```
Algorithm StudentGradingSystem
BEGIN
    WHILE True
        PRINT "=== STUDENT GRADING SYSTEM ==="
        PRINT "1. Enter student data"
        PRINT "2. Exit"
        PRINT "Choose option: "
        READ choice

        IF choice == 2 THEN
            PRINT "Thank You!"
            BREAK
        ELSE IF choice == 1 THEN
            CALL ProcessStudent()
        ELSE
            PRINT "Invalid choice! Please try again."
        END IF
    END WHILE
END

ProcessStudent()
RETURNS Void
START
    Declare total_marks as FLOAT = 0
    Declare subject_count as INTEGER = 0
    Declare student_name as STRING
    Declare student_id as STRING
    PRINT "Enter student name: "
    READ student_name
    PRINT "Enter student ID: "
    READ student_id

    Declare num_subjects as INTEGER
    Declare marks as FLOAT
    Declare average as FLOAT
    WHILE True
        TRY
            PRINT "Enter number of subjects: "
            READ num_subjects

            IF num_subjects <= 0 THEN
                THROW Exception("Number of subjects must be positive")
            END IF
            BREAK
        CATCH Exception
            PRINT "Error: Please enter a valid positive number"
        END TRY
    END WHILE


    FOR i = 1 TO num_subjects
        WHILE True
            TRY
                PRINT "Enter marks for subject " + i + " (0-100): "
                READ marks
                IF marks < 0 OR marks > 100 THEN
                    THROW Exception("Marks must be between 0 and 100")
                END IF

                total_marks = total_marks + marks
                subject_count = subject_count + 1
```

```
                BREAK

            CATCH Exception
                PRINT "Error: Invalid marks! Please enter between 0-100"
            END TRY
        END WHILE
    END FOR

    average = total_marks / subject_count
    grade_letter = CalculateGrade(average)


    CALL DisplayReport(student_name, student_id, total_marks, average, grade_letter)
END

FUNCTION CalculateGrade(average as FLOAT)
RETURNS STRING
START
    IF average >= 90 THEN
        RETURN "A"
    ELSE IF average >= 80 THEN
        RETURN "B"
    ELSE IF average >= 70 THEN
        RETURN "C"
    ELSE IF average >= 60 THEN
        RETURN "D"
    ELSE
        RETURN "F"
    END IF
END

FUNCTION DisplayReport(name as STRING, id as STRING, total as FLOAT, avg as FLOAT, grade as
STRING)
RETURNS Void
START
    PRINT "=== STUDENT GRADE REPORT ==="
    PRINT "Name: " + name
    PRINT "ID: " + id
    PRINT "Total Marks: " + total
    PRINT "Average: " + ROUND(avg, 2)
    PRINT "Grade: " + grade
    PRINT "============================"
END
```