**資料結構 #7**

**Sorting 演算法執行效率**

學號: B113040052

名字: 陳育霖

A 類排序演算法: 1. Selection sort

B 類排序演算法: 2. Merge sort

C 類排序演算法: 1. Radix sort

D 類排序演算法: C qsort()函式

E 類排序演算法: C++ sort()函式

A、B、C類排序演算法時間複雜度如下:

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Average | Worst-case | Best-case |
| Selection Sort | O (n2) | O (n2) | O (n2) |
| Merge Sort | O (n log (n)) | O (n log n) | O (n log (n)) |
| Radix Sort | O (kn) | O (kn) | O (kn) |

電腦的執行環境如下:

CPU處理器: R7 3700X 8核心16執行續 3.6GHz - 4.4GHz

RAM記憶體: DDR4 3600 8G x 4條 = 32G

SSD固態硬碟: KC3000 1TB讀7000M/寫6000M

GPU顯示卡: RX6800XT 16G

OS系統: Windows 11 專業版 22H2版本

Compiler編譯器: Code::Blocks 20.03

一張含有 文字, 收據, 字型, 代數 的圖片

自動產生的描述一張含有 文字, 收據, 螢幕擷取畫面, 字型 的圖片

自動產生的描述

實驗結果如下表:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 資料量 | Insertion | Merge | Radix | C qsort | C++ sort |
| 100 | 0.001 | 0.0002 | 0.0002 | 0.0001 | 0.0001 |
| 500 | 0.0018 | 0.0002 | 0.0001 | 0.0001 | 0.0002 |
| 1000 | 0.0039 | 0.0003 | 0.0002 | 0.0003 | 0.0002 |
| 5000 | 0.0368 | 0.0013 | 0.0012 | 0.0011 | 0.0016 |
| 10000 | 0.1196 | 0.0027 | 0.0026 | 0.0021 | 0.0029 |
| 50000 | 2.4684 | 0.0148 | 0.0139 | 0.0125 | 0.0151 |
| 100000 | 9.6324 | 0.0310 | 0.0267 | 0.0256 | 0.0295 |
| 500000 | 179.634 | 0.1489 | 0.1415 | 0.1297 | 0.1457 |
| 1000000 | 730.169 | 0.2912 | 0.2945 | 0.2663 | 0.2982 |
| 5000000 | TLE | 1.6520 | 1.6058 | 1.4359 | 1.5692 |

註：時間以秒為單位，「TLE」(Time Limit Exceeds，超過時間限制)均是 > 20min

**結論**

從實驗結果來看，可以發現Merge Sort、Raidx Sort、C qsort與C++ sort所花的時間是差不多的，而Insertion Sort相對其他四個慢了許多，當資料量越大，差距越明顯。