**CPP程式設計題**

|  |
| --- |
| **命題者︰PWC** |
| **題目名稱(中文/英文)：Student Score System** |
| **主要測試觀念：** function overloading   |  |  | | --- | --- | | **Basics** | **Functions** | | * C++ BASICS 1 * FLOW OF CONTROL * FUNCTION BASICS   ■ PARAMETERS AND OVERLOADING   * ARRAYS * STRUCTURES AND CLASSES * CONSTRUCTORS AND OTHER TOOLS * OPERATOR OVERLOADING, FRIENDS,AND REFERENCES * STRINGS * POINTERS AND DYNAMIC ARRAYS | * SEPARATE COMPILATION AND NAMESPACES * STREAMS AND FILE I/O * RECURSION * INHERITANCE * POLYMORPHISM AND VIRTUAL FUNCTIONS * TEMPLATES * LINKED DATA STRUCTURES * EXCEPTION HANDLING * STANDARD TEMPLATE LIBRARY * PATTERNS AND UML | |
| **題目說明：** Make a student management system, which has below commands:  **insert (name) (score):** Add a student to management system.  If insert a exist name, output error message **"insert fail with &name!!!"**, where **&name** is the inserted name.  **search (name|score):** The search command has 2 kinds of inputs:   1. Input **name**:   Print the student’s name and score. If serach(name) command cannot find the name, output error message **"<name> is not in system!!!".**   1. Input **score**:   If the student score is bigger than the input <score>, print the student’s name and score. Note that records need to be printed in inserted order.  If search(score) command cannot find any score that is bigger than input, output **"no one match the condition".**  **update (name) (newName|newScore):** Update a student’s name or score with **newname** or **newScore**. If update command cannot find the name, output error message **"<name> is not in system!!!"**.  **delete (name):** Delete a student. If delete command cannot find the name, output error message **"<name> is not in system!!!"**.  **score (name):** Print a student’s score represented by grade. If score command cannot find the name, output error message **"<name> is not in system!!!"**.  **show:** Print all names and scores in inserted order. This command doesn’t need to output error message.  Grade is transformed by the rules below:  score >= 90, grade = “A+”  score >= 80, grade = “A”  score >= 70, grade = “B”  score >= 60, grade = “C”  score < 60, grade = “F”  The **Student** class contains two variables: **name** and **score**. A student only has one **score**, and **name** is a English string. The **score** range is 1~100.  **輸入說明：**Input several commands.  **輸出說明：**Output like sample output  **IO範例 :**   |  |  |  | | --- | --- | --- | |  | **Sample Input** | **Sample Output** | | 第一組測資與輸出 | insert john 60 insert john 60 show search john search cena search 60 search 59 update cena allen update cena 70 show update john 70 update john allen show score john score allen delete john delete allen show insert cena 10 insert john 20 insert allen 30 show search 15 delete john show insert john 40 show | insert success with name john insert fail with john!!! john, 60 john score is 60 cena is not in system!!! no one match the condition john cena is not in system!!! cena is not in system!!! john, 60 update value success update name success allen, 70 john is not in system!!! grade: B john is not in system!!! remove allen insert success with name cena insert success with name john insert success with name allen cena, 10 john, 20 allen, 30 john allen remove john cena, 10 allen, 30 insert success with name john cena, 10 allen, 30 john, 40 | | … |  |  | |
| * 易，僅需用到基礎程式設計語法與結構 * 中，需用到多項程式設計語法與結構   ◼ 難，需用到多項程式結構或較為複雜之資料型態或結構 |
| **解題時間：40**分鐘。 |
| **其他註記：** |