Week Assignment (3 marks)

Note: Try to make the best use of appropriate C++ features.

The goal of this assignment is to develop a Student Grade Management System using modern C++ programming techniques. The system should allow you to add students, record and validate their grades, calculate average scores, identify top and failing students.

Tasks:

1. Define a NoGradesException Class

- Inherit from std::exception.
- Store an error message as a private std::string.
- Construct the message like:
  " Bob: No grades available."
- Override the what() method.

2. Implement a Student Class

This class represents an individual student.

- **Private members**:
  - std::string name
  - std::vector<double> grades
- **Constructor**:
  - Accepts the student's name.
- **Public methods**:
  - void addGrade(double grade)
    - Adds a grade to the student.
    - Throws std::invalid_argument if the grade is not between 0.0 and 100.0.
  - double averageGrade() const
    - Returns the average of the grades.
    - Throws a **NoGradesException** if the student has no grades.
  - std::string getName() const
    - Returns the name of the student.
  - bool hasGrades() const
    - checks whether the student has any grades recorded.

3. Implement a StudentManager Class

This class manages a group of Student objects.

- **Private member**:
  - std::vector<Student> students
- **Public methods**:
  - void addStudent(const Student&)
    - Adds a student to the system.
  - void filterFailingStudents() const
    - Prints failing students whose average is **below 50.0**.

    - Skips students with no grades.
    - Prints “None.” if no failing students.
  - void printTopStudents(int n) const
    - Print the top n students based on the average grade, along with their average grade, formatted to **two decimal places**, like
      Kevin: 76.66
    - Prints “None.” if no students.
  - void operator()() const
    - Prints students with their average grade, formatted to two decimal places.
    - Prints “None.” if no students.

Input:

Please use the provided ***run_wa7.cpp*** to test your implementation.

Don’t modify the main function in ***run_wa7.cpp***

Output: (should be like the below)

```
Output 1:
Invalid grade. Must be between 0.0 and 100.0.
Bob: No grades available.

Output 2:
Student Management System:
None.
------------------------------
Failing students:
None.
------------------------------
Top 2 students:
None.

Output 3:
Student Management System:
Alice: 83.58
Bob: No grades available.
Charlie: 42.50
Kevin: 55.00
------------------------------
Failing students:
 - Charlie
------------------------------
Top 2 students:
1. Alice : 83.58
2. Kevin : 55.00
```

Submit:

For a more efficient marking process, we kindly request that you submit each file individually, instead of in a zip file.

1, all C++ source code

2, **wa.txt**: a txt file contains all the source code for plagiarism review.
(If you are unable to create the .txt file correctly, please configure your system to display file extensions.)
3**,** **output.jpg** (or png, bmp): a screenshot of the output by your program.

Please refer to the submission page for the Marking Rubric.