

Homework 3 成大小吃店

Deadline: 2018-11-25 23:59:59

前言

曾先生是成大小吃店的老闆，雖然生意很好但卻人手有限，常常處於手忙腳亂的狀態，無法有效管理顧客頻繁的點餐需求和安排廚師製作餐點。因此本作業要求同學以 JAVA 程式語言實做一個簡易的點餐系統，幫助老闆可以透過輸入簡單指令為顧客點餐以及安排廚師們的工作分配，並且隨時可以了解餐點準備狀況。

作業需求

■ Functional Requirements

- 成大小吃店一共有四個座位，每個座位一次只能坐一個人
- 成大小吃店內共有兩位廚師，兩位廚師分別擅長不同的餐點，廚師和餐點的介紹分別如下：

餐點編號	餐點名稱	正常準備時間	是否可一次準備多份	食用時間
M1	炒飯	6 分鐘	是	10 分鐘
M2	鍋燒意麵	4 分鐘	否	8 分鐘
M3	炒青菜	2 分鐘	是	2 分鐘
M4	奶油千層蛋糕	8 分鐘	否	5 分鐘
M5	綜合果汁	2 分鐘	是	3 分鐘

廚師編號	擅長餐點
S1	炒飯、炒青菜
S2	奶油千層蛋糕、綜合果汁

廚師一次僅能負責一項餐點，但當廚師負責到其擅長的餐點時可以縮短一半的準備時間。
舉例來說，S1 廚師準備炒飯時僅需要 3 分鐘、S2 廚師準備炒青菜和奶油千層蛋糕共需要 6 分鐘。

- 每位客人在餐點上桌後會根據餐點食用時間用餐，並且吃完所有點選的餐點後就會離開，該客人所在座位即會釋出留給下一位客人使用。

例 1：小明點了炒飯和綜合果汁兩份餐點，曾老闆將炒飯分配給 S1 廚師、綜合果汁分配給 S2 廚師。由於綜合果汁在第 1 分鐘時已準備完成，小明即開始用餐，並且在第 4 分鐘時食用完畢。由於用餐期間炒飯也已準備完成，因此小明可以直接繼續食用炒飯餐點，最後在第 14 分鐘時食用完所有餐點並離開。

例 2：小美點了奶油千層蛋糕和綜合果汁，曾老闆將奶油千層蛋糕分配給 S1 廚師、綜合果汁分配給 S2 廚師。由於綜合果汁在第 1 分鐘時已準備完成，小美即開始用餐，並且在第 4 分鐘時食用完畢。奶油千層蛋糕在第 8 分鐘時才準備完成，因此小美必須要等到第 13 分鐘時才食用完所有餐點並離開。

- 本點餐模擬系統採用指令輸入模式，讓曾老闆可以有效管理小吃店的運作狀況和進行點餐

- **點餐模式**

點餐指令	指令格式	範例	說明
點選新餐點	add [餐點編號] [廚師編號]	add M2 S1	點選鍋燒意麵餐點並安排給 S1 廚師負責
點餐結束	end	end	結束點選餐點

點選新餐點後，該餐點會加入該廚師的餐點準備清單中，所有餐點的準備皆依照順序進行。若餐點準備清單中有餐點是可一次準備多份的（炒飯、炒青菜或綜合果汁），廚師可以一次準備清單中所有該餐點，而準備時間仍為一份的準備時間。

舉例來說，若廚師的餐點準備清單中的順序為「炒飯、炒青菜、炒飯、鍋燒意麵」，則廚師下一份餐點可以一次準備兩份炒飯，但準備時間仍為一份炒飯的準備時間。準備炒飯完後，廚師餐點準備清單的順序剩下為「炒青菜、鍋燒意麵」。

- **管理模式**

管理指令	指令格式	範例	說明
安排顧客座位	new [桌位編號]	new 1	安排客人至 1 號桌
查看店內狀況	status table	status table	查看店內餐桌狀況
查看廚師狀況	status chef	status chef	查看所有廚師準備狀況
模擬經過時間	next [時間（分）]	next 5	模擬過去 5 分鐘
結束系統	exit	exit	結束系統

安排顧客座位時，可以指定安排顧客的桌位，若該桌位已有人使用，系統會提示該桌位目前使用中。接著系統會切換到點餐模式，為顧客點選新餐點。

查看店內狀況時，可以查看目前店內所有桌位的狀況，系統會呈現哪些桌位有人、個別桌位上已上桌和剩餘未上桌的餐點有哪些。餐點僅需呈現目前在該座位上客人所點選的餐點，不須回顧之前在該座位上用餐過的其他客人。

查看廚師狀況時，可以查看目前店內廚師的狀況，系統會呈現個別廚師正在準備的餐點和剩餘要準備的餐點有哪些。

模擬經過時間時，系統會模擬時間過去多久，並更新目前餐桌的資訊（是否仍有客人在用餐、已上桌和未上桌的餐點等等）和廚師的資訊（正在準備和剩餘要準備的餐點）。

模擬經過時間僅需假設該時間過去，並呈現各資訊的更新狀況，**並不需要實際以計時器等方式運作**。

■ 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
- **本次作業要求除 hw3.java 外，至少需額外包含兩個以上的 class，須具備實際設計意義並於文件中說明。**

■ Execution Flow (Use Case)

1. 執行程式後，系統顯示歡迎訊息，並直接進入**管理模式**
2. 在**管理模式**下，使用者可以輸入管理指令並執行相對應的行為，並重複執行步驟 2，直到使用者輸入**安排顧客座位**的指令，系統會切換到點餐模式
3. 在**點餐模式**下，使用者可以輸入點餐指令並執行相對應的行為，並重複執行步驟 3，直到使用者輸入**點餐結束**指令，系統會切換到管理模式，並回到步驟 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。

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「hw3」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「**hw3.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**