COSC1436 – PROJECT

TITLE: File access – Array Access - Grading Students

TIME TO COMPLETE: 7 weeks

HOW TO DO PROJECT
From now and on **yourLastName** will be changed to your last name
***Step1:** Read the requirement, write the **pseudo-code** in a word document by listing the step by step what you suppose to do in main() and then save it with the name as
Project_pseudoCode_OutputPictues_**yourLastName.docx**

***Step2:**
 -start Virtual Studio C++, create the project → project name: FA2021_PROJECT_yourLastName
 -add .cpp file
                FA2021_CSGradingStudentsApplication_yourLastName cpp

***Step3**: follow step by step in the pseudo-code to write the C++ code
***Step4**: compile and run the program
***Step5:** debug if there are any errors to complete the program

PROJECT REQUIREMENT

Use C++ to provide the application that helps to calculate the numeric grade and letter grade of students based on the scores of 8 assignments: policy quiz, quizzes, homework, labs, project, discussion topic, teamwork and tests.

Using the following arrays of size 8 to hold the information 8 assignment types:

const string **ASSIGNMENT_NAMES**[] = { "POLICY QUIZ", "QUIZZES", "HOMEWORK", "LABS", "PROJECT", "TEAMWORK", "DISCUSSION TOPIC", "TESTS" };

int **assignmentSize**[8]; // to store the number of assignments each type, for example assignmentSize[7]=3 if there are 3 tests, assignmentSize[1] = 10 if there are 10 homework

float **maxScore**[8]; //to store the max scores of assignments, for example, maxScore[3] = 30 if max score of each lab is 30 points

float **studentScores**[8]; //to store the scores of students, for example studentScores[7] stores total scores of 3 student tests. studentScores[3] stores total scores of student labs

You may declare one string array **listScores** to store the list of scores of each assignment type **for** the output. For example, users enter 3 tests score:
75.5 82.75 80.5
Then the listScores[7] = "75.5 82.75 80.5"

Then display the following menu:

FA2021_CSGradingStudentsApplication_Smith.cpp
TASK OF GRADING – JAMES SMITH

1.Grading One Student
2.Printing The Grade of One Student
3.Printing The Grades of Class
0.Exit

CASE 1: GRADING ONE STUDENT
Read course name, for example COSC1436

INSIDE THE LOOP:
Read student id, last name, first name of one student
Read the scores of 8 assignment types to store in the array studentScores. The scores of one assignment type are provided separate by spaces. For example: if there are 3 tests then input looks like below:
78.5 80.25 82.5
The scores are read and **add up together** then store to the array at studentScore[7] = 241.25
To keep the list of scores of each assignment type for output, you should store the string of assignment scores

Calculate the total of scores stored in the array **studentScore**s,
        Total student score = studentScores[0] + studentScores[1] + …
Calculate total **max scores** in the array maxScore. Since the policy quiz is bonus score, **the total max score does not include maxScore[0]** ➔ total max score = maxScore[1]*assignmentSize[1] + maxScore[2] *assignmentSize[2]+ ….. + maxScore[7]*assignmentSize[7]

Calculate **the numeric grade** = 100 * total student score / total max score
With numeric grade, determine the **letter grade** based on the ranges in the following table:

Letter Grade	Numeric grade
A	90 - 100
B	80-89
C	70-79
D	60-69
F	0-59

Display the output in the following:

```
FA2021_CSGradingStudentApplication_Smith.cpp
FINAL GRADE OF STUDENT - JAMES SMITH
----------------------------------------------------------------------------
COURSE NAME:         COSC1436
STUDENT:             1234567
Name:                Mary Lane
POLICY QUIZ:         5.00
QUIZZES:             4.50 3.25 2.75 5.00 5.00 4.25 5.00 2.25 4.50 5.00 2.50 3.50 4.00 4.50
HOMEWORK:            7.50 8.00 9.50 10.00 10.00 7.50 9.25 9.00 8.75 10.00
LABS:                27.00 25.50 22.50 28.00 22.25 25.50 30.00
PROJECT:             65.00
TEAMWORK:            20.00
DISCUSSION TOPIC:    20.00
TESTS:               75.00 82.50 78.25
----------------------------------------------------------------------------
Total scores:        672.00/800
Numeric grade:       84.00
Leter grade:         B
```

WRITE TO FILE:
Display message to ask and read the answer: If users want to store the grade of this student to file
-If yes, ask for file name then **open output file** to append, then **write one line** to the file in the following format:

 studentId**,**firstname lastname**,**numericGrade**,**letterGrade,5.00,4.5 3.25 2.75 5.00 4.25 5.00, 2.25 4,50 2.50 3.50 4.00 4.50, 7.50 8.00 9.50 10.00 10.00 7.50 9.25 9.00 8.75 10.00, 27.00 25.50 22.50 28.00 22.25 25.50 30.00, 65.00, 20.00, 20.00, 75.00 82.50 78.25
Close file

ASK IF CONTINUE GRADING OTHER STUDENTS
-if yes, loop back for grading one student; -If no, display the main menu of Tasks

CASE 2: PRINT THE GRADE OF ONE STUDENT
-Ask users to provide the ID of the student to print
-open file studentGrade.txt to read
-for each line if the student id matches to the ID then split information of the line into studentid, name, numericGrade, letterGrade, score of policy quiz, list of quizzes score, list of homework scores, list of lab scores, project score, teamwork score, topic score, list of test scores
-provide the output of the grade of the student in the following format

```
STUDENT:             1234567
STUDENT NAME:        Mary Lane
POLICY QUIZ:         5.00
QUIZZES:             4.50 3.25 2.75 5.00 5.00 4.25 5.00 2.25 4.50 5.00 2.50 3.50 4.00 4.50
HOMEWORK:            7.50 8.00 9.50 10.00 10.00 7.50 9.25 9.00 8.75 10.00
LABS:                27.00 25.50 22.50 28.00 22.25 25.50 30.00
PROJECT:             65.00
TEAMWORK:            20.00
DISCUSSION TOPIC:    20.00
TESTS:               75.00 82.50 78.25
NUMERIC GRADE:       84.00
LETTER GRADE:        B
```

CASE 3: Display the Grades of class
-ask for the file name. -Open file to read. If the file exists read each line if the file
For each line, split information into studentId, name, numeric grade, letter grade.
Display the information of each on screen in the following format:

```
FA2021_CSGradingStudentApplication_Smith.cpp
LIST OF STUDENTS' GRADES - JAMES SMITH
------------------------------------------------------------
COURSE NAME:        COSC1436
STUDENT ID:         1212121
NAME:               Isra Ahmed
NUMERIC GRADE:      84.00
LETTER GRADE:       B
------------------------------------------------------------
COURSE NAME:        COSC1436
STUDENT ID:         1313131
NAME:               Jawad Ahmed
NUMERIC GRADE:      90.93
LETTER GRADE:       A
------------------------------------------------------------
COURSE NAME:        COSC1436
STUDENT ID:         1414141
NAME:               Adje Akoi
NUMERIC GRADE:      71.53
LETTER GRADE:       C
------------------------------------------------------------
COURSE NAME:        COSC1436
STUDENT ID:         1515151
NAME:               Toan Cao
NUMERIC GRADE:      82.5
LETTER GRADE:       B
------------------------------------------------------------
```

HOW TO TURN IN

You should turn in the following files:

Project_pseudoCode_OutputPictues_yourLastName.docx (pseudocode and output pictures)

FA2021_CSProductSaleApplication_yourLastName.cpp

FA2021_PROJECT_yourLastName.exe

IF YOU GET ANY PROBLEM TO SUBMIT FILE .exe, YOU CAN SUBMIT ALL PROJECT FOLDER

INTO ONE FILE .zip or .rar

HOW TO GRADE THE LAB

ITEMS	SCORES
TURN IN ON TIME	10
Submit all files requested	3
compile success with all the requirements	10
Pseudo-code or flowchart	5
Write the comment in the program	5
Using at least 5 user-defined functions and call them in main() with reference parameters or array parameters	5
Using at least 3 arrays	3
Manage main menu to re-display after finishing each task and terminate when users selects exit	3
TASK1: Grading one student process: read info, scores, calculate the numeric grade correct, letter grade correct, output correct format	15
Write to output file append, correct format, close file	3
Loop back to allow users to work on other student	2
TASK 2 Open file, ask ID, read file, compare id Print the grade of one student as required format	8
TASK3 Open file to read, open file to write, close file, while loop to read the file, read one line, display on the screen	8
PROJECT SCORES	80 POINTS