資料結構Homework1報告

* 摘要
* 程式規劃
* 實際設計
* 遇到困難
* 老師給PPT的評語
* 結語
* [Github連結](https://github.com/sheng411/School/tree/master/C%2B%2B/data_structure_course/HW1)

摘要

透過insertion sort(插入排序),selection sort(選擇排序),bubble sort(泡沫排序)及merge sort(合併排序)來對各資料進行排序並分析其花費時間。

程式規劃

一開始的想法就是先把四個排序法寫在最上面，透過亂數生成產生所需的所有參數，並在程式最上方設定資料量(fre)及重複執行次數(smfre)這樣一來再跑資料時即可用更快速的方式來更改參數。在下方我透過switch來分類個排序法，當在終端機輸入1時可產生插入排序法之所有執行後數據，並自動輸入在count.txt裡面以便觀察及記錄。

實際設計

一開始我先新建一名為001.cpp的程式，把需要調整的參數(fre(資料數量) and smfre(執行次數))放在最上面，並開始撰寫三個sort(不含merge)，再來寫出名為out 的function來做資料的測試，最後寫出gogo function來把資料輸出到count.txt(方便使用者閱讀用)。Merge sort則是以檔名為mergesort.cpp來做執行(概念和001.cpp相似)。

遇到困難

1. C Language無法放入較大的空間🡪使用new來解決其問題。
2. Std::array無法放入較大的空間🡪透過static解決無法擴充空間的問題。
3. Merge sort無法再001程式裡執行🡪新增一個程式檔來執行merge sort。
4. 計時器在function會無法計時(溢位或偵測不到數值)🡪將程式改在main裡執行。
5. Vector會因為跑的次數增加而佔用過多記憶體🡪透過clear()來清除以排序過的資料來騰出空間。

老師給PPT的評語

1. 可以把每個圖表的y軸設定為相同的max值(以利閱讀者一眼就看出三種陣列的差異)。
2. 新建一個以陣列型態為主的圖表，可以分析出該陣列在每個資料型態所跑出的值為何。

結語

首先很感謝王三元老師、黃韋源助教及蘇宥彤同學從中引導及協助我完成本次作業，在過程中我有許多未接觸過的語法都是透過老師、助教及同學的指點才能順利完成，這也讓我了解到我對於C++這個語言還有許多需要專研的細節及用法。

這次的作業我不僅對四種排序法有更深的印象，在C Language array , std::vector and std::array上都有更清楚的了解及應用，並學習到一些平常不見得會使用到的語法。

2022/12/07 林祐陞撰