

作業五貼文

(一)簡介:

任務一：外部排序 external sort

依照【量化權重】由大到小排序

首先，先確認是否有先前建立的資料庫，如果有便刪掉，先尋找同名的二進位檔，如果不存在同名二進位檔，會顯示錯誤訊息並請使用者再次輸入，成功找到同名二進位檔後，以二進位方式照老師要求以 200 筆為限制的緩衝區讀檔並每 200 筆邊做內部排序（氣泡排序），同時紀錄內部排序及讀檔時間，接著做外部排序（合併排序）及寫檔，並同時紀錄外部排序及寫檔時間。

內部排序（氣泡排序）：

先取得檔案總數量數除以 BUFFER_SIZE(200)確認是否能整除 200，是否會有多餘的（不到 200 筆的），接著以緩衝區一次讀取 200 筆資料將此資料做內部排序（氣泡排序）後推進資料庫，直到所有資料皆讀完（如果檔案總數量數無法整除 200，剩下不到 200 的會自成一檔，同樣做內部排序後推進資料庫）

外部排序（合併排序）：

外部排序=>每次皆推進一個緩衝區的數量成兩個檔案再丟去做合併排序

以緩衝區為 200 筆的限制每每去做合併排序，每次合併排序如果比到後面有一邊先放完了（例如剩 50 筆），那剩下的會移到下一次（要比之緩衝資料庫前 50 筆是上一回還未比完的），合併排序的過程中每次皆會紀錄上一次剩下的，直到兩個檔案其中一個檔案全部比完(另一檔案如有剩下的會直接按照原始順序複製到後面)，最後兩者合併成新檔案

任務二：建立主索引 primary index

首先要先判斷是否執行過任務一否則沒有 sorted 二進位檔去進行任務二，接著一樣去判斷是否有先前建立的資料庫，如果有便刪掉，以 200 筆緩衝區為限制讀進 sorted 二進位檔以建立主索引，最後依序顯示主索引的每筆紀錄(包括量化權重和檔案位址)

建立主索引 primary index:

首先，取得檔案大小並計算出總數量整除 200 的量（num 為有多少個 200 筆資料），宣告暫放 200 筆資料的緩衝區和 divide（紀錄 1 是分資料到新葉 2 是分節點）以一個迴圈去跑去建立資料庫。

每一次讀 200 筆後放資料至主索引，會先一路抵達葉節點，若葉節點還有空間則直接放入，否則會回到上一個節點，判斷 divide 是 1 還是 2，如果是 1 則分資料到新葉；如果是 2 則分節點，直到跑完 num，判斷如果有剩餘的資料（不到 200 筆的）做一樣的事，先一路抵達葉節點，有空間則放入，無則回到上一個節點，判斷 divide 是 1 還是 2，如果是 1 則分資料到新葉；如果是 2 則分節點。

每次放資料，如果判斷樹葉 3 個都滿了則回去上一個節點分新支

Divide 為 1: 要分裂的地方為樹枝，樹枝數小於 6

Divide 為 2: 要分裂的地方為節點，樹枝數等於 6

任務三：範圍搜尋 range search

首先要先判斷是否執行過任務一和任務二，接著判斷是否有先前建立的資料庫，如果有則啟用資料庫，接著請使用者輸入門檻。

目的是用任務二的主索引去搜尋量化權重符合使用者輸入門檻該範圍內的所有資料，首先用任務二建好的 **B+ tree**，沿著搜尋用節點逐步往下尋找到存放指定量化權重的資料庫，取得其位移數，作為終止的數量。

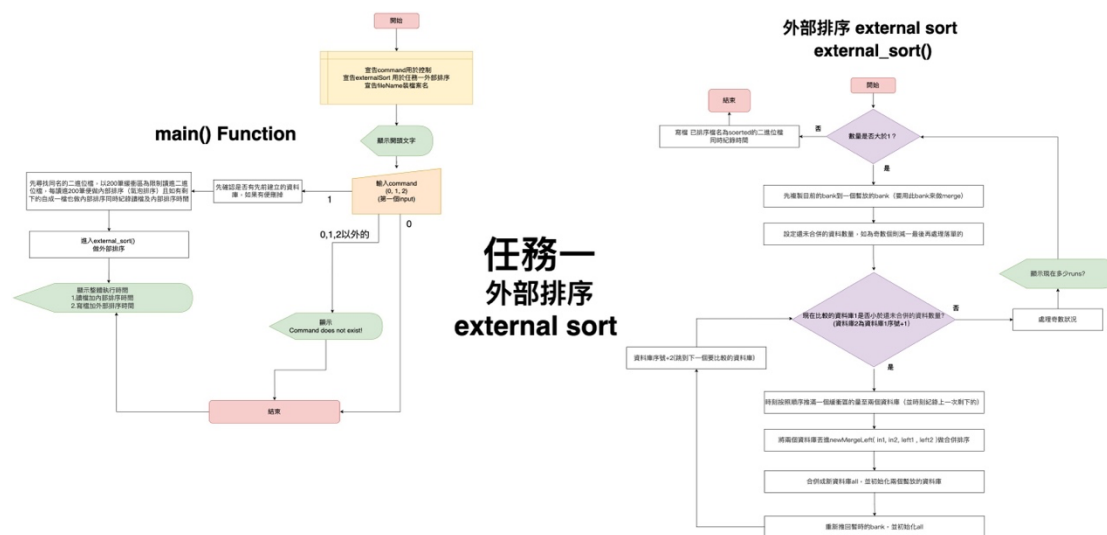
最後打開任務一建立好的已排序檔案，以每次最多讀入 200 筆的方式，印出答案到螢幕上，直到印出資料數量達終止數。

心得:

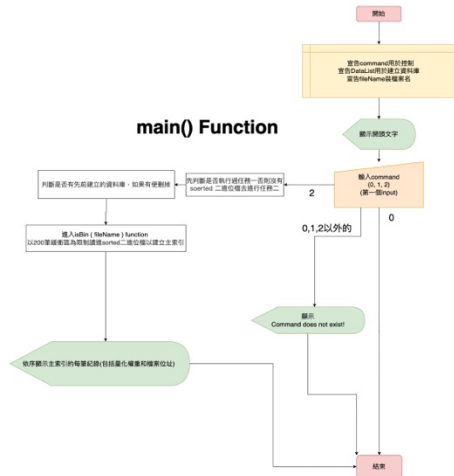
這次作業五最困難的地方在任務一，光是理解題目就花了一點時間且怕誤會題目被扣分因此想要先確認好題目正確的讀檔方式和外部排序的定義，在任務一就花了將近一個禮拜的時間搞定，到後面任務二的部分，我們是用 **b+tree**，由於 **b tree** 類似於之前寫過的 234 樹因此很快便能理解原理並寫出來，不過雖然仍然花了一點時間建立 **b+tree**，在任務三範圍搜尋時便能很方便印出量化權重符合使用者輸入門檻該範圍內的所有資料，這次的作業真的是學到非常多，了解了更多的讀檔方式，知道了內部排序外部排序的分別定義，也更了解了如何建立 **b tree** 和以此作範圍的搜尋。

(二)圖示:

任務一:



任務二:

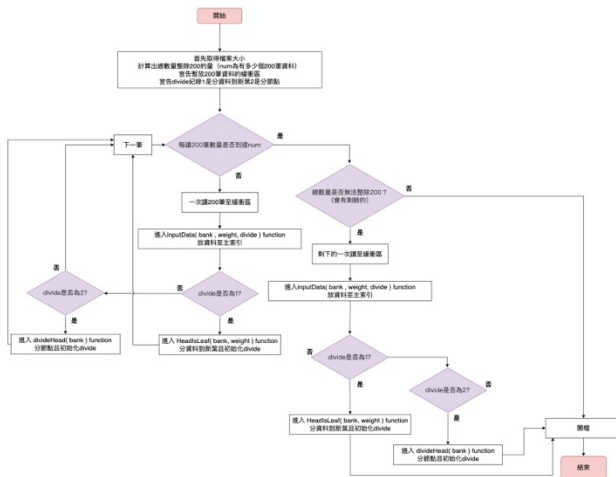


處理二進位檔 isBin (string input)



任務二 建立主索引 primary index

建立主索引 primary index checkBinfileline(string input)



放資料至主索引 inputData(Data2* temp , float now, int& divide)

