

Homework 2 個人單月行事曆系統

Deadline: 2018-11-02 23:59:59

前言

小明是一個剛考上國立成功大學電機系的大學生，他發現大學生活和高中時期完全不一樣，除了修習課業以外，還要參與各種社團活動，因此時間的安排變得十分重要。本作業希望同學以 Java 程式語言幫忙小明建立一個單月行事曆系統，透過新增、刪除及瀏覽行程等功能，讓他可以有效地管理自己的時間。

作業需求

■ Functional Requirements

- 此行事曆系統有月行事曆和單日行事曆兩個使用模式，且兩個模式可以自由切換。
- 在月行事曆模式，使用者可以瀏覽 2018 年 10 月之月曆，有活動地日期會以 * 符號標記（第一次執行程式時，無預設存在任何活動，因此不會有 * 標記），並要求使用者輸入指令進行後續操作（詳見指令表說明）。月行事曆範例如下：

2018 年 10 月						
Sun.	Mon.	Tue.	Wed.	Thu.	Fri.	Sat.
	1	2	3*	4	5*	6*
7	8	9	10	11	12	13
14*	15	16	17	18*	19	20
21*	22	23	24	25*	26	27
28	29*	30*	31*			

於月行事曆模式下，使用者可進行以下兩種操作：

- A. 查詢該月任一日之所有活動（即切換到單日行事曆模式）
- B. 結束系統：系統顯示結束訊息並結束系統

- 在單日行事曆模式下，使用者可以瀏覽當日的**所有活動和活動時間**（本系統一律採用 24 小時制，即一日時間為 00:00~23:59，無跨日活動，且第一次執行程式時，無預設存在任何活動），並可透過指令（詳見指令表說明）進行新增、刪除、修改等操作。單日行事曆範例如下：

2018 年 10 月 10 日

活動：

1. 10:00~12:00 看國慶表演
2. 12:30~14:00 同學會聚餐
3. 20:00~22:00 撰寫計概 Java 程式作業

於單日行事曆模式下，使用者可進行以下四種操作：

- A. 切換至月行事曆模式
- B. **新增該日活動**：每新增一筆活動，系統會檢查是否與當日已存在活動發生衝突，若有衝突發生，系統會提示錯誤訊息並要求使用者重新輸入。除此之外，系統會將新增的活動按照時間順序由早到晚進行排序，並賦予每個活動由 1 開始的編號，最後會呈現新的單日行事曆。以上述例子來說，當使用者新增「15:00~18:00 社團練習」的活動後，系統會呈現如下，並要求使用者繼續輸入指令。

2018 年 10 月 10 日

活動：

1. 10:00~12:00 看國慶表演
2. 12:30~14:00 同學會聚餐
3. 15:00~18:00 社團練習
4. 20:00~22:00 撰寫計概 Java 程式作業

- C. **刪除該日活動**：每刪除一筆活動，系統會將下一個活動的編號向前遞補，並呈現新的單日行事曆。以原先的例子來說，當使用者刪除編號為 1 的「看國慶表演」活動時，系統會呈現如下，並要求使用者繼續輸入指令。

2018 年 10 月 10 日

活動：

1. 12:30~14:00 同學會聚餐
2. 20:00~22:00 撰寫計概 Java 程式作業

- D. **結束系統**：系統顯示結束訊息並結束系統

- 月行事曆和單日行事曆之指令表

模式	指令格式	範例	說明
月行事曆模式	day-[日期]	day-10/03	切換至 10 月 3 日之單日行事曆模式
	exit	exit	結束系統
單日行事曆模式	add-[活動時間]-[活動名稱]	add-15:00~18:00-社團練習	新增活動
	del-[活動編號]	del-2	刪除該日編號為 2 的活動
	month	month	切換至月行事曆模式
	exit	exit	結束系統
	del-all	del-all	刪除該日所有活動 (額外加分)
	edit-[活動編號]-[活動時間]	edit-1-9:00~10:30	修改編號為 1 之活動的活動時間 (額外加分)

■ Non-functional Requirement

- 提供完善、人性化、易閱讀的使用者介面
- 當使用者輸入指令後，需有相對應的回應，提供良好的使用者體驗
- 由於作業評分需要，**不可**擅自增加額外的停頓點或延遲效果
- 系統中使用的指令請確實依照**指令表**的格式進行實作。當指令中有包含時間（日、小時、分）的部分，若出現個位數字時請補上零，範例如下：

錯誤指令	正確指令
day-10/3	day-10/03
add-07:06~9:6-sleep	add-07:06~09:06-sleep

■ Execution Flow (Use Case)

1. 執行程式後，系統顯示歡迎訊息，並直接進入月行事曆模式
2. 在**月行事曆模式**下，系統顯示 2018 年 10 月的月曆（請參閱 Functional Requirements 中說明），並要求使用者輸入指令，若使用者輸入指令切換到單日行事曆模式則進入步驟 3
3. 在**單日行事曆模式**下，系統顯示當日所有活動（請參閱 Functional Requirements 中說明），並要求使用者輸入指令，若使用者輸入指令切換到月行事曆模式則回到步驟 2

額外加分項目

- 在不破壞執行流程的限制下，可為此系統加入新功能或新特色以獲得額外加分，**但需在 readme.txt 中明確說明**，包含如何觸發或使用等等。
- User Interface 排列整齊且精美者，將能獲得額外加分。
- 系統可以判斷指令錯誤的類型，例如大小寫錯誤、指令不完全、指令時間與現實不符等等，**但須在 readme.txt 中明確說明**

作業繳交方式

請參考計概網站上的「**作業繳交說明與規範**」將作業相關檔案上傳到資料夾「hw2」當中。作業相關檔案分為以下四部分，括號中的數字為該部分的配分比：

1. **程式原始碼檔案 (70%)**：以 JAVA 根據作業要求撰寫而成的檔案，主程式檔名為「**hw2.java**」，主程式需撰寫**註解**幫助別人看懂你的程式碼，若同學的程式執行時需要用到其他檔案，如「ConsoleIn.java」，也需作為程式檔案的一部分一起繳交上來。
2. **文件檔案(27%)**：檔名為「document.doc」、「document.docx」或「document.pdf」，文件需包含以下內容：
 - (1) **需求描述**：描述使用者在使用此程式時可能會有何需求？你設計程式時如何考慮這些需求？程式中有哪些地方特別吸引使用者？
 - (2) **程式流程**：說明程式進行流程，建議使用利用流程圖或是其他有助說明的圖示來幫助說明程式整體上如何運作

- (3) 功能/邏輯說明：說明自己的程式裡，自己針對什麼樣的「功能」特地設計了什麼樣特別的「邏輯」來達成？建議用畫圖的方式來說明
- (4) 使用說明：以教導使用者的角度，說明程式要如何執行使用或是需特別注意的點...等等。(不可只寫「看了就會使用」之類的話。)
- (5) 其他：任何有助於別人了解或使用你的程式的說明。切勿流於閒聊

3. **readme.txt (3%)**：內容請按照「作業繳交說明與規範」當中的要求撰寫

4. **TimeLog.doc**：記錄此次作業撰寫時間。(「確實」填寫該表的同學可獲加 5 分)

特別注意事項

- 1. 繳交作業請使用 SVN 上傳，若有相關問題請參考計概網站上的 [計概帳號註冊/作業上傳教學](#)，上傳後可以使用論壇上「**check homework submission**」的功能進行確認
- 2. 請務必確保檔案擺放及命名符合規定，若作業檔案放錯資料夾、檔名大小寫或拼寫有誤，將可能造成作業無法正確批改。因此，**出現此情形將酌情扣分**
- 3. **作業請勿抄襲，所有作業皆會經過程式比對判斷是否抄襲，若發現一律以零分計算**