

// 只上傳可成功編譯的原始碼(.cpp/.c/.h/.hpp)含註解、檔名請用「DS1ex5_分組編號_學號」，欠缺任何一項各扣 5 分！

// 程式碼前三行必須要有註解附上該組每位同學的中文姓名和學號，資訊不完整先扣 5 分！

// 每組只需要一位組員上傳程式碼和貼文，務必要在標題加上分組編號，兩份結果或標題不正確各扣 5 分！

// 非 C/C++ 程式 或 無法成功執行 一律視為「未完成」並以零分計！

一、題目

完成下列兩項任務，將二者整合在一個簡易選單下，未整合或介面無法連續執行先扣 5 分。

前言：

- 佇列 Queue 可以應用在許多層面，諸如工作排程，飛機起降，點餐系統等，以印表機列印多個檔案為例，先送到的檔案先列印，其餘檔案則暫時在佇列中等候。數據機傳輸也是，網路堵塞時先暫存資料在佇列中，等網路暢通時再繼續傳送。
- 假設 CPU 排程處理的工作 (job/process) 表示成四個欄位：「工作編號」JobID、「進入時刻」Arrival (第幾毫秒)、「工時長度」Duration (多少毫秒) 和「逾時時刻」TimeOut (第幾毫秒)， $Duration > 0$ 且 $Arrival + Duration \leq TimeOut$ 。
- 原始資料檔是文字檔，第一列由左至右依序是四個欄位名稱，以定位符號 ('t') 間隔，其餘的每一列各代表一筆工作，四個欄位的數值均為正整數，也以定位符號 ('t') 間隔，預設是沒有排序過的，檔名如 input501.txt、input502.txt。

(任務一) 排序後另存新檔

輸入：原始資料檔。

描述：自行撰寫希爾排序(shell sort)法，依「進入時刻」遞增排序，多筆相同進入時刻者則另以「工作編號」遞增排序，排序後另存新檔，並測量讀檔，排序和寫檔的執行時間。

輸出：儲存排序後的新檔 (檔名如 sort501.txt、sort502.txt)，並顯示三個執行時間於螢幕上。

佇列模擬原則：(違反一項扣 10 分)

1. 每一筆工作一旦開始執行後都不能被中斷 (not preemptive)。
2. 進出佇列一律採用先進先出 (FIFO) 策略。
3. 佇列不得使用內建結構，必須自行以環狀陣列 (circular array) 實作成 C++ Class，而且空間上限只能存放最多 5 筆工作。
4. 每筆工作只能因佇列空間不足而立即捨棄，並累計於『捨棄工作清單』：欄位包括「工作編號」、「捨棄時刻」Abort (第幾毫秒) 和「延遲長度」Delay (多少毫秒)。
 - 在上述情況下，「捨棄時刻」即為該筆工作的「進入時刻」，「延遲長度」則均為 0。
5. 自佇列取出的工作，若發現一定會逾時，同樣也予以捨棄，並累計於『捨棄工作清單』。
 - 在上述情況下，「捨棄時刻」為自佇列取出該工作的時刻，「延遲長度」則為「捨棄時刻」減去該筆工作的「進入時刻」。

6. 執行中舊工作的「完成時刻」和新工作的「進入時刻」相同時，一律先移除舊工作，以避免新工作因空間上限而無法進入。

（任務二）單一佇列模擬

輸入：讀入任務一排序後的工作資料檔，存放於一個動態陣列中。

描述：(1) 從第一筆開始模擬工作進入單一 CPU 排程的等候狀態（waiting state）及執行狀態（running state），無法立即執行的工作先放入佇列。

(2) 執行一筆工作時，先檢查是否會逾時，若會逾時就立即捨棄，計算其延遲長度後並累計於『捨棄工作清單』。

(3) 可成功執行的工作，則依序累計於『執行工作清單』：欄位包括「工作編號」、「完成時刻」Departure（第幾毫秒）和「延遲長度」Delay。

(4) 模擬完成後，統計『平均延遲長度』（捨棄工作也要算在內）及『成功執行比例』（成功執行工作筆數佔所有工作的百分比），一律取四捨五入至小數點後兩位的數值。

- 在上述情況下，『平均延遲長度』為每筆工作的「延遲長度」總和除以工作總筆數，『成功執行比例』為『執行工作清單』的工作筆數除以工作總筆數。

輸出：依序將『捨棄工作清單』、『執行工作清單』、『平均延遲長度』及『成功執行比例』寫成一個文字檔（檔名如 output501.txt、output502.txt）。

二、參考範例

（任務一）排序後另存新檔

Input a file number (e.g., 501, 502, 503, ...): 501

JobID	Arrival	Duration	TimeOut	//依序為「工作編號」、「進入時刻」、「工時長度」、「逾時時刻」
103	6	7	15	
104	11	9	22	
112	5	6	13	
101	3	9	12	
106	6	9	17	
108	6	8	18	
105	10	6	20	

Reading data: 2 ms

Sorting data: 0 ms

Writing data: 3 ms

JobID	Arrival	Duration	TimeOut	// saved as sort501.txt
101	3	9	12	

112	5	6	13
103	6	7	15
106	6	9	17
108	6	8	18
105	10	6	20
104	11	9	22

（任務二）單一佇列模擬

[捨棄工作清單]

// saved as output501.txt

	JobID	Abort	Delay	
				// 由左而右依序為「工作編號」、「捨棄時刻」、「延遲長度」
[1]	104	11	0	// 佇列已放滿五筆工作，故立即被捨棄，延遲長度為 0
[2]	112	12	7	// 佇列的第一筆工作，發現會逾時，故捨棄之。(勿提早檢查)
[3]	103	12	6	// 佇列的後續三筆工作，發現也會逾時，所以依序捨棄之
[4]	106	12	6	
[5]	108	12	6	

[執行工作清單]

// 由左而右依序為「工作編號」、「完成時刻」、「延遲長度」

	JobID	Departure	Delay	
[1]	101	12	0	
[2]	105	18	2	// 佇列的最後一筆工作並不會逾時，從 10 等到 12 才執行

[平均延遲長度] // $(0+7+6+6+6+0+2) / 7$

3.86 ms

[成功執行比例] // $100 * 2 / 7 \%$

28.57 %

// 注意：程式跑不出正確結果，或未依規定格式輸出，均視同「未完成」，並以零分計！

三、預交（分數不打折）的必要條件

步驟 1. 同組兩人均有簽到。

步驟 2. 期限前完成一項任務，成功上傳程式碼後找助教或「已完成同學」展示正確結果。

步驟 3. 助教在「上機評分表」上勾選已完成預交。

// 注意：兩項任務在上機練習時完成者，可預約提前機測，機測前都要先在討論版貼文！

四、程式簡介、流程圖及答問

截止日期前必須在本次上機練習的討論版張貼這一篇文章，否則成績歸零。

1. 簡介：以文字簡述程式主旨，假設，遇到的困難和解法，勿直接剪貼題目字句！
2. 流程圖：每項任務各一張流程圖，以附圖置於貼文之後！
3. 答問：若直接輸入尚未排序的原始資料檔，任務二的程式碼要做哪些調整？從執行效率和撰寫程式的觀點探討任務一的必要性。

五、機測程序及分數配置

步驟 1. 已完成提前機測、未上傳程式碼或未貼文者，均不列入機測名單。

步驟 2. 遵循公告名單的指定助教和機測時段到機房，遲到或缺席者視同放棄，一律零分。

步驟 3. 只限下載所上傳的程式碼，重新編譯後執行。經助教同意，只能用自己筆電機測者，一律先扣 10 分。

步驟 4. 機測評分後，助教將根據是否完成預交予以打折。

步驟 5. 兩人一組時，一人負責機測一項任務，若只有一人機測，將會少一項任務的成績。

項目 1. (任務一) 40%

項目 2. (任務二) 40%

項目 3. (1) 程式碼和註解易讀性、執行介面友善度 10%

(2) 程式簡介、流程圖及答問 10%

六、機測的評分方式

前兩個項目在機測現場評分，項目 3.則在機測之後由助教自行檢視

(階段一：實作) 隨機施測 1-3 個不同輸入

1. 答案完全正確 得 30 分
2. 輸出結果只出現 1 筆錯誤的答案，依助教指示及時修正 得 25 分
3. 輸出結果出現多於 1 筆錯誤的答案，依助教指示及時修正 得 20 分
4. 未能依助教指示及時修正，但是助教認定已大部份完成 得 10 分
5. 其他 得 0 分

(階段二：原理) 抽問程式相關的 1-3 個問題

1. 回答正確且能清楚解說程式碼 得到 10 分
2. 無法正確回答 1 個問題或無法清楚解說 1 行程式碼 得 5 分
3. 無法正確回答超過 1 個問題或無法清楚解說 1 行以上的程式碼 得 0 分

// 注意：成績公佈後才開始以軟體及人工比對程式碼相似度，由老師做最後裁定，相似度高於門檻的雙方都一律零分。