ASSIGNMENT 2

PROGRAMMING TECHNIQUE 1 (SECJ1013)

SECTION 04, SEM 1 (2024/2025)

INSTRUCTIONS TO THE STUDENTS

- *This assignment must be done **in pairs**, except for students explicitly instructed to complete it individually.*
- *Please check your name in the **Assignment 2 & 3 Groups List**:*
  - ✓ *If your name is listed, you are the **leader** of your group.*
  - ✓ *As a leader, choose a partner from students not listed in the group and complete the partner's name in the provided table.*

- ***Program Requirements:***
  - ✓ *Your program must strictly follow the input and output formats as specified in the question and examples. Test your program thoroughly using the given input examples and other possible test cases.*
  - ✓ *Plagiarism is strictly prohibited. Students who copy or share their work will receive ZERO marks (both the one who copies and the one who shares).*
  - ✓ *Programs detected as being 100% generated by AI tools will also receive ZERO marks.*
  - ✓ *Include the following details in the comments section of your program:*
    - *Your name and your partner's name (if applicable)*
    - *Matric number(s)*
    - *Date of completion of the assignment.*

SUBMISSION PROCEDURE

- *Submit the assignment **before Sunday, January 12, 2025, at 12:00 AM**.*
- *Only one submission per group is required, which includes two files: the source code (the file with the extension .cpp) and the input file (the file with the extension .txt).*
- *Submit your assignment via the UTM e-learning system.*
- *Your submission will be evaluated based on correctness, clarity, formatting, and adherence to the requirements.*

QUESTION

Develop a C++ program to manage and analyze student performance in a course. The program should calculate total marks, assign grades, and provide statistical analysis of overall student performance. This assignment evaluates your understanding of arrays, functions, control structures, file operations, and output formatting. The tasks for this program are outlined below:

Task 1: Declare Parallel Arrays

Declare the following parallel arrays:

(a) Student names.
(b) Marks for assignments (20), lab exercises (15), midterms (30), and final exams (35).
(c) Total marks.
(d) Grades.

Task 2: Implement the `readInput` Function

Write a function named `readInput` that:

(a) Accepts arrays for student name, assignment marks, lab exercise marks, midterm marks, and final exam marks.

(b) Reads data from an input file containing student records in the following format: *Student Name,Assignment Marks Lab Exercise Marks Midterm Marks Final Exam Marks*. Figure 1 shows an example of data in the input file.

```
Siti Aisyah Halim,18 12 22 28
Rahim Abdullah,19 13 24 20
Lee Mei Yin,15 13 20 26
Nurul Huda Rosli,10 9 13 20
Tan Wei Chong,16 14 21 27
Samuel Lim,17 12 22 25
Amina Rosli,19 14 25 29
Kumar Devan,14 12 18 20
Zainab Ismail,18 8 17 23
Shahrul Nizam Abu,9 7 17 18
Ahmad Faizal Ali,20 15 25 30
Faridah Hamzah,15 11 19 25
Hafiz Norhadi,18 13 23 27
Amirah Zahid,16 11 22 29
Zulkifli Hassan,11 8 13 15
Aisyah Abdullah,19 14 24 28
Faisal Ahmad,20 12 26 32
Nadia Rahman,5 2 8 15
Luqman Hakim,14 9 17 20
Zarina Ismail,17 12 20 32
Hafizah Razali,16 13 21 28
Haziq Roslan,18 11 13 20
```

Figure 1: Sample data in the input file

(c) Dynamically calculates the total number of students based on the file contents.

(d) Populates the parallel arrays accordingly.

(e) Returns the total number of students.

Task 3: Implement the `calculateTotalAndGrade` Function

Write a function named `calculateTotalAndGrade` that:

(a) Accepts arrays for student name, assignment marks, lab exercise marks, midterm marks, and final exam marks, total marks and grades, along with the total number of students.

(b) Calculates total marks for each student using the formula:

Total Marks = Assignment + Lab Exercise + Midterm + Final Exam

(c) Assigns grades based on total marks using the following criteria:

Total Marks (%)	Grade
85-100	A
70-84	B
55-69	C
40-54	D
Below 40	F

Task 4: Implement the `displayOutput` Function

Write a function named `displayOutput` that:

(a) Is a non-returning function.
(b) Accepts the following arrays and the total number of students as input parameters: student name, assignment marks, lab exercise marks, midterm marks, final exam marks, total marks and grades.
(c) Displays student records in a tabular format, including Student Name, Assignment Marks, Lab Exercise Marks, Midterm Marks, Final Exam Marks, Total Marks, and Grade.
(d) Figure 2 provides an example of the output displayed on the screen, based on the data from the input file illustrated in Figure 1.

```
Student Name        Assignments   Lab Exercises   Midterm Tests   Final Exams   Total Marks   Grade
-------------------------------------------------------------------------------------------------
Siti Aisyah Halim      18.0           12.0            22.0           28.0         80.00         B
Rahim Abdullah         19.0           13.0            24.0           20.0         76.00         B
Lee Mei Yin            15.0           13.0            20.0           26.0         74.00         B
Nurul Huda Rosli       10.0            9.0            13.0           20.0         52.00         D
Tan Wei Chong          16.0           14.0            21.0           27.0         78.00         B
Samuel Lim             17.0           12.0            22.0           25.0         76.00         B
Amina Rosli            19.0           14.0            25.0           29.0         87.00         A
Kumar Devan            14.0           12.0            18.0           20.0         64.00         C
Zainab Ismail          18.0            8.0            17.0           23.0         66.00         C
Shahrul Nizam Abu       9.0            7.0            17.0           18.0         51.00         D
Ahmad Faizal Ali       20.0           15.0            25.0           30.0         90.00         A
Faridah Hamzah         15.0           11.0            19.0           25.0         70.00         B
Hafiz Norhadi          18.0           13.0            23.0           27.0         81.00         B
Amirah Zahid           16.0           11.0            22.0           29.0         78.00         B
Zulkifli Hassan        11.0            8.0            13.0           15.0         47.00         D
Aisyah Abdullah        19.0           14.0            24.0           28.0         85.00         A
Faisal Ahmad           20.0           12.0            26.0           32.0         90.00         A
Nadia Rahman            5.0            2.0             8.0           15.0         30.00         F
Luqman Hakim           14.0            9.0            17.0           20.0         60.00         C
Zarina Ismail          17.0           12.0            20.0           32.0         81.00         B
Hafizah Razali         16.0           13.0            21.0           28.0         78.00         B
Haziq Roslan           18.0           11.0            13.0           20.0         62.00         C
```

Figure 2: Example of output

Task 5: Implement the `displayAnalysis` Function

Write a function named `displayAnalysis` that:

(a) Is a non-returning function.
(b) Accepts arrays for student name, total marks, and grades, along with the total number of students.
(c) Calculates and displays names and marks of the students with the highest and lowest scores.
(d) Calculates and displays average marks of the class.
(e) Calculates number of students in each grade category.
(f) Displays a grade distribution histogram where each grade is represented by a horizontal bar made up of +, with each "++" representing one student.
(g) Figure 3 provides an example of the analysis results displayed on the screen, based on the data from the input file illustrated in Figure 1.

```
Class Average Marks: 70.73
Highest Score: Ahmad Faizal Ali (90.00)
Lowest Score: Nadia Rahman (30.00)

Grade Histogram
A: ++++++++ (18.18%)
B: ++++++++++++++++++++ (45.45%)
C: ++++++++ (18.18%)
D: ++++++ (13.64%)
F: ++ (4.55%)
```

Figure 3: Example of analysis results

Task 6: Implement the `main` Function

Write a `main` function to:

(a) Declare the necessary parallel arrays as specified in Task 1 with a maximum size of 50 elements.
(b) Call the functions implemented in Tasks 2–5 to:
- ✓ Read input data.
- ✓ Calculate total marks.
- ✓ Display individual student information.
- ✓ Display student performance analysis.

(c) Ensure the output is formatted properly.

Additional Notes:

(a) Include meaningful comments in your code for clarity.
(b) Handle file-related errors gracefully (e.g., file not found).
(c) Use appropriate formatting for both input and output operations to ensure user readability.