The following materials have been collected from the numerous sources including my own and my students over the years of teaching and experiences of programming. Please help me to keep this tutorial up-to-date by reporting any issues or questions. Please send any comments or criticisms to [idebtor@gmail.com](mailto:idebtor@gmail.com). Your assistances and comments will be appreciated.

A doubly linked list with sentinel nodes(ver 2.)

# Step 8. Test scores

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| N | | 10,000 | 100,000 | 1,000,000 |  |
| Pop\_all  O(n) | my code | 0.000127 sec | 0.000797 sec | 0.00603 sec |  |
| listdsx | 7.9e-05 sec | 0.000821 sec | 0.006581 sec |  |
| unique  O(n) | my code | 0.00062 sec | 0.004819 sec | 0.05776 sec |  |
| listdsx | 0.000414 sec | 0.002406 sec | 0.021543 sec |  |
| selection sort  O(n^2) | my code | 0.162987 sec | 16.0236 sec | xxxx | takes too long unless use quicksort |
| listdsx | 0.213495 sec | 21.2948 sec | xxxx |
| reverse  O(n) | my code | 0.000192 sec | 0.001512 sec | 0.013911 sec |  |
| listdsx | 0.000267 sec | 0.001684 sec | 0.01765 sec |  |
| Shuffle/half  O(n) | my code | 0.00025 sec | 0.001938 sec | 0.02017 sec |  |
| listdsx | 0.000172 sec | 0.00129 sec | 0.017868 sec |  |
| push sorted  O(n) | my code | 0.00025 sec | 0.001425 sec | 0.014054 sec |  |
| listdsx | 0.000221 sec | 0.00443 sec | 0.063345 sec |  |
| push sortedN  O(n^2) | my code | 0.150513 sec | 19.2364 sec | More than 150sec | N =  10, 000,  100,000, 1,000,000 |
| listdsx | 0.119145 sec | 17.0234 sec | 39.3045 sec |
| push sortedN  O(n log n) | my code | 0.001952 sec | 0.018693 sec | 0.168583 sec |
| listdsx | 0.002165 sec | 0.016676 sec | 0.171754 sec |