**Results of comparing the pure binary search with its 1/5 - 4/5 variation**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **List** | **Number of elements in the list** | **Search key** | **Success or failure?** | **In what part of the array was it (if success) or was it meant to be (if failure)?** | **Number of comparisons - 1/5 - 4/5 strategy** | **Number of comparisons - pure binary** |
| List 1 | 1100 | 5146 | Success | within 1/5th (position 104) | 9 | 10 |
| List 2 | 2000 | 25070 | Failure | within 2/5 (around position 513) | 12 | 12 |
| List 3 | 3000 | 81318 | Success | close to the middle (position 1590) | 15 | 11 |
| List 4 | 4000 | 88260 | Failure | within 3/5th (around position 1735) | 17 | 13 |
| List 5 | 5000 | 194072 | Success | within 4/5th (position 3825) | 17 | 12 |
| List 6 | 6000 | 261693 | Failure | within 5/5th (around position 5137) | 18 | 14 |
| List 7 | 7000 | 30789 | Failure | within 1/5th (around position 624) | 11 | 14 |
| List 8 | 8000 | 129017 | Success | within 2/5 (position 2556) | 17 | 13 |
| List 9 | 9000 | 215198 | Failure | close to the middle (around position 4323) | 16 | 14 |
| List 10 | 10000 | 280058 | Success | within 3/5th (position 5555) | 15 | 13 |
| List 11 | 11000 | 394381 | Failure | within 4/5th (around position 7777) | 14 | 14 |
| List 12 | 12000 | 560417 | Success | within 5/5th (position 11111) | 14 | 13 |

Legend: numbers in green show that this type of search is better (faster) than the other type; yellow numbers show that both searches are identical; red numbers show that this type of search is slower.

**Conclusion: the results show that if the search key is in the first 1/5th of the array, then the 1/5 - 4/5 strategy is always better (faster) than the pure binary search. However, if the search key is in the last 4/5th of the array, then the pure binary search is better (faster) or the same as the 1/5 - 4/5 strategy.**