**SUMMARY**

## Datapoints

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| M+N | Time in MS (Basic) | Time in MS (Efficient) | Memory in KB (Basic) | Memory in KB (Efficient) |
| 16 | 0.000 | 0.0 | 14520320 | 14316 |
| 64 | 0.997 | 0.996 | 14573568 | 14276 |
| 128 | 2.992 | 4.987 | 14639104 | 14256 |
| 256 | 7.978 | 19.946 | 15208448 | 14364 |
| 384 | 16.954 | 59.841 | 15859712 | 14352 |
| 512 | 29.920 | 77.826 | 15683584 | 14244 |
| 768 | 68.819 | 178.522 | 15421440 | 14496 |
| 1024 | 125.664 | 331.149 | 15671296 | 14368 |
| 1280 | 208.443 | 514.657 | 15638528 | 14360 |
| 1536 | 299.200 | 751.026 | 15851520 | 14692 |
| 2048 | 531.578 | 1677.514 | 16076800 | 14728 |
| 2560 | 947.467 | 2026.588 | 16486400 | 14788 |
| 3072 | 1304.512 | 2990.038 | 16388096 | 14972 |
| 3584 | 1795.200 | 4872.938 | 15634432 | 14920 |
| 3968 | 2593.068 | 5576.123 | 15904768 | 15072 |

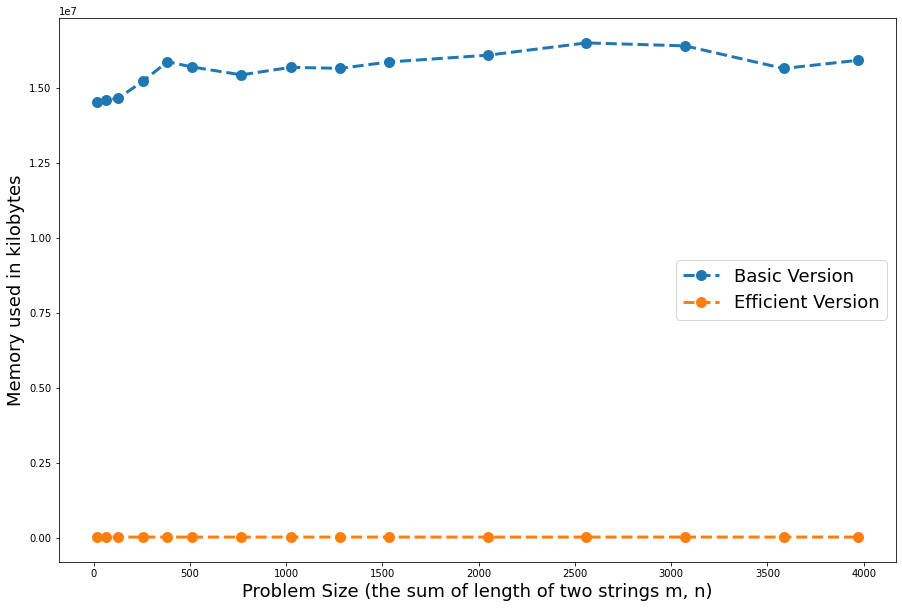
## Insights

Compared to the Basic method, Efficient method consumes more time and occupies less memory.

When the problem size is small, the difference in time consumption between the two methods is not obvious. When the size of the problem becomes larger, the time consumption of the Efficient method is significantly higher than that of the Basic method, approaching twice of Basic method.

However, in terms of memory usage, the Efficient method always occupies much less than the Basic method.

### Graph1 – Memory vs Problem Size (M+N)



#### Nature of the Graph (Logarithmic/ Linear/ Exponential)

Basic: Polynomial

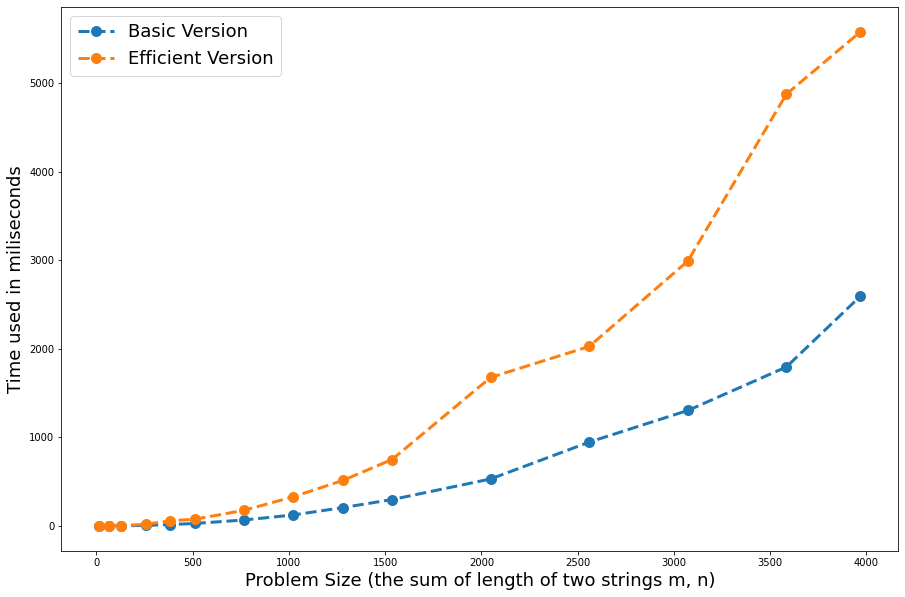
Efficient: Linear

#### Explanation:

The space occupation of Basic method is O(m·n), which is polynomial in the variables m and n.

The space occupation of Basic method is O(m+n), which is linear, since the coefficients of m and n are constants.

### Graph2 – Time vs Problem Size (M+N)



#### Nature of the Graph (Logarithmic/ Linear/ Exponential)

Basic: Polynomial

Efficient: Polynomial

#### Explanation:

The time consumption of both method is O(m·n). However, Efficient method is about twice of Basic method.