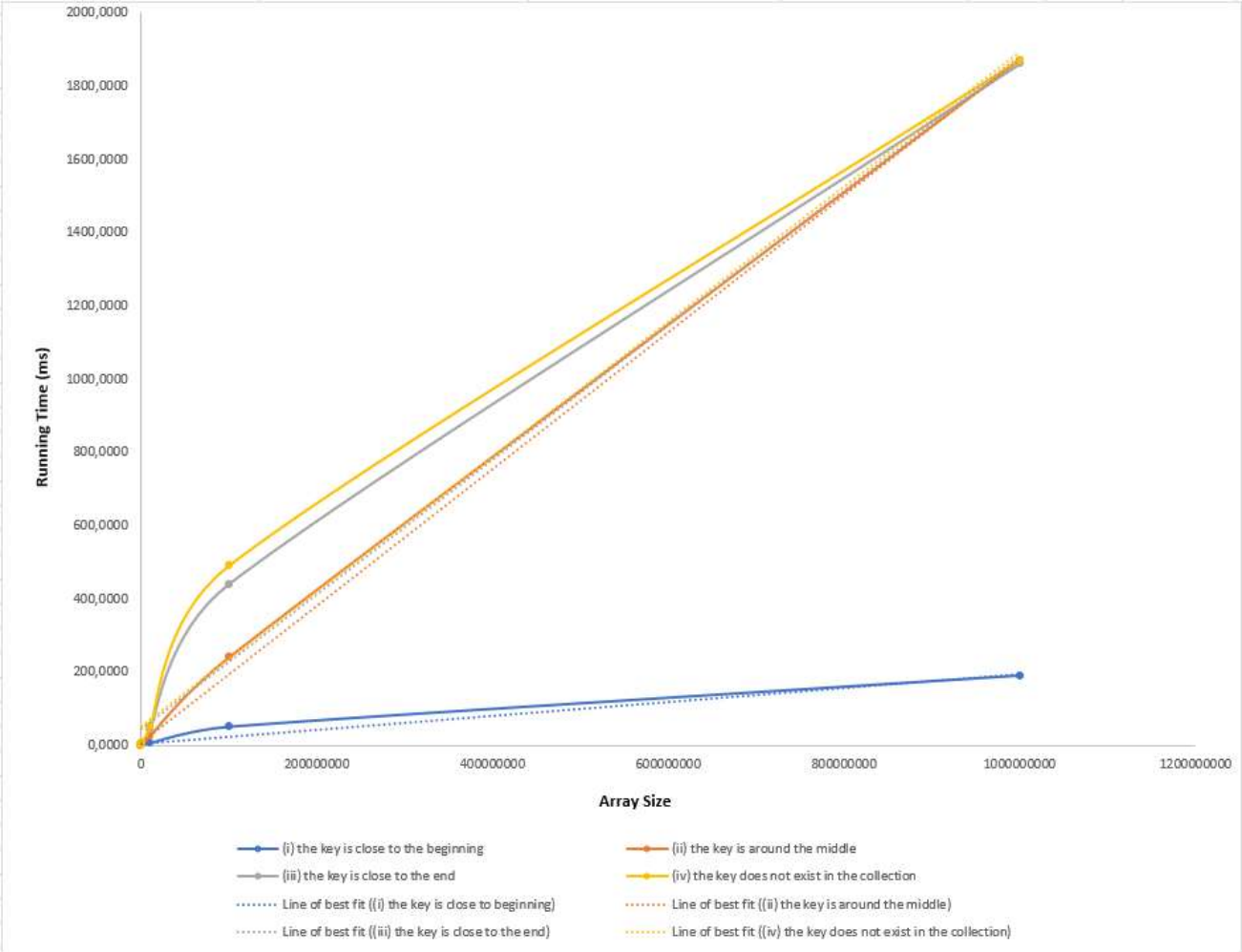
Processor: 4.1 GHz Intel Core i7

Memory: 8GB 2666 MHz DDR4

Storage: 5400 RPM 1TB HDD 256 GB SSD

Array Size	(i) the key is close to the beginning (10% of array size)	(ii) the key is around the middle (50% of array size)	(iii) the key is close to the end (90% of array size)	(iv) the key does not exist in the collection (110% of array size)
1000	0,0006	0,0024	0,0040	0,0050
10000	0,0050	0,0200	0,0050	0,0400
100000	0,0500	0,2400	0,4400	0,4800
1000000	0,5000	2,4000	4,4000	4,8000
10000000	5,0000	25,0000	44,0000	49,0000
100000000	50,0000	240,0000	440,0000	490,0000
1000000000	190,0000	1870,0000	1860,0000	1870,0000

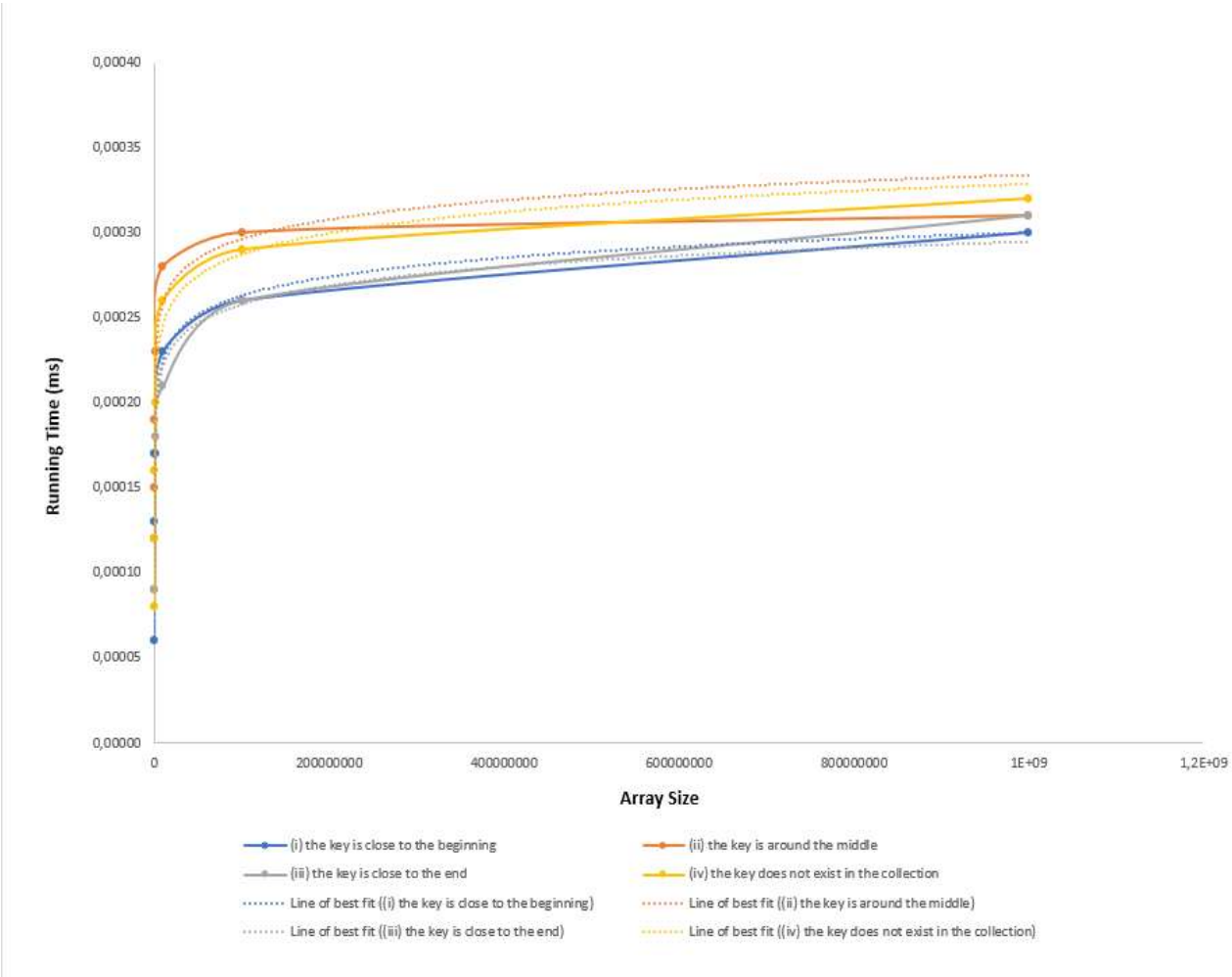
➢ Table 1: Time values in milliseconds to perform a **sequential search** in different array sizes depending the position of the key (target value).



➢ Figure 1: Graph of experimental time values for **sequential search** according to the position of the key (target value).

Array Size	(i) the key is close to the beginning (10% of array size)	(ii) the key is around the middle (50% of array size)	(iii) the key is close to the end (90% of array size)	(iv) the key does not exist in the collection (110% of array size)
1000	0,00006	0,00009	0,00009	0,00008
10000	0,00013	0,00015	0,00012	0,00012
100000	0,00017	0,00019	0,00012	0,00016
1000000	0,00017	0,00023	0,00018	0,00020
10000000	0,00023	0,00028	0,00021	0,00026
100000000	0,00026	0,00030	0,00026	0,00029
1000000000	0,00030	0,00031	0,00031	0,00032

➢ Table 2: Time values in milliseconds to perform a **binary search** in different array sizes depending the position of the key (target value).



➢ Figure 2: Graph of experimental time values for **binary search** according to the position of the key (target value).

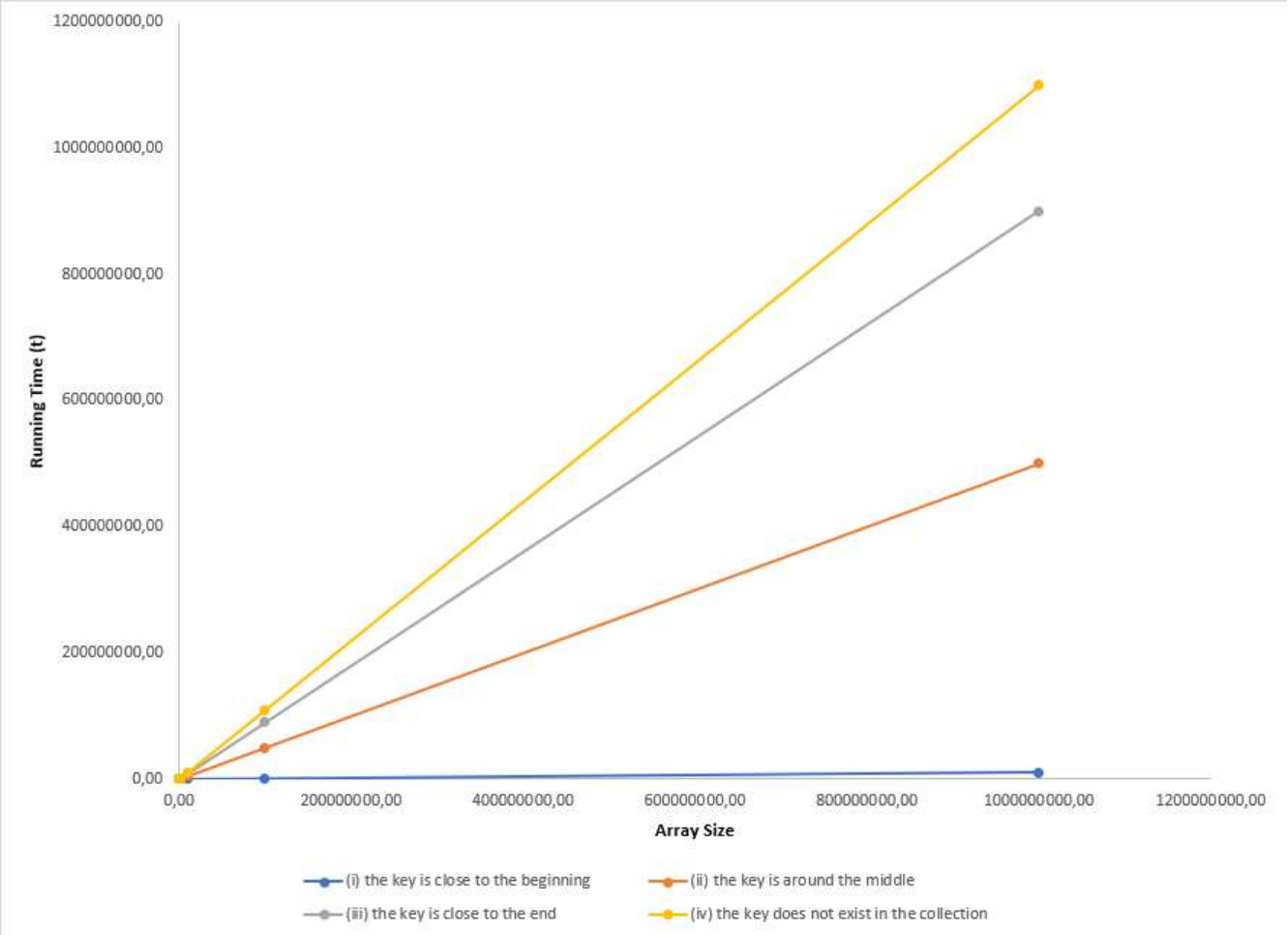
Best, average and worst case:

In **sequential search**, when looked at the graph, the **best case** appears to be that when the key (target value) is close to the beginning, the **average case** when the key is in the middle and the **worst case** is when the key does not exist in the collection.

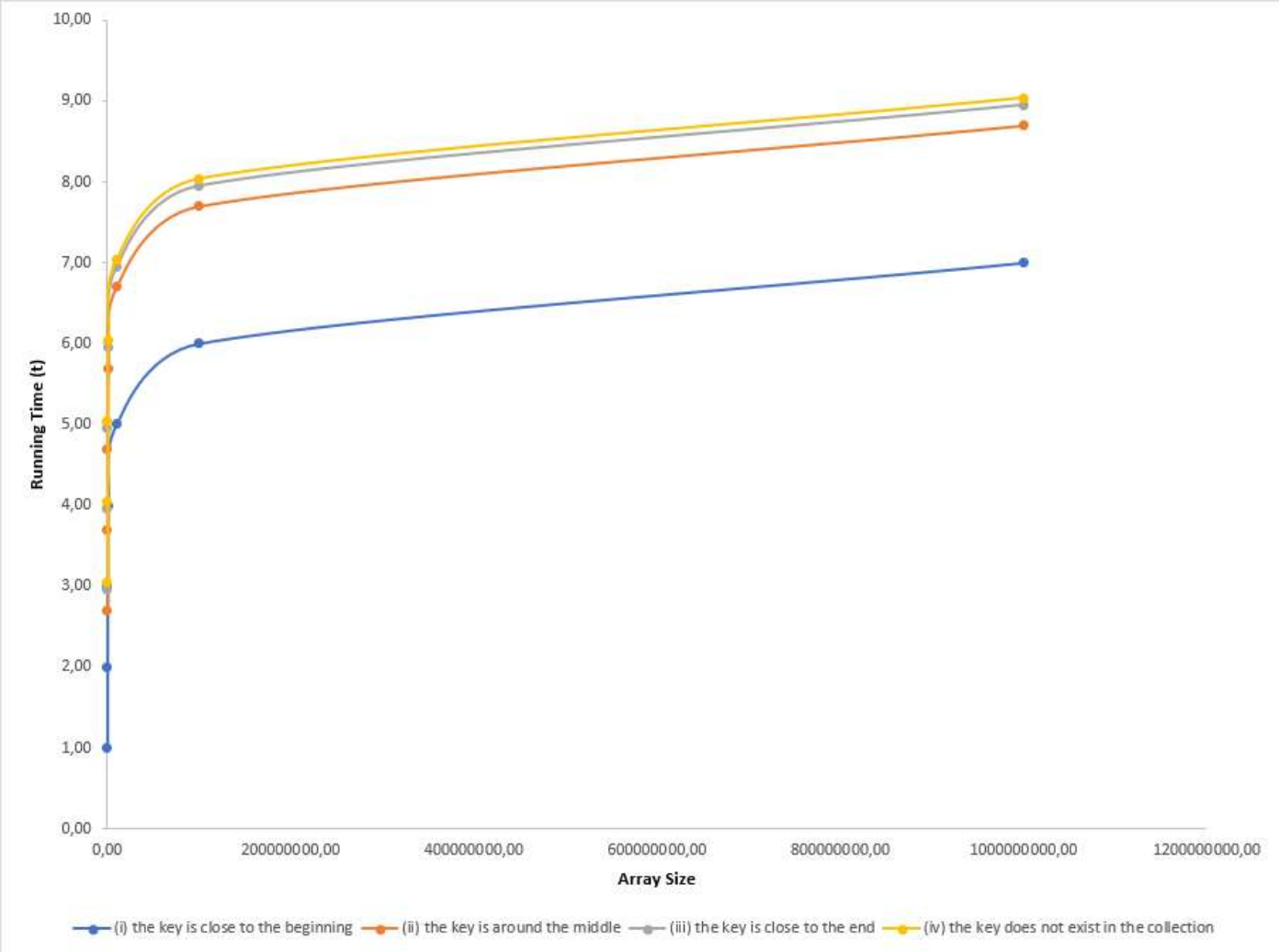
In **binary search**, when looked at the graph, all of the cases are closer to each other than it was in sequential search. Cases line up in an ascending order according to spent time as:

1. When the key is close the end
2. When the key is close to the beginning
3. When the key does not exist
4. When the key is in the middle

Hence, our **best case** is when the key is in the end, our **average case** is close to the beginning and our **worst case** is when the key is in the middle.



➢ Figure 3: Graph of theoretical time values for **sequential search** according to the position of the key (target value).



➢ Figure 4: Graph of theoretical time values for **binary search** according to the position of the key (target value).

Conclusion:

When experimental and theoretical graphs are contrasted, for **sequential search** the ranking of running time does not change (in an ascending order):

1. When the key is close to the beginning
2. When the key is in the middle
3. When the key is close the end
4. When the key does not exist

This proves that, although the graph experimental results does not have the same structure with the graph of theoretical values, the results are accurate as they rank in the same order with theoretical results. The structure of the experimental results for the **sequential search** is not perfectly linear, but

the closest structure of the graph would be a linear plot, which is excepted, as the time complexity of the **sequential search** is O(N), where N is the size of the input. The error in the experimental results for **sequential search,** however, can be caused from the computational errors that the computer the program is executed on or the different behavior of small inputs, as the most corruption in the graph when compared to the theoretical graph is mostly in the beginning, where the small inputs are given as inputs.

In **binary search**, the theoretical values do not line up in the same way with the experimental results, but as the experimental values are close to each other, the error in the experimental graph may be caused by some external factors caused by the computer the program was executed in. On the other hand, the structure of both graphs of **binary search** are the same and they are in the shape of a logarithmic graph, which was excepted as the time complexity of binary search is O(logN), where N represents the input size.