

一、規格要求，違反者以零分計！

- (1) 以 Dev-C++ 或 Code::Blocks 編譯與成功執行的 C/C++ 程式碼，必須有充分的註解。
- (2) 任何一部分程式碼都不得被偵測為抄襲，程式碼的第一列要註解學號和姓名。
- (3) 檔名限以「**DS2ex1_組別_學號_學號**」開頭，**兩人一組只限繳交一份**。

二、作業內容

整合下列任務於同一個程式及單一操作選單下，未整合、無法連續執行或沒有輸入防呆措施，都各扣 5 分。若導致任務無法正常運作，該任務以零分計。

資料檔格式簡述：

- (1) 取自教育部統計處的大學校系畢業生資訊，前三列為標題和欄名，第四列起為各校系某班制的畢業生資訊，共 **11 個欄位**，以定位符號（'\t'）間隔，依「學校代碼」遞增排序。
- (2) 欄位由左至右依序為：學校代碼、學校名稱、科系代碼、科系名稱、日間 / 進修別、等級別、學生數、教師數、上學年度畢業生數、縣市名稱、體系別，資料檔名如 **input101.txt**。
- (3) **按照原始的存檔次序為每筆資料附上從 1 號開始的唯一『序號』。**

規範：每個任務違反一項各扣 5 分

- (1) 預先不知道資料筆數，禁止使用固定筆數的靜態陣列，必須採用**動態陣列**或 vector。
- (2) 必須為主要資料結構定義**專屬的 C++ 類別**，至少要有最小堆積和最小最大堆積類別。

（任務一）建立最小堆積 min heap

輸入：原始資料檔，資料檔名如 input101.txt。

步驟：

- (1) 每筆資料附上**唯一序號**後，以『**上學年度畢業生數**』建立一棵最小堆積，依照序號由小到大**一筆一筆新增**至原有的堆積結構中，每個節點只存放一筆資料對應的（序號、**上學年度畢業生數**）。
 - (2) 找到最小堆積的樹根，**底部節點**（bottom），以及最**左下角落**的節點。
- 輸出：分別顯示前述三個節點對應的（序號、上學年度畢業生數）於螢幕上。

（任務二）建立最小最大堆積 min-max heap

輸入：原始資料檔，資料檔名如 input101.txt。

步驟：

- (1) 以『**上學年度畢業生數**』再建立一棵最小最大堆積，依照任務一所附加的序號由小到大**逐筆新增**至原有的堆積結構，每個節點存放一筆資料對應的（序號、上學年度畢業生數）。
- (2) 找到最小最大堆積的樹根，**底部節點**（bottom），以及最**左下角落**的節點。

輸出：分別顯示前述三個節點對應的（序號、學生數）於螢幕上。

程式碼：上機練習前一天上傳原始碼至/**評量區/作業/作業 #題次 上傳**/，隔天可補繳。

流程圖：上機練習前先畫好初稿，加入貼文時修正，超過期限不得更新，也不接受補繳。

三、評分項目

- (1) 作業程式碼：任務一、二程式碼各佔 20 分，一個錯誤扣 5 分，兩個或更多以零分計。
- (2) 作業流程圖：一項任務的流程圖佔 10 分，一個錯誤扣 5 分，兩個或更多以零分計。
- (3) 挑戰程式碼：前兩項任務都完全正確才計分，任務三的程式碼佔 20 分，一個錯誤扣 5 分，兩個或更多以零分計。
- (4) 文件報告：簡介佔 5 分，一個錯誤(字)以零分計。解說佔 10 分，一個缺陷扣 5 分，兩個或更多以零分計。
- (5) 機測報告：每位組員單獨受測，成績分開計算，在時限內正確回答 2-4 個提問，一個錯誤扣 5 分，共佔 15 分。

四、評分流程

- (1) 上機練習兩週前公布作業的題目，上機練習當天才公布上機挑戰的題目。
- (2) 每項任務可能會以非公開測資評分，程式碼的耐用性要自行檢測，並力求完美！
- (3) 機測時段在機測前一天公布，至少有一項任務上傳程式碼且貼文完整，才列入機測！

五、偵測抄襲

- (1) 嚴禁抄襲網路上或相關課程的舊程式碼，老師提供或重修生自己以前寫的程式碼除外。
- (2) 一旦偵測程式、助教、和老師均認定抄襲，即使是一小部分的程式碼，一律以零分計。

六、參考範例，格式自訂，內容不可欠缺！

(任務一) 建立最小堆積

Input the file number ([0]: Quit): 101

<min heap>

root: [10] 37

bottom: [12] 86

leftmost bottom: [4] 187

(任務二) 建立最小最大堆積 min-max heap

Input the file number ([0]: Quit): 101

<min-max heap>

root: [10] 37

bottom: [26] 77

leftmost bottom: [16] 49