

Data Structures

Projects 5: Sorting Exercise Worksheet

Due Date: 2009.01.06 (anytime before midnight 2009.01.06)

1. Record the number of comparisons made using different sorting algorithms.

| Data Set | Data Size | Insertion Sort | Heap Sort | Quick Sort | Merge Sort |
|----------|-----------|----------------|-----------|------------|------------|
| 1 | 10 | 36 | 37 | 34 | 26 |
| 2 | 50 | 634 | 411 | 291 | 236 |
| 3 | 100 | 2892 | 1036 | 715 | 559 |
| 4 | 500 | 9721 | 7816 | 4449 | 2699 |
| 5 | 1,000 | 317005 | 17173 | 11573 | 6684 |
| 6 | 5,000 | 7150220 | 107909 | 75360 | 55130 |
| 7 | 10,000 | 26218201 | 235614 | 161668 | 122426 |
| 8 | 50,000 | 633290015 | 1410405 | 986859 | 731814 |
| 9 | 100,000 | 2500864941 | 3019785 | 2058509 | 1565991 |
| 10 | 500,000 | 62532948700 | 17397368 | 11475186 | 8849113 |

2. Record the number of swaps made using different sorting algorithms.

| Data Set | Data Size | Insertion Sort | Heap Sort | Quick Sort | Merge Sort |
|----------|-----------|----------------|-----------|------------|------------|
| 1 | 10 | 27 | 25 | 10 | 36 |
| 2 | 50 | 585 | 243 | 65 | 294 |
| 3 | 100 | 2793 | 583 | 166 | 688 |
| 4 | 500 | 9222 | 4385 | 1908 | 4492 |
| 5 | 1,000 | 316006 | 9095 | 3379 | 9984 |
| 6 | 5,000 | 7145221 | 57202 | 15302 | 63160 |
| 7 | 10,000 | 26208202 | 124330 | 32119 | 136320 |
| 8 | 50,000 | 633240016 | 737793 | 189169 | 796496 |
| 9 | 100,000 | 2500764942 | 1575122 | 410708 | 1692992 |
| 10 | 500,000 | 62532448701 | 9024634 | 2488637 | 9490272 |

3. What is the theoretical time complexity of each of the above sorting algorithms?
Do the above experiments confirm with those theoretical bounds? Please explain.

Theoretical time complexity

| Sorting algorithm | Insertion Sort | Heap Sort | Quick Sort | Merge Sort |
|-------------------|----------------|---------------|---------------|---------------|
| Time complexity | $O(n^2)$ | $O(n \log n)$ | $O(n \log n)$ | $O(n \log n)$ |

Insertion Sort：是，但是在 1000 筆資料的時候高於理論值

Heap Sort：是，但是比較次數明顯高於理論值

Quick Sort：是，但是在 1000 筆資料的時候開始超越理論值，
交換次數明顯小於理論值

Merge Sort：大致合於理論值

4. From the above two experiments, explain the efficiency of the different sorting algorithms.

共通點：當資料量越大，時間複雜度越趨於理論值

Insertion Sort：最好情況就是，序列已經是升序排列， $O(n)$ 。最壞情況就是，序列是降序排列， $O(n^2)$ 。插入排序適合用於數據量很小(<1000)的排序。

Heap Sort：需要一直去調整 heap 使其常為已排序狀態，比較次數太多

Quick Sort：此種排序方法是不穩定的，且其時間複雜度在平均狀況時為 $O(n \log n)$ ，最壞情況其時間複雜度為 $O(n^2)$ 。但是他是最有效率的 sorting algorithms.

Merge Sort：若是不需要交換太多次的序列，用 Merge Sort 是個不錯的選擇，而且它是穩定的排序方法 $O(n \log n)$