

## SUMMARY

USC ID/s:  
2853724015

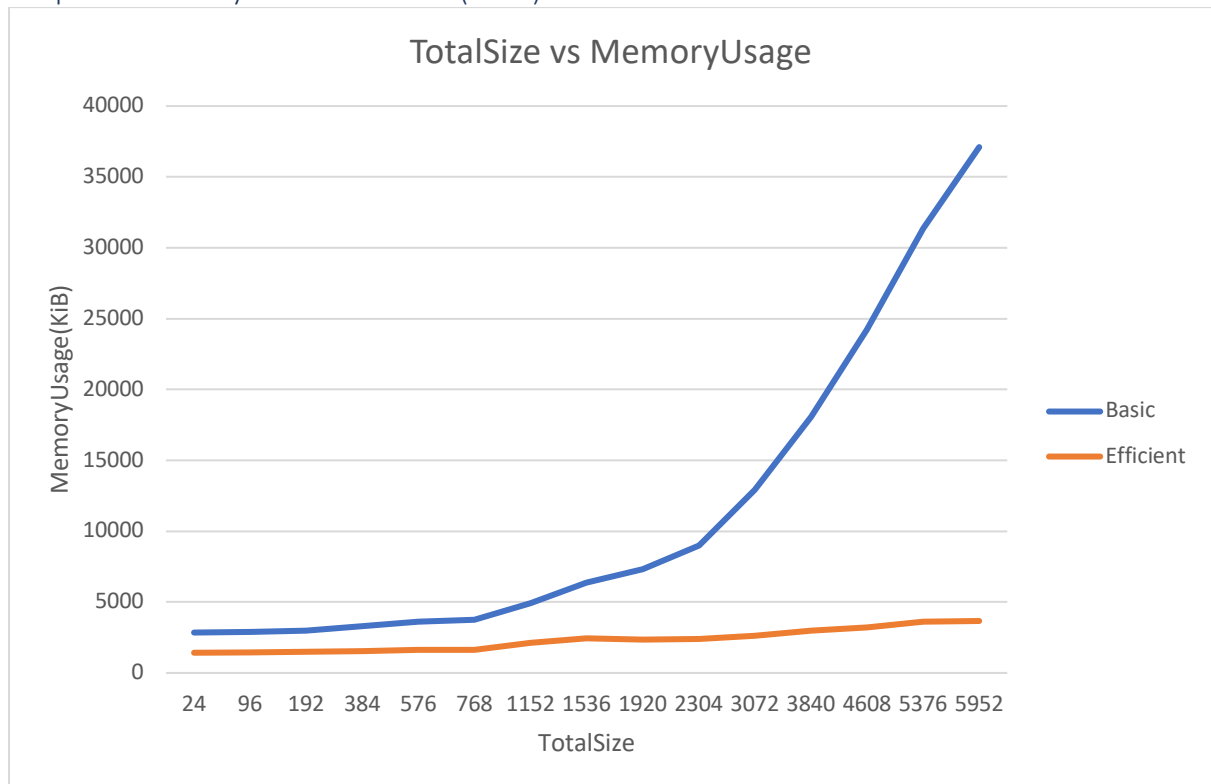
### Datapoints

M+N	Time in MS (Basic)	Time in MS (Efficient)	Memory in KB (Basic)	Memory in KB (Efficient)
24	0.273	0.083	1424	1424
96	0.474	0.501	1424	1456
192	0.72	1.492	1488	1504
384	2.773	5.141	1744	1536
576	5.401	10.432	2016	1616
768	8.442	16.492	2096	1632
1152	16.703	35.432	2800	2128
1536	25.919	64.016	3952	2432
1920	43.767	94.457	4976	2336
2304	57.602	124.701	6592	2416
3072	123.1	243.191	10320	2608
3840	168.151	362.338	15072	2992
4608	246.269	478.232	20992	3216
5376	297.722	731.405	27728	3632
5952	429.487	869.705	33440	3664

### Insights

We can see that during the first sequences with small size, the execution time and memory usage won't be different that much. However, as the sequences become larger, the memory usage of efficient method can be 10 times smaller than the basic method, which is very impressive. In addition, the execution time of the efficient method becomes about twice as the basic method, which similar to the analysis of the time complexity  $O(2 * C mn)$  vs  $O(mn)$

Graph1 – Memory vs Problem Size (M+N)



*Nature of the Graph (Logarithmic/ Linear/ Polynomial/ Exponential)*

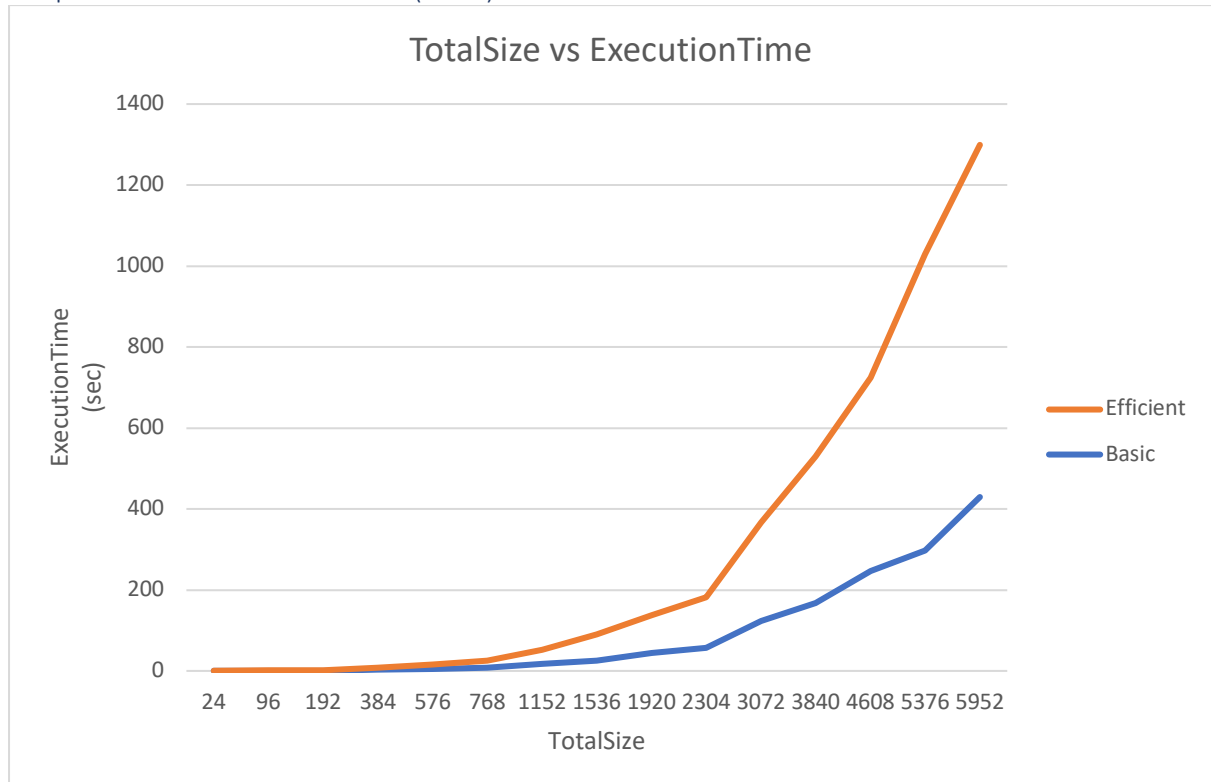
**Basic:** Exponential

**Efficient:** Linear

*Explanation:*

For efficient method is  $O(m + n)$ , where  $m$  = length of sequence 1,  $n$  = length of sequence 2. This is because we apply recursive calls sequentially and reuse the working space from one call to the next.

Graph2 – Time vs Problem Size (M+N)



*Nature of the Graph (Logarithmic/ Linear/ Polynomial/ Exponential)*

Basic: Polynomial

Efficient: Exponential

*Explanation:*

When the sequence become much more bigger, the efficient method have to divide several times to cut down the problem, and that process may affect the total execution time than basic method, which deal with the sequence directly.

*Contribution*

(Please mention what each member did if you think everyone in the group does not have an equal contribution, otherwise, write "Equal Contribution")

<USC ID/s>: <Equal Contribution>

2853724015: I did it myself and I really enjoy the process