1. **Architec****tural Design (class interface) 架構設計範例**

* **class Main**

private IdExist() // 呼叫GradeSystems確認使用者當前輸入的ID是否存在

private UserInput() // 若使用者輸入Q則結束系統，否則呼叫IdExist()確認ID是否存在。若輸入正確ID則正確登入系統，否則回到重新提示使用者輸入

private functionMode() // 使用者登入後即呼叫此function處理使用者輸入

main() // 處理使用者登入及操作系統

* **class UI (user interface)**

checkCommand(String cmd) // 根據使用者輸入(cmd)，呼叫GradeSystems處理

checkID(ID) // 確認輸入ID是否存在於系統

promptCommand() // 提示使用者有5種功能可供使用

promptID() // 提示使用者利用ID登入，Q結束系統

showFinishMsg() // 提示結束訊息

showWelcomeMsg() // 歡迎對應學生登入

showStudentIdWrongMsg() // 提示輸入ID錯誤

UI() 建構子 建構 GradeSystems

* **class GradeSystems**

儲存a list of student grades, a list of weights a list of string for display

containsID(ID) // 看aGradeSystem有否含此ID

showGrade(ID) //顯示成績

showRank(ID) //顯示排名

updateWeights () //更新配分

showAverages() //顯示全班平均

GradeSystems () //建構子

* **class Grades** 儲存 ID, name, gradeList, totalGrade

gradeList的儲存項目分別為lab1, lab2, lab3, midTerm, finalExam

calculateTotalGrade(weights) //計算加權平均

Grades () { } //建構子

**De****tailed Design (design sketch and pseudo-code)**

**細部設計範例**

#####################################################################本Grade system 讓使用者(學生)取得他的總成績total grade 及排名 rank.

Total grade 基於配分weights 來算 而 weights可以update.

Rank 表示此 total grade 在全班學生的分數排序

Input file名字必須為inputdata.txt

Input file:全班學生的分數 例如

962001044 凌宗廷 87 86 98 88 87

962001051 李威廷 81 32 50 90 93

注意 data field names 如下:

ID name lab1 lab2 lab3 midTerm finalExam

#####################################################################

public class Main{

public static void main (String args[])

UI ui = new UI() // 建構UI

while begin

USER\_INPUT() // 使User輸入Q或學號

Function\_Mode() // 使User輸入G, R, A, W, E等功能

end

end class Main

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

class UI (user interface)

checkCommand(String cmd)

checkID(ID)

promptCommand()

promptID()

showFinishMsg()

showWelcomeMsg()

showStudentIdWrongMsg()

UI() 建構子 建構 GradeSystems

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

* public class UI

**UI() 建構子**

* 1. construct a new grade system.

**checkCommand(String cmd)**

1. If cmd equals to G, ask gradeSystems to showGrade(curStudentID), and return true.

2. If cmd equals to R, ask gradeSystems to showRank(curStudentID), and return true.

3. If cmd equals to A, ask gradeSystems to showAverage(), and return true.

4. If cmd equals to W, ask gradeSystems to updateWeights(curStudentID), and return true.

5. If cmd equals to E, return false.

6. Else, show "Input Command Wrong!" and return true.

**checkID(ID)**

1. Record current log-in student ID.

2. Ask gradeSystems to do containID(studentID). Return true if student ID exist in system, otherwise return false.

**promptCommand()**

1. Print out command message.

**promptID()**

1. Print out welcome message.

**showFinishMsg()**

1. Print out shut down message.

**showWelcomeMsg()**

1. Print out welcome message.

**showStudentIdWrongMsg()**

1. Print out student ID wrong message.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

class GradeSystems儲存 a list of student grades, a list of weights, a list of string for display

containsID(ID) //看aGradeSystem有否含此ID

GradeSystems () //建構子

showGrade(ID) //顯示成績

showRank(ID) //顯示排名

updateWeights () //更新配分

showAverages() //顯示全班平均

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

* public class GradeSystems

float[5] weights; (初始值 lab1 0.1, lab2 0.1, lab3 0.1, midTerm 0.3, finalExam 0.4)(注意weights 之和須等於1.0)

String[] gradeItemList; 用於在螢幕上顯示成績所需的文字，為了程式簡潔，列出各項成績的部分皆以for loop來完成，搭配gradeItemList的固定文字和aGrade中gradeList的資料結構，並用if判斷式加註”\*”在小於60的成績之後。Total grade不在gradeList中需另外顯示

用Java LinkedList class 的object叫**aList** (它是 doubly-linked list) 來儲存 **aGrade** objects.

見下面的設計草圖 (design sketch):

|  |  |  |
| --- | --- | --- |
|  | null header |  |

|  |  |  |
| --- | --- | --- |
|  | 962001044凌宗廷 87 86 98 88 87 88 |  |

|  |  |  |
| --- | --- | --- |
|  | 962001051李威廷  81 32 50 90 93 91 |  |

**public GradeSystems () //建構子**

1. 開檔 input file

2. 用**Java** **LinkedList**建構an empty list of grades叫 aList

3. read line

4. while not endOfFile

1.call Grades() 建構aGrade

2.用 **Java BufferedReader** 來 read line 並用split把各欄位存入aGrade

3. aGrade.calculateTotalGrade(weights) 回傳aTotalGrade把它存入aGrade

4. 把 aGrade 存入 aList

5. if endOfLine then read line end if

end while

**public showGrade(ID) { show 這 ID 的 grade }**

1. loop aGrade in aList
2. if ID match, show the grades

**public showRank(ID)**

1. 令rank 為 0
2. 找到ID的aGrade

3. 令rank 為 1

4. loop aGrade in aList if aTotalGrade > theTotalGrade then rank加1(退1名) end loop

4. 回傳 rank

**public updateWeights ()**

1. show old weights

2. get new weights

3. show new weights

4. ask the user if the update is correct

5. set weights and show success messenger if correct

6. loop aGrade in aList to calculateTotalGrade(weights) end loop

public containsID(ID) loop through aList看 ID 是否含在 aGradeSystem內

end class GradeSystems

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

class Grades 儲存 ID, name, gradeList, totalGrade

gradeList的儲存項目分別為lab1, lab2, lab3, midTerm, finalExam

calculateTotalGrade(weights)

Grades () { } //建構子

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

* public class Grades

String name, ID; int gtradeList[5];

(注意 input file內無 totalGrade )

**public Grades () { } //建構子**

產生gradeList的instance

參數依序存入ID,name,gradeList中

**public calculateTotalGrade(weights) {totalGrade須四捨五入}**

1. gradeList的每個項目分別乘上weight後相加
2. 結果取四捨五入存到totalGrade
3. 回傳totalGrade

end class Grades