King Abdulaziz University
Faculty of Computing and Information Systems
Spring 2018 – 2nd Term 2018

Course Code: CPCS203 **Course Name: Programming II**

Assignment # 1 (Exam Grading Program)

Assigned Date : Sunday 11/02/2018
Delivery Date and time: Sunday 25/02/2018 at 11:00PM

WARNING:

- This program must ONLY be submitted on the Blackboard!
- This project worth 6% of the overall module marks (100%).
- NO assignment will be accepted after 11:59 pm for any reason
- Students can submit their assignment between 11 and 11:59 PM but in this case it will be consider as late submission, and they will lose 2 points from the total mark of the assignment.
- For discussion schedule, check the teacher name, date and time on the blackboard. **Further information is provided in the course syllabus.**

Objectives

- Learn how to use and implement multidimensional array.
- Performing procedure on array elements.
- Learn to use and implement String, File I/O (Reading/Writing from/to files).

Delivery

- Submit your assignment on the Blackboard ONLY.
- **Make sure to add your names / IDs / Section / course name / Assignment number, as comment at the beginning of your program**.

Description

This program is an **Examinations Grade Calculating Program** as shown in **figure 1**. Faculty wants you to computerize their Marks calculations for different courses to find the sum of all Exams and the Highest Marks gainer in specific course. Following are the details of the project.

The Data Creation of the Program

Grades For Exam Grading System for the Course CPCS202

Separate 1-D String Array

	Exam Type						
Student Name	Exam1(10)	Exam2(15)	Final(25)	Lab Exams(15)	Assignments(30)	Quiz(5)	∑ (Total)
Fatima	9	15	23	14	25	5	91
Laila	8	13	18	13	28	4	84
Maryam	7	12	16	12	23	5	75
Khalida	8	12	15	10	22	3	70
Zayneb	5	11	15	11	20	2	64
						Max. marks	91

Fatima Topper for 202

Separate 2-D String Array

Separate 3-D int Array to store marks obtained

Figure 1

As you can see from the figure 1, you will need to make different arrays to store all of the data used in the program, for example you need one dimension array to store Exam Types. As students obtain their marks based on the Exam type, such as [Exam1, Exam2, Final Exam, Lab Exams, Assignments and Quizzes] each of which will contains [10, 15, 25, 15, 30 and 5 respectively] total 100 Marks.

Exam Grading program must store the following data:

- ✓ Courses taught in the Faculty.
- ✓ Students name from different courses.
- ✓ Exam type based on which Student Will get the marks.
- ✓ Marks obtained for all Exam type of each student from all the courses.
- ✓ Topper result of the students for each course.
- ✓ Topper result of all students from all courses.

The Initial Procedure of the Program

You will use File I/O to read input from a given input file[input.txt]. Make sure the file exists or display a message that the file does not exist. The file consists of:

- 5 integers to determine array size for course and Total Students registered in the course [see the input file]:

    a. The first number (4) in the file refers to the number of courses in the System [means system will accept ONLY Four courses]
    CPCS202, CPCS203, CPCS204 and CPCS353.
    b. The second number (5) refers to the number of Students registered in the course CPCS202 [means system will accept ONLY 5 students' details for CPCS202]
    c. The third number (3) refers to the number of Students registered in the course CPCS203 [means system will accept ONLY 3 students' details for CPCS203]
    d. The fourth number (4) refers to the number of Students registered in the course CPCS204 [means system will accept ONLY 4 students' details for CPCS204]
    e. The fifth number (3) refers to the number of Students registered in the course CPCS353 [means system will accept ONLY 3 students' details for CPCS353]

The commands you will have to implement are as follows:

- **Add_Course** – Your program must reads Courses code and stores in 1-D array to be used in the system. [see input.txt]

Add_Course CPCS202 CPCS203 CPCS204 CPCS353

In above line **Add_Course** is a command and CPCS202 CPCS203 CPCS204 CPCS353 are course codes.

- **Add_ExamType** – Your program must reads Exam types and stores in an array to be used in the system. [see input.txt]

Add_ExamType Exam1 Exam2 FinalExam LabExams Assignments Quiz

In above line **Add_ExamType** is a command and Exam1 Exam2 FinalExam LabExams Assignments Quiz are six Exam types for the program.

- **Add_Student_For_Course** – Your program must read Students name and store in an array to be used in the system. [see input.txt]

Add_Student_For_Course 0 Fatima Laila Maryam Khalida Zayneb

In above line **Add_Student_For_Course** is a command and 0 is the index, which indicate CPCS202) and Fatima, Laila etc are the name of students registered in CPCS202, similarly,

.

.

Add_Student_For_Course 3 Hina Reem Mona

In above line **Add_Student_For_Course** is a command and 3 is the index, which indicate course CPCS353 and **Hina Reem Mona** are the name of students registered in CPCS353.

- Add_Marks - Your program must reads Students Marks Obtained for a Course, for a student of six Exam types and stores them in an array to be used in the system as in an input file. [see input.txt]

Add_Marks

9 15 23 14 25 5
8 13 18 13 28 4
7 12 16 12 23 5
8 12 15 10 22 3
5 11 15 11 20 2
8 13 24 13 24 5
7 15 23 14 30 5
3 10 20 12 22 5
9 15 25 14 28 4
8 14 24 15 26 5
9 15 23 13 27 5
6 12 22 15 29 4
3 10 22 12 25 3
7 15 23 13 28 4
8 13 24 14 23 5

In above lines **Add_Marks** is a command and in Yellow color five students' marks, indicate course CPCS202 and Fatima Laila Maryam Khalida Zayneb are the name of students registered in CPCS202, similarly

.

.

in green color three students' marks, indicate course CPCS203 and Waleed Rashid Hassan are the name of students registered in CPCS203, and so on.

Note: They will be total 15 students' marks in our case 5+3+4+3

- **Print_Result** – Your program must reads Students marks for a course and automatically calculates the sum of marks of all students out of 100 and finds the maximum marks gainer (Topper) of the course and also stores marks result of each students in an output file. [see print.txt]

Print_Result CPCS202

In above line **Print_Result** is a command and CPCS202 is the Course, your program must read Students marks for course CPCS202 and automatically calculates the sum of marks of all students and finds the maximum marks gainer (topper) of the course CPCS202 and also stores marks result of each students in an output file. [see print.txt], similarly

.

.

Print_Result CPCS353

In above line **Print_Result** is a command and CPCS353 is the Course, your program must read Students marks for course CPCS353 and automatically calculate the sum of marks of all students and finds the maximum marks gainer (topper) of the course CPCS353 and also store marks result of each students in an output file. [see print.txt]

- Print_Result_For_All - Your program must read Students marks for all courses and automatically finds the maximum marks gainer (Topper) of All courses as in an output file. [see print.txt]

[YOU MUST GENERATE EXACTLY SAME OUTPUT FILE AS GIVEN TO YOU (PRINT.TXT)]

The Data Creation of the Program

NOTE: For Marks Array, use the data given in the last few pages of this file.

- Create a Single Dimension Array to store course codes and create a method that read course codes from input.txt file and store in array.

  // String array for the course code

- Create a Single Dimension Array to store Exam Type and create a method that read types of Exam from input.txt file and store in array.

  // String array for the Exam types

- Create a Two Dimension Array to store Students names registered in a Course and create a method that read Student name from input.txt file and store in array.

  // String array for the Students

- Create a 3-Dimentional array to store Marks of each student of different courses for all the Exam Types, and create a method, that reads 3D array of all students Marks. As given Figure 2.

MARKS OBTAINED

		Exam1(10)	Exam2(15)	Final(25)	Lab Exams(15)	Assignments (30)	Quiz(5)
CPCS202	Student-1	9	15	23	14	25	5
	Student-2	8	13	18	13	28	4
	Student-3	7	12	16	12	23	5
	Student-4	8	12	15	10	22	3
	Student-5	5	11	15	11	20	2

		Exam1	Exam2	Final	Lab Exams	Assignments	Quiz
CPCS203	Student-1	8	13	24	13	24	5
	Student-2	7	15	23	14	30	5
	Student-3	3	10	20	12	22	5

		Exam1	Exam2	Final	Lab Exams	Assignments	Quiz
CPCS204	Student-1	9	15	25	14	28	4
	Student-2	8	14	24	15	26	5
	Student-3	9	15	23	13	27	5
	Student-4	6	12	22	15	29	4

		Exam1	Exam2	Final	Lab Exams	Assignments	Quiz
CPCS353	Student-1	3	10	22	12	25	3
	Student-2	7	15	23	13	28	4
	Student-3	8	13	24	14	23	5

Figure 2. Students marks in 3 Dimension Array

The 3- Dimensional array virtually link Course array and student array. Each coursey has number of registered students, You will use the 3 D array to store student marks, so, this 3 Dimensional array have different rows and columns based on the total students registered in the course. Figure 2 and figure 3 illustrates the structure of the 3D array.

courses[0] --> CPCS202	EXAM1	EXAM2	FINAL	LABS	ASSIGN..	QUIZZES
student-1	9	15	23	14	25	5
student-2	8	13	18	13	28	4
student-3	7	12	16	12	23	5
student-4	8	12	15	10	22	3
student-5	5	11	15	11	20	2

Figure 3 The structure of the 3D array

COURSE [0] MARKS

COURSE [1] MARKS

Figure 4 The structure of the 3D array

// Three Dimensional Array

[YOU MUST GENERATE EXACTLY SAME OUTPUT FILE AS GIVEN TO YOU (PRINT.TXT)]

Important Notes:

- Your Code ,output, results etc must be in a readable form.
- Organize your code in separate methods.
- Repeat the program until Quit command is read by your program.
- Use comments in your code.
- Use meaningful variables.
- Use dash lines separator between each method.

Deliverables:

You have to submit **only one java** file.The file name must be like

BA_1110349_P1_ExamGradingProgram_Main.java where BA is your section, **1110349 your ID**

NOTE: your name, ID, and section number must be included as comments in the file!

Output Format

Your program must generate output in a similar format to the given sample output file [print.txt].

Good Luck and Start Early!