

資料結構-作業 1 報告

- 使用環境:

使用 Windows10 作業系統，並且使用 `code-blocks` 編譯器撰寫程式碼

- 遇到問題與困境，與解決辦法:

- ◆ **PROBLEM-1**

- 1.遇到困境(想直接設定二維整數矩陣大小並傳入):

- 第一題原本想要使用整數二維陣列來儲存輸入資料，後來發現資料可能會產生超過儲存空間的情形要報錯，因此改用另一種方式，就是先將輸入資料存入字元陣列

- 2.解決辦法:

- 判斷空格情形來決定輸入資料是否可以成為方陣來計算轉置矩陣，如果不行則回傳報錯輸出，再來將正確的字元陣列，一一轉成整數方陣二維陣列儲存，再來就是遍歷所有非零數值，並使用一個計數器來計數。接著將所有非零數字位置的第?列第?行輸出並輸出該數值，最後將其轉置後的座標位置與原先對調，並且根據由第一列輸出完後接續第二列的輸出規定完成後續動作，並得出與測資相同的答案。

- 程式內容:

- 1 首先宣告變數與二維陣列(含字元陣列與整數陣列各一)，將輸入讀取成字元陣列並使用迴圈讀取每一個字元，把字元陣列中的值一一轉換為整數型態的值並存入已經設定好大小的整數方陣(二維陣列)，中間利用判斷式辨別多位數與空格還有換行處理

- 2 再來判斷遇到空格或是換行代表前面有一項，計算矩陣中的項目是否可以成為一個方陣，如果計數器不符方陣所需項目個數，回傳 **Input matrix has wrong size. Please input again.**如果符合則計算非零項的個數，並在輸出呈現矩陣大小與非零項數目。

- 3 最後則是處理輸出部分，遍歷整數矩陣後把非零項的列與行數字紀錄下來，並輸出數值，轉置矩陣則是行列互換之後由一樣紀錄列與行數並輸出數值。

- 遇到問題與困境，與解決辦法:

◆ PROBLEM-2

1. 遇到困境(由後序輸出數值需要考慮四捨五入到整數並且考慮 $3/2=1.5$ 會進位成 2 的情形):
根據上述情形，我原本將 `stack` 函式中的變數宣告與 `pop, push` 回傳變數型態都是使用 `int`，所以面對以上情形，需要考慮先存成浮點數型態方便後續操作。
2. 解決辦法:
使用 `float` 來取代原本 `int` 的型態，讓中間運算子運算元作四則運算還有最後回傳 `result` 的數值都能用 `float` 格式回傳函式值，接著我們使用 `roundf` 函式來判斷四捨五入，這邊先將 `value` 設定成函式回傳的 `float result`，再來使用 `roundf` 將 `value` 四捨五入成為 `float fvalue`，最後才訂定整數 `int ac` 把 `fvalue` 轉成整數型態儲存，並輸出。
*(部分程式碼)

```
value = eval_post(result);  
float fvalue = roundf(value);  
int ac = (int)fvalue;  
printf("> %d",ac);
```

- 程式內容:

- 1 首先宣告變數與回傳用函數還有堆疊，並且將輸入的 `infix` 轉 `postfix` 函數寫出並判斷運算子的先後順位來存取進入堆疊
- 2 堆疊基礎的三個回傳函式:`push, pop, empty`
- 3 計算 `postfix` 轉輸出的數值，把剛剛存取的堆疊，判斷遇到的是運算元則直接 `push` 並 `pop`，剩餘的運算子則是根據四則運算的順序:括弧-乘除-加減來一一加入堆疊並 `pop` 出(根據 `isp, icp` 的順序判定)最後加入剛才的運算元做運算得出數值回傳
- 4 主函式則是將前面函式寫的回傳值，配合規定格式做輸出