**Class interface 架構設計**

* **class GradeSystemProject**

main ( )

* **class UI (user interface)**

login()

checkID(ID)

promptCommand()

promptID()

promptExit()

getNewWeights()

showNewWeights(NewWeights)

showFinishMsg()

showWelcomeMsg(id)

weightFunction()

UI() //建構子

* **class GradeSystems(儲存 a list of student grades, 配分，讀檔)**

assignData(data)

showGrade(ID)

showRank(ID)

showAverage()

showOldWeights()

getRank(myGrade)

containsID(ID) // 看aGradeSystem有否含此ID

updateWeights (newWeight)

setNewWeights(newWeights)

findNameByID(ID)

GradeSystems () //建構子

* **class Grades** 儲存 ID, name, lab1, lab2, lab3, midTerm, finalExam, and totalGrade

updateTotalGrade(weights)

getID()

getGrades()

getTotalGrade()

getName()

Grades () { } //建構子

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

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

本GradeSystem讓使用者利用學號(ID)登入，提供功能有顯示成績、顯示排名、顯示平均與更新配分。顯示成績為顯示使用者個人成績，顯示排名為顯示使用者根據總成績在系統中之名次，顯示平均為各項成績之系統平均分數，更新配分為改變各項成績之比重並計算新配分的總成績，更改後會更新整個系統之總成績。

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 extends Object {

public static void main (String args[])

1. call UI() 建構UI
2. loop1 until “Q”

ui.login() 輸入學號

ui.showWelcomeMsg(id) 顯示歡迎訊息

loop2 until “E”

ui.promptCommand() 使用者輸入執行指令

end loop2

end loop1

end class Main

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

class UI (user interface)

login()

checkID(ID)

promptCommand()

promptID()

promptExit()

getNewWeights()

showNewWeights(NewWeights)

showFinishMsg()

showWelcomeMsg(id)

weightFunction()

UI() //建構子

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

public class UI

GradeSystems gradeSystems;

----------------------------------------------------------------------------------------------------------------

UI() 建構子

----------------------------------------------------------------------------------------------------------------

創建GradeSystems()

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

checkID(int id)

----------------------------------------------------------------------------------------------------------------

parameter : id 學生學號

time estimate : O (1)

----------------------------------------------------------------------------------------------------------------

1.gradeSystems call containsID這個method,並回傳之

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

promptID()

----------------------------------------------------------------------------------------------------------------

time estimate : O (1)

----------------------------------------------------------------------------------------------------------------

1.螢幕顯示輸入ID

2.利用變數id記住使用者輸入的學號

3.回傳id

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

showFinishMsg()

----------------------------------------------------------------------------------------------------------------

Time estimate : O (1)

----------------------------------------------------------------------------------------------------------------

1.顯示結束畫面

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

showWelcomeMsg(int id)

----------------------------------------------------------------------------------------------------------------

parameter : id 學號

time estimate : O (1)

----------------------------------------------------------------------------------------------------------------

1.呼叫gradeSystems的findNameByID method取得使用者名稱

2.螢幕顯示歡迎與使用者名稱

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

promptCommand(int id)

----------------------------------------------------------------------------------------------------------------

parameter : id 學號

time estimate : O (1)

----------------------------------------------------------------------------------------------------------------

1.顯示可輸入之指令

2.使用者輸入欲執行項目之相對指令,呼叫相對應的method

3.回傳boolean是否離去

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

weightFunction()

----------------------------------------------------------------------------------------------------------------

time estimate : O (1)

----------------------------------------------------------------------------------------------------------------

1.呼叫showOldWeights顯示當前權重分佈

2.呼叫getNewWeights取得新權重array

3.呼叫showNewWeights顯示更新權重分佈

4.顯示是否確認修改

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

promptExit()

----------------------------------------------------------------------------------------------------------------

time estimate : O (1)

----------------------------------------------------------------------------------------------------------------

1.螢幕顯示使用者需要輸入之指令

2.跑while迴圈,直到使用者輸入對應之指令

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

getNewWeights() throws InputMismatchException

----------------------------------------------------------------------------------------------------------------

time estimate : O (1)

----------------------------------------------------------------------------------------------------------------

1.初始化新的權重陣列皆為0

2.顯示欲輸入之成績名稱,並讀取使用者輸入

3.回傳新的權重陣列

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

showNewWeights(int [] NewWeights)

----------------------------------------------------------------------------------------------------------------

time estimate : O (1)

----------------------------------------------------------------------------------------------------------------

1.顯示相對應之成績與權重

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

login()

----------------------------------------------------------------------------------------------------------------

time estimate : O (1)

----------------------------------------------------------------------------------------------------------------

1.利用promptID()取得學號

2.判斷學號是否存在於系統

3.回傳學號或錯誤代碼

end class UI

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

class GradeSystems

assignData(data)

showGrade(ID)

showRank(ID)

showAverage()

showOldWeights()

getRank(myGrade)

containsID(ID) // 看aGradeSystem有否含此ID

updateWeights (newWeight)

setNewWeights(newWeights)

findNameByID(ID)

GradeSystems () //建構子

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

public class GradeSystems

List<Grades> gradeList = new ArrayList<>(); //儲存每人之資料

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 |  |

float[] weights; //系統當前配分

----------------------------------------------------------------------------------------------------------------

GradeSystems() 建構子

----------------------------------------------------------------------------------------------------------------

讀檔並創建學生資料list，初始化配分

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

assignData(String data)

----------------------------------------------------------------------------------------------------------------

parameter : data 個人資料的字串

time estimate : O (1)

----------------------------------------------------------------------------------------------------------------

1.分解data字串取得各項資料

2.利用各項資料創建grades並回傳

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

showGrade(int id)

----------------------------------------------------------------------------------------------------------------

parameter : id 個人學號

time estimate : O (n) n為系統中總人數

----------------------------------------------------------------------------------------------------------------

1.利用迴圈尋找成績系統list中學號相同者

2.螢幕印出該學號之各項成績

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

showRank(int id)

----------------------------------------------------------------------------------------------------------------

parameter : id 個人學號

time estimate : O (n) n為系統中總人數

----------------------------------------------------------------------------------------------------------------

1.利用迴圈尋找成績系統list中學號相同者

2.將該grades的總成績傳入getRank()獲得排名

3.螢幕顯示排名

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

getRank(int myGrade)

----------------------------------------------------------------------------------------------------------------

parameter : myGrade 總成績

time estimate : O (n) n為系統中總人數

----------------------------------------------------------------------------------------------------------------

1.利用迴圈比較總成績,起始r = 1,大於myGrade者使r+1(排名倒退)

2.回傳r

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

containsID(int id)

----------------------------------------------------------------------------------------------------------------

parameter : id 個人學號

time estimate : O (n) n為系統中總人數

----------------------------------------------------------------------------------------------------------------

1.利用迴圈找出系統list中是否存在相同學號者

2.回傳找到與否

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

updateWeights (int [] getNewWeight)

----------------------------------------------------------------------------------------------------------------

parameter : getNewWeight 新配分

time estimate : O (n) n為系統中總人數

----------------------------------------------------------------------------------------------------------------

1.檢查新比重是否高於100或小於0

2.檢查總和是否為100

3.設定比重為新比重

4.更新系統中每位學生之總成績

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

showOldWeights()

----------------------------------------------------------------------------------------------------------------

time estimate : O (1)

----------------------------------------------------------------------------------------------------------------

1.取得系統目前之各項成績權重

2.螢幕以百分比顯示各項成績權重

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

setNewWeights(int[] newWeights)

----------------------------------------------------------------------------------------------------------------

parameter : newWeights 新配分(百分比)

time estimate : O (1)

----------------------------------------------------------------------------------------------------------------

1.設定各項成績權重

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

findNameByID(int id)

----------------------------------------------------------------------------------------------------------------

parameter : id 個人學號

time estimate : O (n) n為系統中總人數

----------------------------------------------------------------------------------------------------------------

1.利用迴圈尋找系統中該學號對應的名字並回傳

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

showAverage()

----------------------------------------------------------------------------------------------------------------

time estimate : O (n) n為系統中總人數

----------------------------------------------------------------------------------------------------------------

1.利用迴圈加總各項成績

2.各項成績總和除以人數取得平均並顯示至螢幕

end class GradeSystems

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

class Grades

updateTotalGrade(weights)

getID()

getGrades()

getTotalGrade()

getName()

Grades () { } //建構子

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

public class Grades

int id, lab1, lab2, lab3, midterm, finalExam;

String name;

----------------------------------------------------------------------------------------------------------------

Grades() 建構子

----------------------------------------------------------------------------------------------------------------

設定個人成績、學號、名字與計算總成績

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

updateTotalGrade(float[] weights)

----------------------------------------------------------------------------------------------------------------

parameter : weights 成績比重

time estimate : O (1)

----------------------------------------------------------------------------------------------------------------

1.利用比重計算加權後的總成績

2.將計算完的總成績做四捨五入,更新至private totalGrade

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

getID()

----------------------------------------------------------------------------------------------------------------

time estimate : O (1)

----------------------------------------------------------------------------------------------------------------

1.回傳private id

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

getGrades()

----------------------------------------------------------------------------------------------------------------

time estimate : O (1)

----------------------------------------------------------------------------------------------------------------

1.創建一個array儲存個人所有成績

2.回傳該array

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

getTotalGrade()

----------------------------------------------------------------------------------------------------------------

Time estimate : O (1)

----------------------------------------------------------------------------------------------------------------

1.回傳總成績

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

getName()

----------------------------------------------------------------------------------------------------------------

time estimate : O (1)

----------------------------------------------------------------------------------------------------------------

1.回傳個人名字

end class Grades