Assignment 1 (10%)

Overview

Write a program to calculate a final grade in the BTE2313 course.

- Prompt the user for the *Coursework* marks and *Test* marks (assuming maximum marks for both are 100). These values should be stored in separate variables.
- Calculate the *Final* grade as the sum of 35% of the *Coursework* and 65% of the *Test* marks.
- display the *Final* grade (use the sample output for the details on how the output should look like)
- The process of getting inputs and calculating the grade will be repeated, until the user enter command/character/value to quit/exit the program. *(*hint: use loop)*

Processing Requirements

1. At the top of your C++ source code, include a documentation box that resembles the following:

```
/*************************************************************
BTE2313                  Assignment 1 – Part A

Programmer:

Date:

Purpose: This program calculates and displays a final grade for
         the BTE2313 course.
*************************************************************/
```

2. Include the following lines of code BELOW the documentation box:

```
#include <iostream>
#include <iomanip>

using namespace std;
```

3. Use type float or double for the two marks and the final grade. Use meaningful variable names.
4. Make sure and test your program with values other than the ones supplied in the sample output.
5. Hand in a copy of your source code (the .cpp file, example of file name *asg1_partA_sulastri.cpp*) in Kalam

Sample Output:

A single run of the program may resemble the following:

```
Enter the coursework marks:
78
Enter the test marks:
50

********************************
Grade calculator

Coursework Marks (35%)  27.30
Test Marks (65%)        32.50

Final Grade             59.80
********************************


Do you wish to continue? (Y/N)
y

Enter the coursework marks:
80
Enter the test marks:
67

********************************
Grade calculator

Coursework Marks (35%)  28.00
Test Marks (65%)        43.55

Final Grade             71.55
********************************


Do you wish to continue? (Y/N)
n

--------------------------------
Process exited after 12.05 seconds with return value 0
Press any key to continue . . .
```

Use various *cout* statements to make the line (of *'s) as well as the text. Be sure that the amounts are lined up vertically as shown.

Part B

Overview

Write a program to calculate a final grade in the BTE2313 course.

- Calculate the *Final* grade as the sum of 35% of the *Coursework* and 65% of the *Test* marks (just like in Part A)
- Use the calculated *Final* grade to determine (and display) the letter grade. The basic scale is:

  A: 90.00% - 100%
  B: 80.00% - 89.99%
  C: 70.00% - 79.99%
  D: 60.00% - 69.99%
  F: 0.00% - 59.99%

Also make sure to take into account the ***special rule*** about having to pass both parts of the course. A student must achieve *Test* and *Course-work* marks of at least **55 marks** to pass the course. If a student receives less than 55 marks on either part of the course, display a letter grade of F and a brief explanation as to why that grade was earned.

Processing Requirements

1. At the top of your C++ source code, include a documentation box that resembles the following:

```
/*************************************************************
BTE2313                  Assignment 1 – Part B

Programmer:

Date:

Purpose:
*************************************************************/
```

2. Include the following lines of code BELOW the documentation box:

```
#include <iostream>
#include <iomanip>

using namespace std;
```

3. Use type float or double for the two marks and the final grade. Use meaningful variable names.
4. Make sure and test your program with values other than the ones supplied in the sample output.
5. Hand in a copy of your source code (the .cpp file, example of file name *asg1_partB_sulastri.cpp*) in Kalam

Sample Output:

A single run of the program may resemble the following:

SAMPLE 1

```
Enter the test scores: 78
Enter the maximum test scores: 67


********************************
     Grade Calculator

Coursework (35%)       27.30
Test (65%)             43.55

Final Grade            70.85
Letter Grade           C
********************************
```

SAMPLE 2

```
Enter the coursework marks: 75.26
Enter the test marks: 52.31


********************************
Grade Calculator

Coursework (35%)       26.34
Test (65%)             34.00

Final Grade            60.34
Letter Grade           F - the coursework or test average was below 55%
********************************
```

Use various *cout* statements to make the line (of *'s) as well as the text. Be sure that the amounts are lined up vertically as shown.

Codes Evaluation – 30 marks

Part A: 15 marks

The source code will be evaluated in the following format:

Items	**Rubrics**
cin, cout (4 statements or more)	2 marks
Pre-processor*: iostream, iomanip, etc.*	2 pre-processors and more: 1 mark 1 pre-processor only: 0 mark
Readability	Easy to read: 1 mark Poorly organized: 0 mark
Variables	Good names: 2 marks Appropriate names: 1 mark Bad variable name: 0 mark
Comments	Comments are properly placed: 2 marks Unnecessary comment/very minimum: 1 mark No comment at all: 0 mark
Proper indentations	Yes: 1 mark No: 0 mark
Displayed Outputs	Precisely aligned: 2 marks Some misaligned: 1 marks Unaligned: 0 mark
Control structure statement (loop)	1 statement: 1 mark 0 statement: 0 mark
Displayed Numbers	Use formatted numeric features: 1 mark Did not use formatted numeric feature: 0 mark
Could the program be executed?	Yes: 2 marks Yes, but with run-time/logic error: 1 mark No: **-2 mark**

Part B: 15 marks

The source code will be evaluated in the following format:

Items	**Rubrics**
cin, cout (4 statements or more)	2 marks
Pre-processor*: iostream, iomanip, etc.*	1 mark
Readability	Easy to read: 1 mark Poorly organized: 0 mark
Variables	Appropriate names: 1 mark Bad variable name: 0 mark
Comments	Comments are placed: 1 mark No comment at all: 0 mark
Indentations	Proper indentations: 2 marks Some indentation: 1 mark No proper indentation: 0 mark
Displayed Outputs	Precisely aligned: 2 marks Some misaligned: 1 marks Unaligned: 0 mark
Decision (*if..else, switch..case*) (4 statements or more)	2 marks
Submission on time	Yes: 1 mark No: 0 mark
Could the program be executed?	Yes: 2 marks Yes, but with run-time/logic error: 1 mark No: **-2 mark**