**C＋＋課程設計指導書**

**一、設計目的**

通過設計、實現一個微型學生養成系統，熟悉對象導向的設計思想；進一步熟悉、掌握STL中有關容器、算法的使用，以及泛型編程的風格。

**二、設計目標**

**微型學生養成系統**

在這個課程設計中，你需要設計幷實現如下場景：在那山的那邊，湖的那邊，有一所學校，學校裏有一幢宿舍樓，宿舍樓有若干層，每層有若干房間 ，一群學生快樂地生活在這裏。他們每天可做的事情有：

1. 換宿舍，從一個房間搬到另一個房間；
2. 退學，亦即搬出宿舍樓；
3. 入學，亦即搬入宿舍樓；
4. 吃飯，吃飯會增加體重，花費金錢；
5. 學習，學習會降低體重，增加知識；
6. 打工，打工會增加金錢，降低體重；
7. 化妝（女生獨有的活動），化妝會花費金錢，提升容貌；
8. 運動（男生獨有的活動），運動會降低體重，增加健康。

上述4~8項活動可改變學生的魅力值，具體改變公式見後面的描述。

學生還可進行戀愛活動，任一學生可以向另外一個學生（爲了與時俱進，不要求二者必須為異性）提出戀愛申請，但只有雙方的魅力值的相對差不大於某閾值（該閾值在配置文件中指定）方可建立戀人關係。假設學生A追求學生B，魅力值相對差的計算公式為：

Delta = |A.魅力值 - B.魅力值| / MAX(A.魅力值, B.魅力值)

世事無常，戀人之間也可能分手。往事如夢，每個學生都要記住自己的戀愛歷史，亦即所有戀人（包括當前的戀人）列表。八榮八恥，不得有吃著碗裡，望著鍋裡之行為。

此外，宿舍樓可進行擴建，亦即增加樓層，增加某層的房間數，以及增加某個房間可容納的人數（不考慮這種任意增加的物理可能性）。經過一段時間的生活後可對學生的狀態信息進行查詢。

**具體要求：**

1. 程序運行的初始參數從某個指定的配置文件中讀取（該文件名作爲第一個參數傳遞給程序）。配置文件的格式詳見範例文件config.txt。
2. 學生的初始房間分配情况從某個指定的文件中讀取（該文件名作爲第二個參數傳遞給程序）。該文件每一行都是一個學生的初始房間分配，參見範例文件 dispatch.txt，格式如下：

【學生姓名】 【性別】 【房間號】

1. 所有的指令描述都存儲在某個指定文件中（該文件名作爲第三個參數傳遞給程序）。該文件每一行都是一個指令的描述，參見範例文件 instruction.txt,指令種類及格式說明如下：

【學生姓名】 MOVE FROM 【原房間號】 TO【新房間號】

解釋：換宿舍。

【學生姓名】QUIT

解釋：學生退學，搬離原房間。若其之後又重新入學，仍然保持退學時的狀態。

【學生姓名】ENROL 【房間號】

解釋：學生入學，搬入房間。

ADD 【層數】FLOOR(S)

解釋：在現有樓層之上增加【層數】個樓層，層號在現有樓層基礎之上順排；每層樓的房間數及每個房間的容量都是初始值。

ADD 【房間數】ROOM(S) TO【層號】 FLOOR

解釋：在第【層號】層樓增加【房間數】個房間，房間號在該層現有房間基礎之上順排，每個房間的容量爲初始值。

ADD 【床位數】BED(S) TO【房間號】

解釋：在【房間號】房增加【床位數】個床。

【學生姓名】【EAT 食物數量】【STUDY 學習時間】【MAKEUP 化妝時間】【EXERCISE 運動時間】【WORK 打工時間】

解釋：每條指令中學生至少執行一種活動，至多執行四種活動（因爲化妝和運動分別爲女生和男生獨有）；活動的出現順序不定；食物數量，學習時間，化妝時間，運動時間均爲1 ~ 120 的正整數。

【學生A】 WOO【學生B】

解釋：學生A 向 學生B求愛

【學生A】【學生B】BREAK UP

解釋：學生A 與 學生B分手

說明：

1. 以上指令描述中【】及其中的漢字代表非終極符，在實際的指令描述文件中爲具體值所代替；大寫的英文字符（無論是否出現在【】中）爲終極符，在實際的指令描述文件中原樣出現。
2. 【房間號】代表某個房間，其格式爲 【樓層號】 - 【某一層中的房間號】，【樓層號】爲兩位整數，取值範圍是01 ~ 99，即便十位爲0也需寫出；【某一層中的房間號】爲兩位整數，取值範圍是01 ~ 50，即便十位爲0也需寫出。
3. 【學生姓名】爲漢字，長度爲2 ~ 4個漢字，不包含空格，數據文件中不會出現不合規則的姓名。
4. 【性別】爲“男”或者“女”，數據文件中不會出現不合規則的性別。
5. 男女生不得同一房間住宿。
6. 女生不會運動，男生不會化妝。
7. 若任何指令及其後果違背了規則（B）、（E）（F）以及常識（譬如不存在的房間號，不存在的學生——入學除外，入住的人數超過了房間的最大容量，沒錢卻執行要花錢的活動，兩人已經是戀人卻再次執行追求指令，兩人不是戀人卻執行分手指令，處於戀愛關係中的人不得追求其他人等），則該指令不予執行。
8. 請自行决定宿舍，樓層，房間的數據結構。
9. 本程序中至少有三個類，男生類，女生類，以及這兩個類的基類——學生類。三個類至少具有的屬性分別是：

學生類的屬性：姓名（不會出現同名的情况），性別，魅力，體重，金錢，知識。

男生類的屬性：繼承而來的屬性以及“健康”。

女生類的屬性：繼承而來的屬性以及“容貌”。

其中金錢，體重爲正實數，知識、健康、容貌、魅力爲1 ~ 100的實數。

1. 每執行一次吃飯、學習、運動、化妝、打工活動後，學生的狀態變化如下：

學生：

**體重增量**：**食物數量 \* 單位食物增加的體重 – 學習時間 \* 單位學習時間减少的體重 – 運動時間 \* 單位運動時間减少的體重**

**金錢**：**打工時間 \* 單位打工時間內獲取的金錢 - 食物數量 \* 單位食物消耗的金錢**

**知識增量：學習時間 \* 單位學習時間增加的知識**

男生：

**健康增量**：**運動時間 \* 單位運動時間增加的健康值**

說明：健康最多增加到100，但學生還可以繼續運動，不算無效指令。

**魅力增量**：**知識增量 / 5 + 健康增量 / 10 – 體重增量 / 5**

說明：魅力最多增加到100，但學生還可以繼續從事增加魅力的活動，不算無效指令。

女生：

**容貌增量**：**化妝時間 \* 單位化妝時間增加的容貌**

說明：容貌最多增加到100，但學生還可以繼續化妝，不算無效指令。

**魅力增量**：**知識增量 / 5 + 容貌增量 / 20 – 體重增量 / 4**

說明：魅力最多增加到100，但學生還可以繼續從事增加魅力的活動，不算無效活動。

1. 當指令描述文件中所有的指令都完成之後，向用戶顯示查詢界面，提供如下查詢功能：
   1. 對于宿舍中的所有學生（包括男女生）、男生、女生，可對體重，金錢，知識，容貌（女生適用），健康（男生適用），魅力進行最大、最小值查詢，範圍查詢（亦即查詢屬性值屬於某個範圍的所有學生）。範圍查詢的結果第一排序依據爲所查詢的屬性，可升序或降序排列，該屬性相等的學生再按照姓名升序或降序排列。
   2. 根據學生姓名（有可能不住在宿舍樓中）查找他（她）所在的房間號。
   3. 根據房間號（有可能該房間號不存在）查找該房間中入住的所有學生的姓名，查詢結果的排序依據爲學生的姓名，可升序或降序排列。
   4. 查詢某個學生的戀愛歷史（按建立戀愛關係的時間排序，戀人姓名排序，魅力值排序），查詢所有學生中戀愛史最多的人，戀人魅力值最高的人。

**注意：在顯示信息時必須合理、美觀，譬如適時地清空整個屏幕；假如信息超出一屏則顯示必須暫停，待用戶指示繼續後再繼續顯示；輸出的信息中應加上適當的空白以方便閱讀。設計良好、簡潔、便于使用的交互界面會有加分，過于拙劣的界面將會扣分。**

**三、課程設計報告**

內容包括：系統功能模塊圖、程序實現思路、設計說明、設計心得和關鍵功能的代碼（關鍵代碼篇幅不得超過整個報告的10%，否則以灌水論處，酌情扣分）。

另外，所提交的程序完整源代碼中一定要有適當的注釋，幷具有良好的編程風格，這些都是評分的標準。

**四、其他事項**

1. 提交源代碼時將所有源文件（包括頭文件）放在一個目錄中，然後將該目錄和設計報告一起打包成一個rar或者zip文件。
2. **千萬不要圖省事將 .exe, .opt, .ncb, .dsw 等非源程序文件打包入所提交的壓縮文件中，否則文件將變得很大，影響下載。**
3. 課程設計成績考核：程序編程測試、設計報告分別占70%、30%。
4. 對于課程設計若有任何疑問請儘快提出，以免影響課程設計的完成。