**STUDENT MANAGEMENT SYSTEM (SMS)**

**CODE EXPLAINATION**

**CREATED BY: TASLEEM AHMAD**

## 1. **Overview**

This program implements a comprehensive **Student Management System (SMS)**. It is designed to handle multiple operations for three types of users: **Admin**, **Teacher**, and **Student**. The system allows managing teacher and student records, assigning grades, calculating CGPA, and generating various reports. The program includes data persistence through file storage.

## 2. **Key Modules and Functions**

The program is divided into several modules, each responsible for a specific functionality:

### **Admin Functions**

* **addteacher()**: Adds a new teacher to the system, ensuring validation for input such as name, ID, and class assignment.
* **deleteteacher()**: Removes a teacher based on their unique ID.
* **saveTeacherToFile() and loadTeacherFromFile()**: Handles saving and loading teacher records to and from the file system.

### **Teacher Functions**

* **addstudent()**: Adds a new student to the system with proper input validation.
* **deletestudent()**: Deletes a student based on their unique ID.
* **assignmarks()**: Assigns marks to students for quizzes, assignments, midterms, and finals.
* **calculateGrades()**: Computes grades based on weighted scores and assigns letter grades.
* **calculateCGPA()**: Calculates the cumulative grade point average (CGPA) for students.
* **showAverageCGPA()**: Displays the average CGPA of all students.
* **listFailedStudents() and listPassedStudents()**: Generates lists of students based on CGPA thresholds.
* **viewMarks()**: Displays detailed marks for all students in a specific subject.

### **Student Functions**

* **viewStudentMarks()**: Allows students to view their marks in quizzes, assignments, midterms, and finals for a specific subject.
* **viewStudentGrades()**: Displays the student’s grades for all subjects.
* **viewStudentCGPA()**: Shows the calculated CGPA for a student.

### **Data Persistence Functions**

* **saveStudentToFile() and loadStudentFromFile()**: Handles saving and loading student records to and from the file system.
* **saveMarkstofile() and loadMarks()**: Handles saving and loading marks data for students.

## 3. **Key Variables**

* **Admin Credentials**:
  + admin\_pass: Password for admin login.
* **Teacher Records**:
  + teacher\_name, teacher\_subject, teacher\_class, teacher\_password, teacher\_ID: Arrays storing teacher details.
  + current\_teacher\_count: Tracks the number of teachers in the system.
* **Student Records**:
  + student\_ID, student\_name, student\_class, student\_password: Arrays storing student details.
  + current\_student\_count: Tracks the number of students in the system.
* **Marks Storage**:
  + student\_quizzes, student\_assignments, student\_midterms, student\_finals: Multi-dimensional arrays storing marks.
* **File Paths**:
  + subject\_files, cgpa\_output\_file,subject\_file\_path: Stores file paths for data persistence.

## 4. **Execution Flow**

### **Main Function**

1. **Initialization**:
   * Loads teacher and student data from files.
2. **Main Menu**:
   * Displays options to log in as Admin, Teacher, or Student.
   * Processes user input to navigate to the respective sub-menu.

### **Admin Workflow**

* Admin logs in using a password.
* Can add or delete teacher records.

### **Teacher Workflow**

* Teacher logs in using their ID and password.
* Can perform the following actions:
  + Add or remove students.
  + Assign marks and calculate grades.
  + Generate CGPA and view average CGPA.
  + List passed or failed students.
  + View marks for all students in their subject.

### **Student Workflow**

* Student logs in using their ID and password.
* Can view:
  + Their marks for quizzes, assignments, midterms, and finals.
  + Their grades for all subjects.
  + Their CGPA.

## 5. **Features**

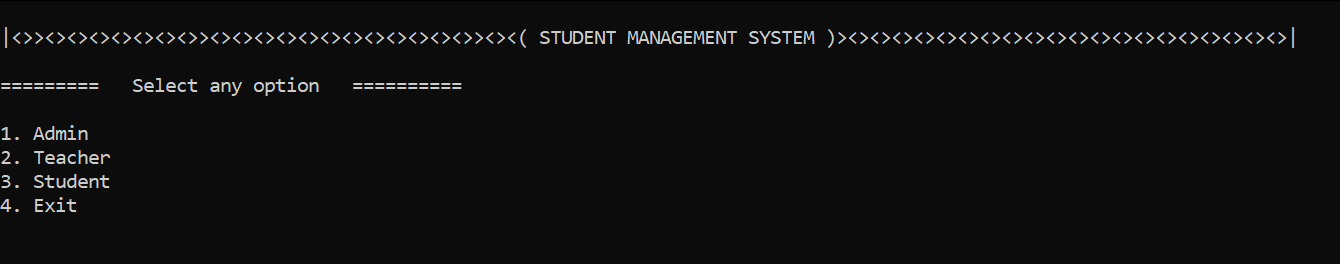
* **Role-based Access**:
  + Admin, Teacher, and Student have distinct privileges and access levels.
* **Data Validation**:
  + Ensures input data meets specified requirements (e.g., valid IDs, marks range).
* **File Persistence**:
  + Saves data (e.g., teacher/student records, marks) to text files for future use.
* **Grading System**:
  + Calculates grades based on weighted scores for quizzes, assignments, midterms, and finals.
* **CGPA Calculation**:
  + Supports calculating and displaying students' CGPA.
* **Reports**:
  + Generates reports for passed/failed students and detailed marks.

## 6. **Usage Instructions**

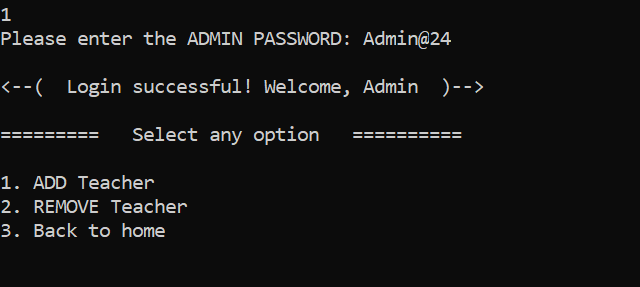
1. **Compiling the Code**:
   * Use a C++ compiler (e.g., g++, Visual Studio) to compile the program.
   * Ensure the necessary text files for data persistence exist in the same directory as the executable.
2. **Running the Program**:
   * Launch the executable.
   * Select the appropriate role (Admin, Teacher, or Student) and log in.
   * Follow the on-screen instructions to perform desired actions.
3. **File Management**:
   * Ensure the following files are maintained for smooth functioning:
     + teachers data.txt
     + students data.txt
     + Subject-specific files (e.g., calculus\_grades.txt).
     + CGPA\_file.txt

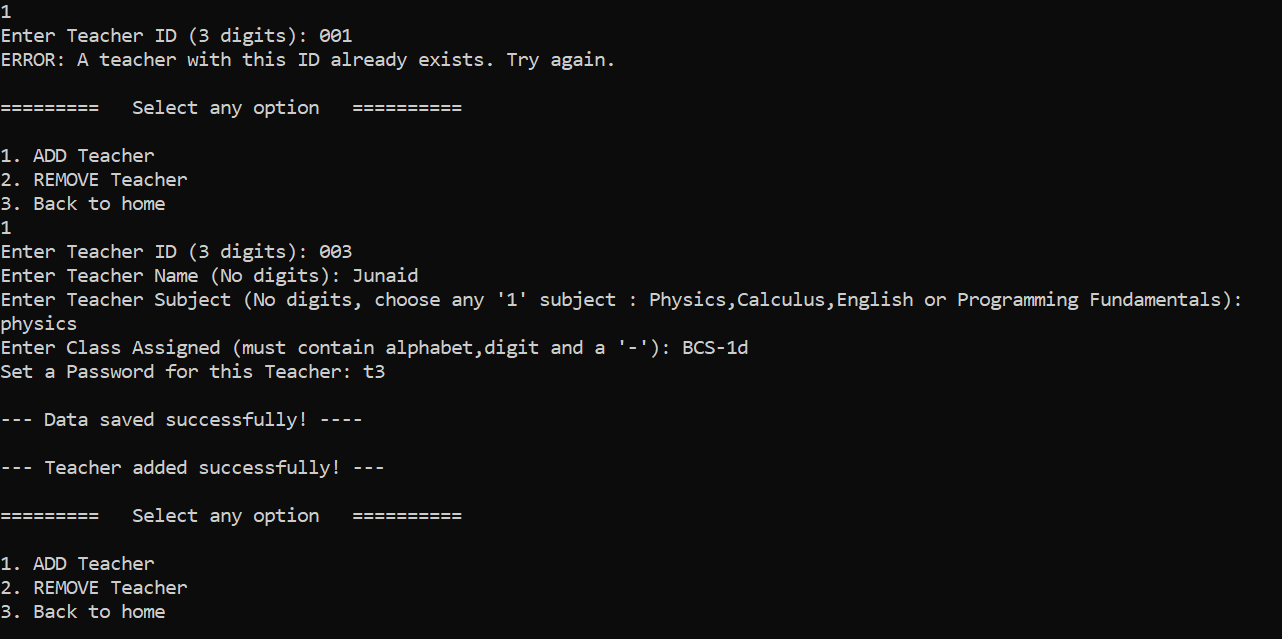
**OUTPUT SCREENSHOTS:**

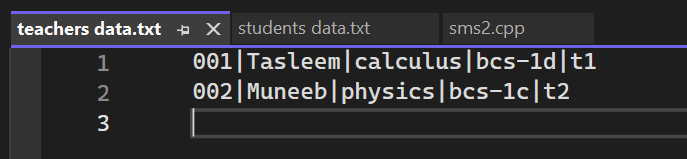
**HOME PAGE:**



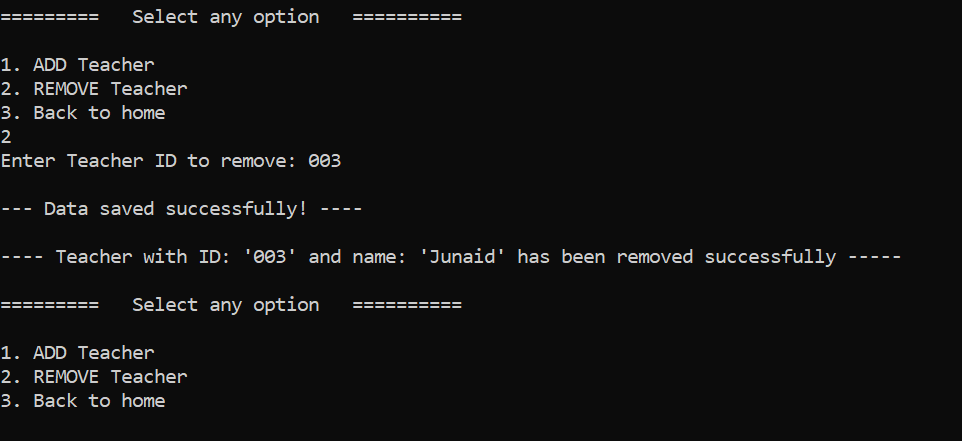
**ADMIN menu:**



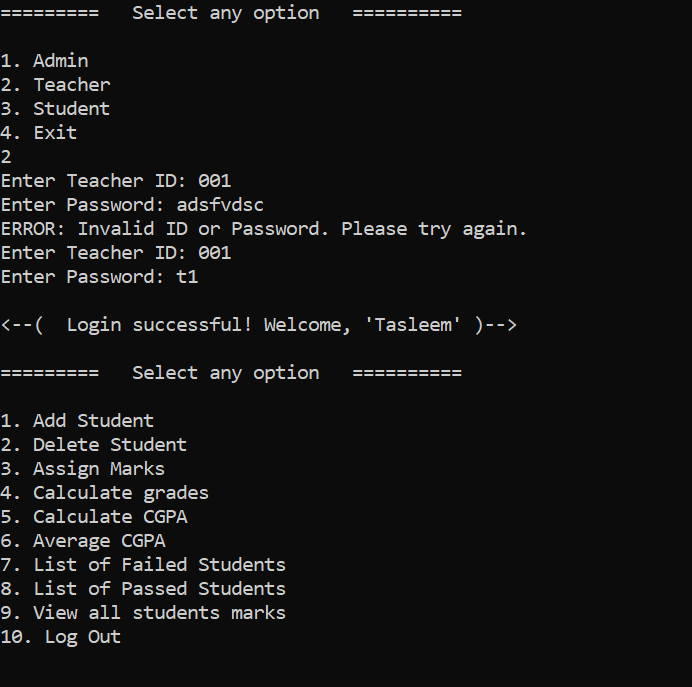
**Adding a Teacher:**



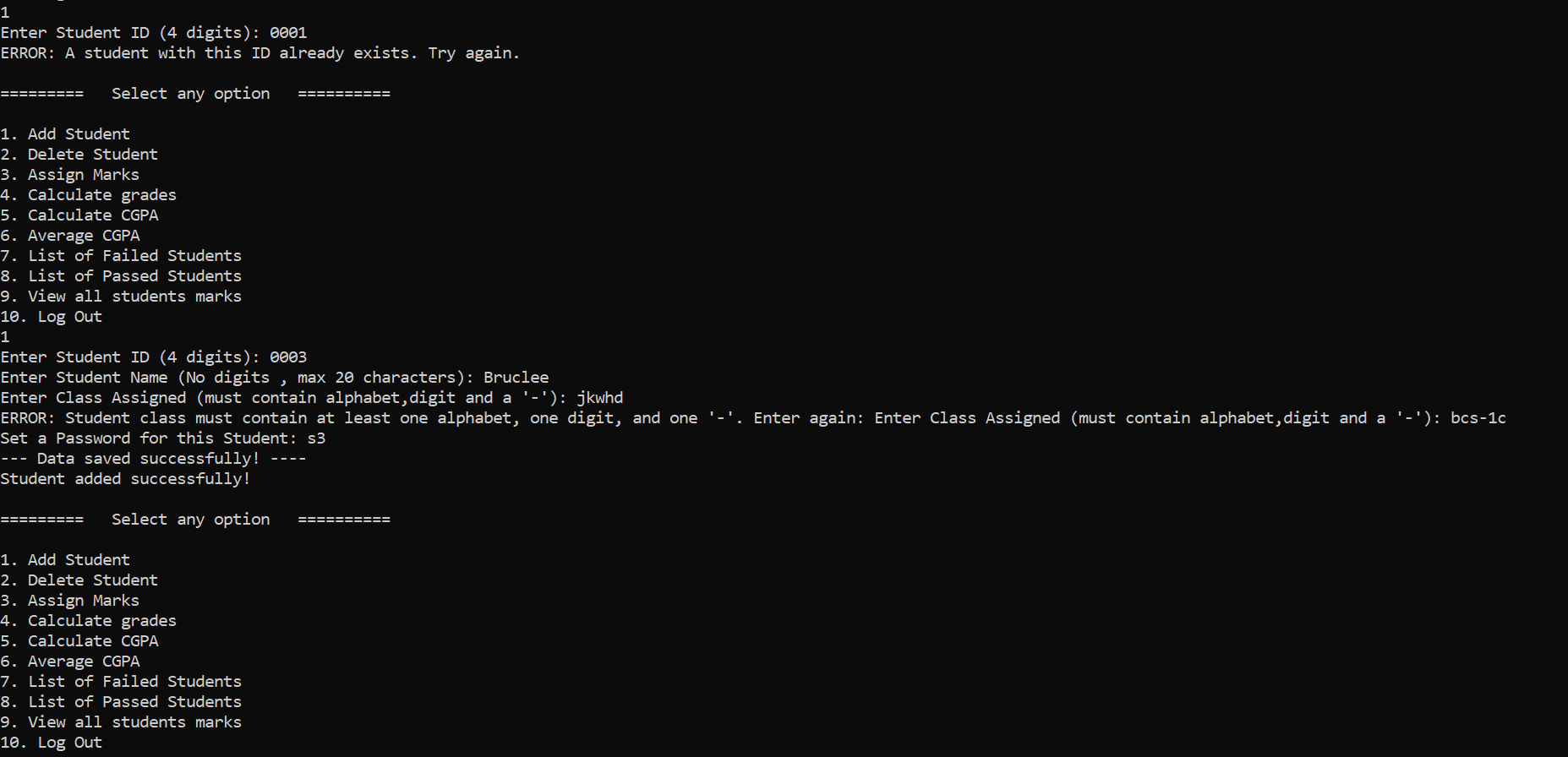
**Deleting a Teacher:**

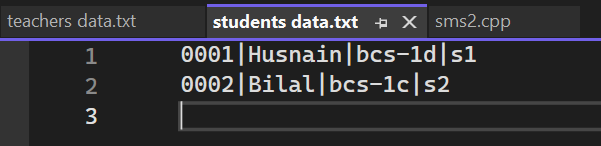


**Teacher LOGIN and MENU:**

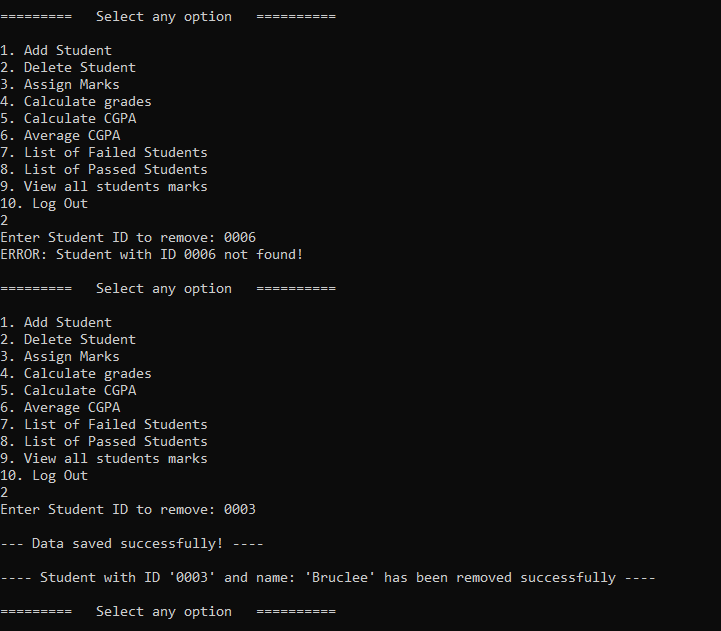


**Adding a Student:**

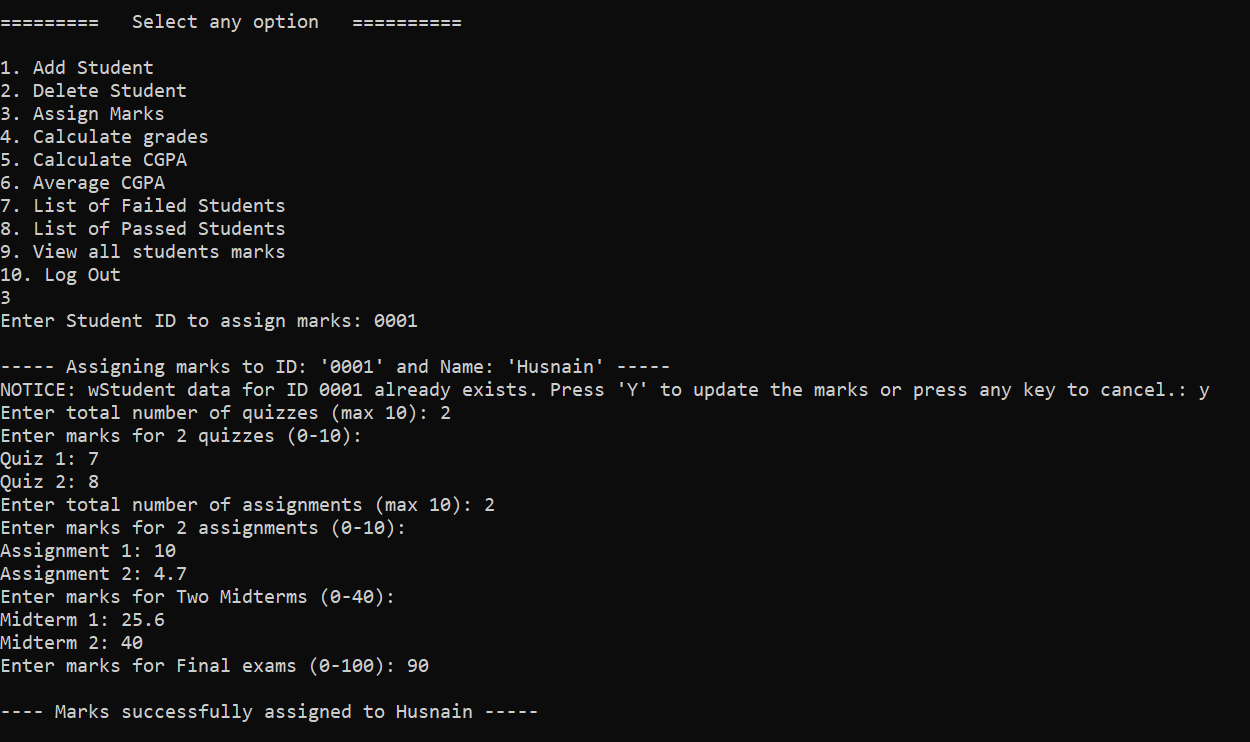


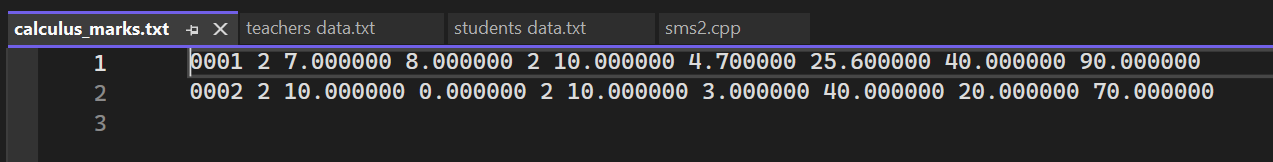


**Removing a Student:**

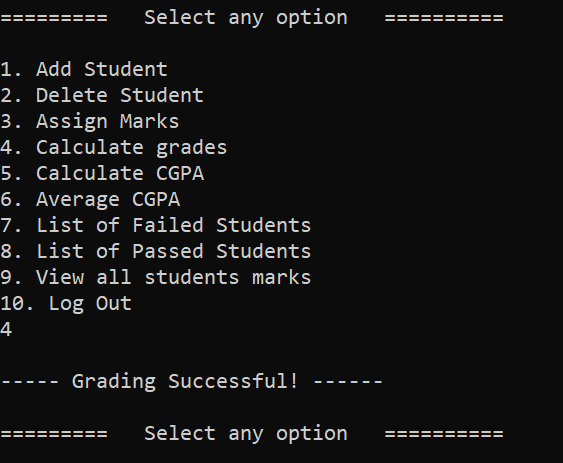


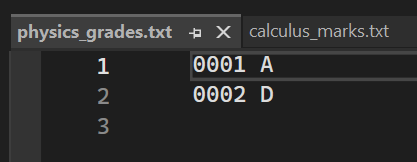
**Assigning Marks:**



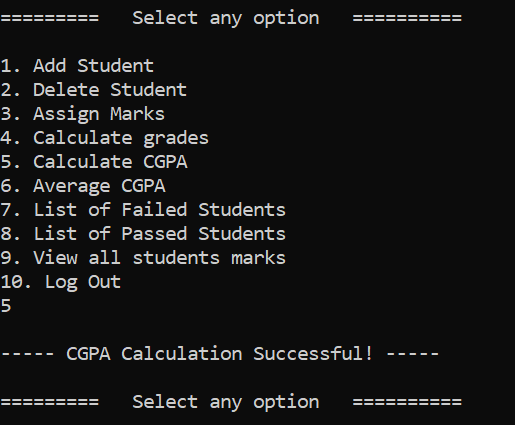
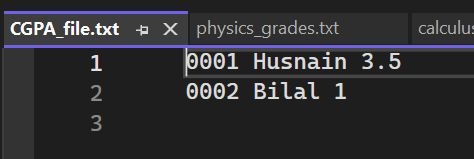


**Calculating Grades:**

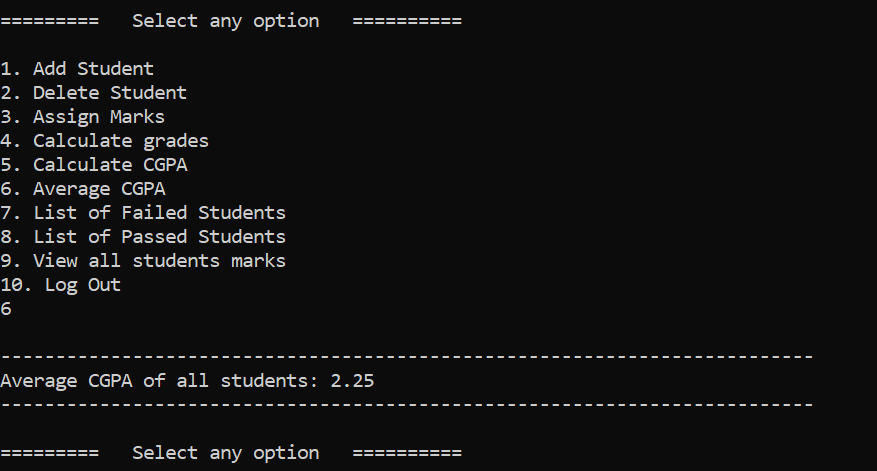




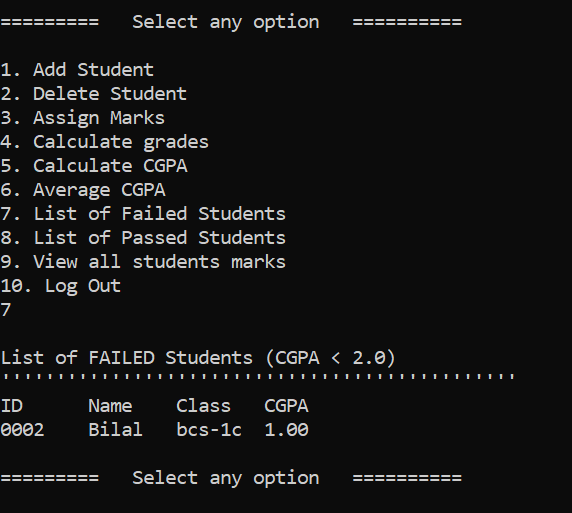
**Calculating CGPA:**



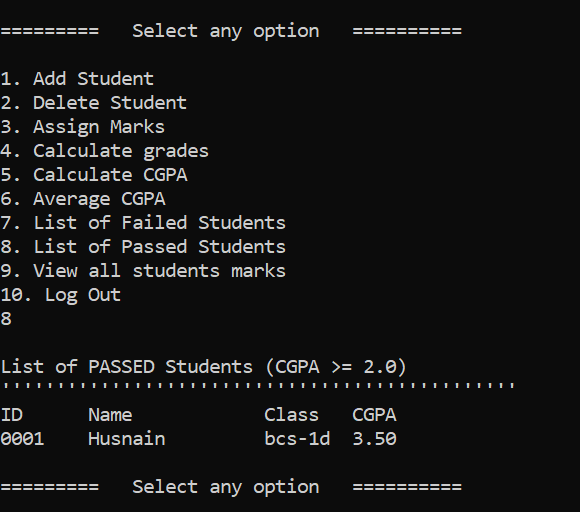
**Show Average CGPA:**



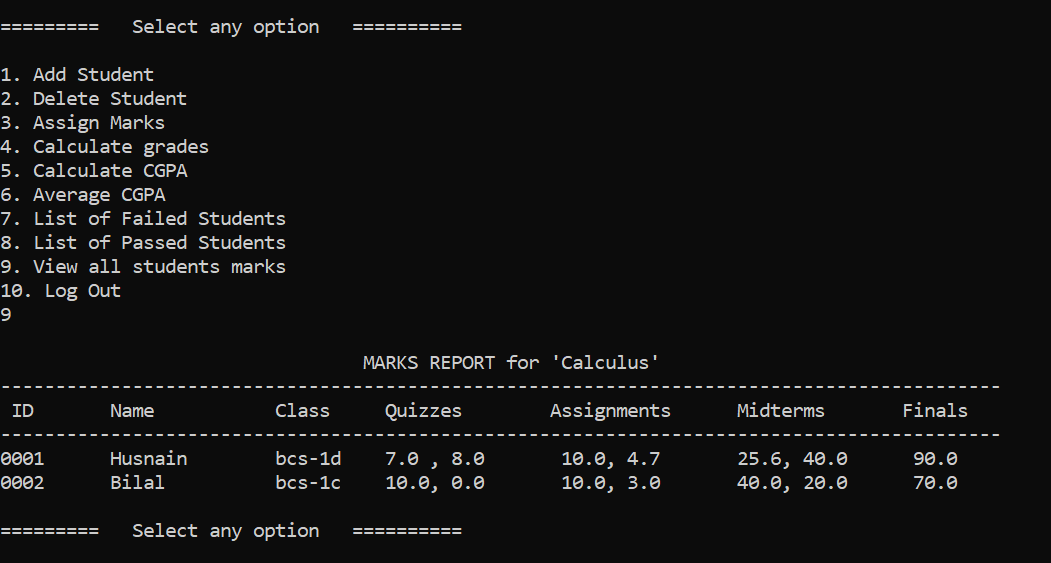
**List of Failed Students on the basis of CGPA:**



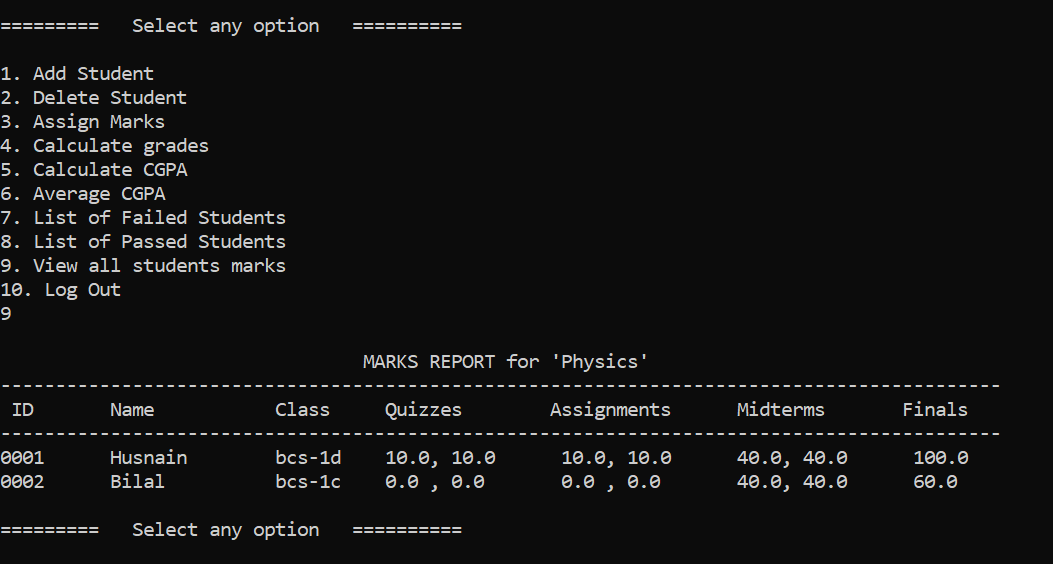
**List of Passed Students on the basis of CGPA:**



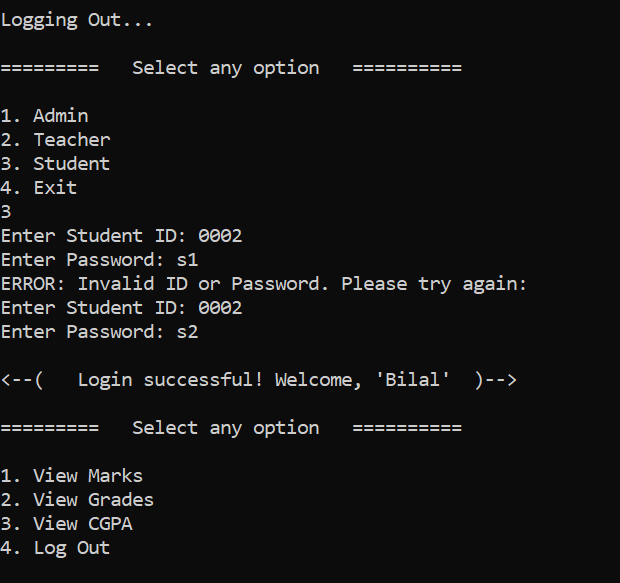
**Viewing Marks of All students of that Subject:**



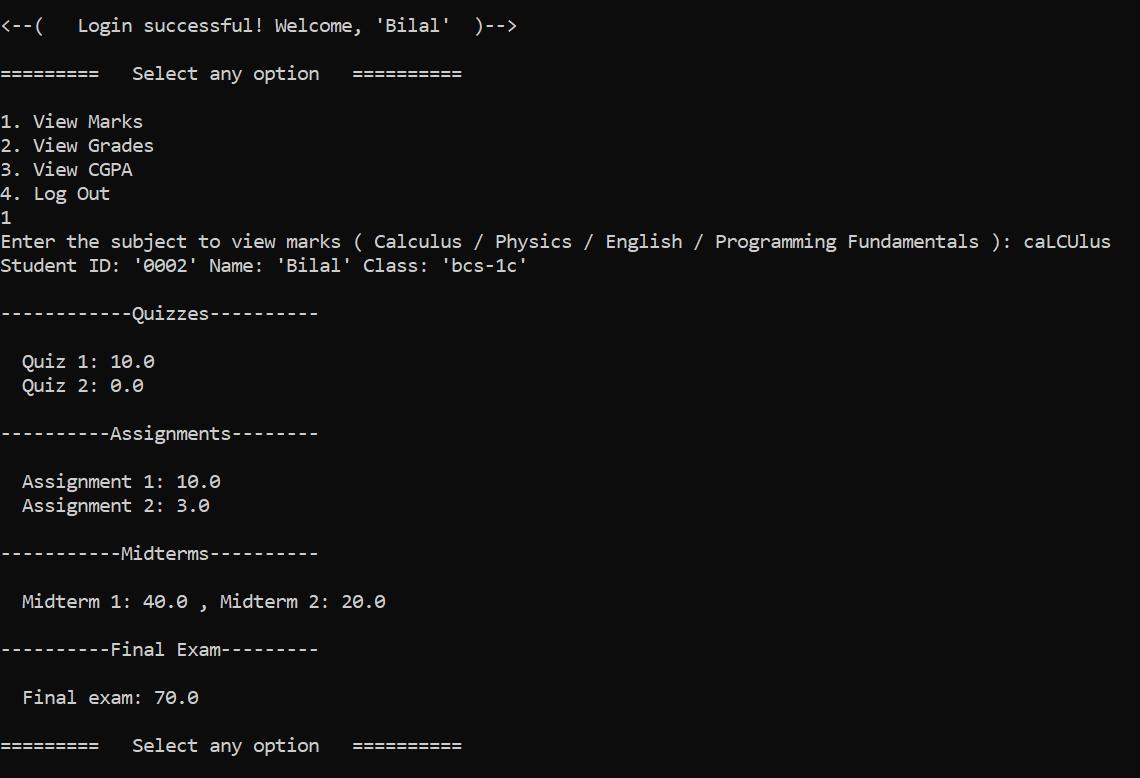
**When logged in by Physics teacher:**

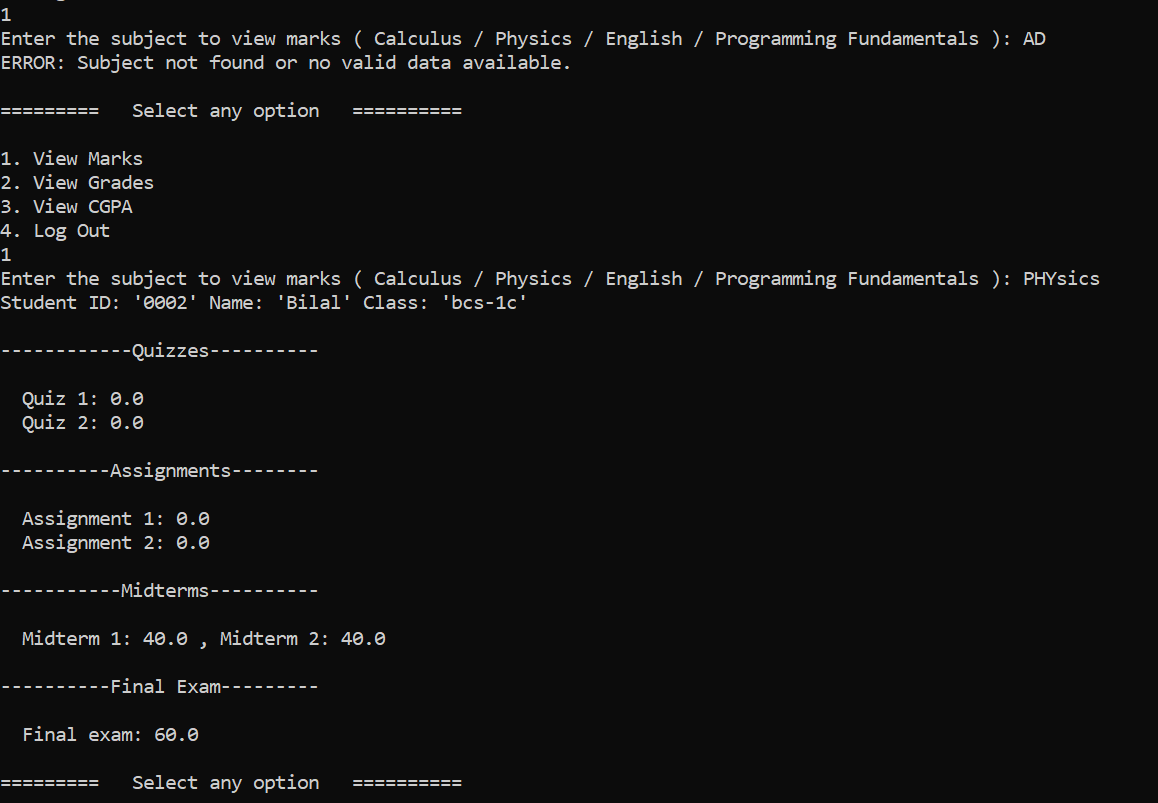


**Student LOGIN and MENU:**

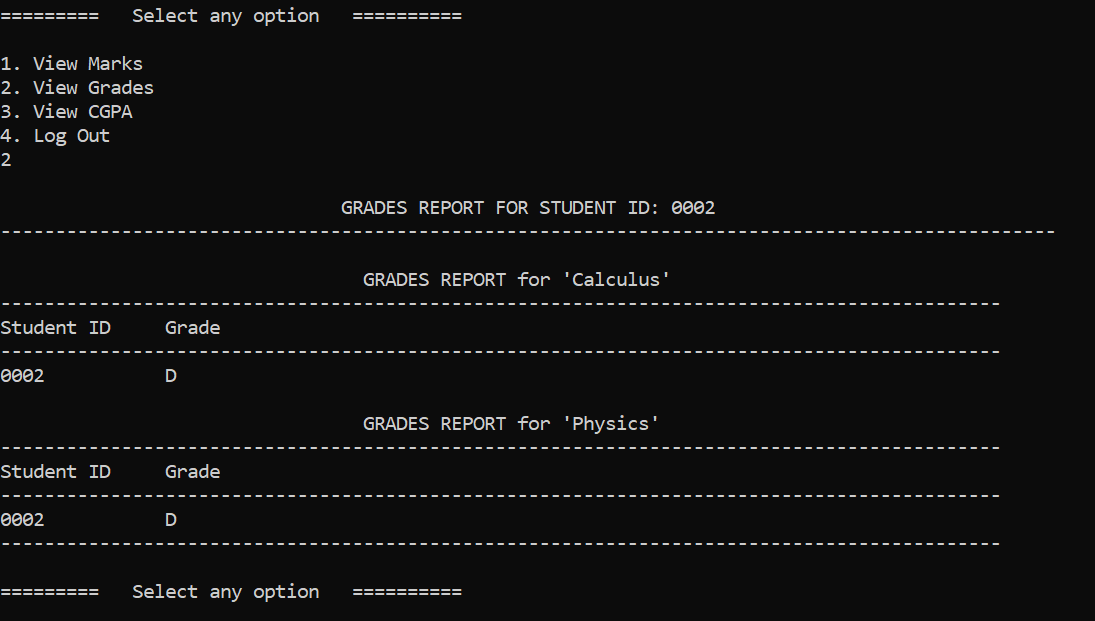


**Viewing Marks:**

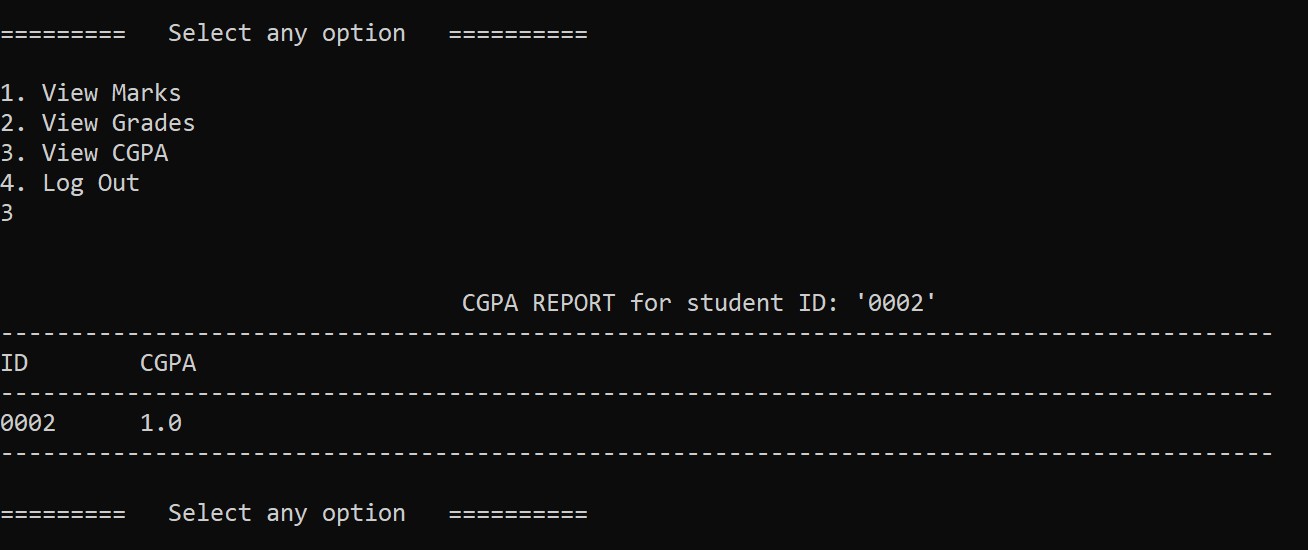




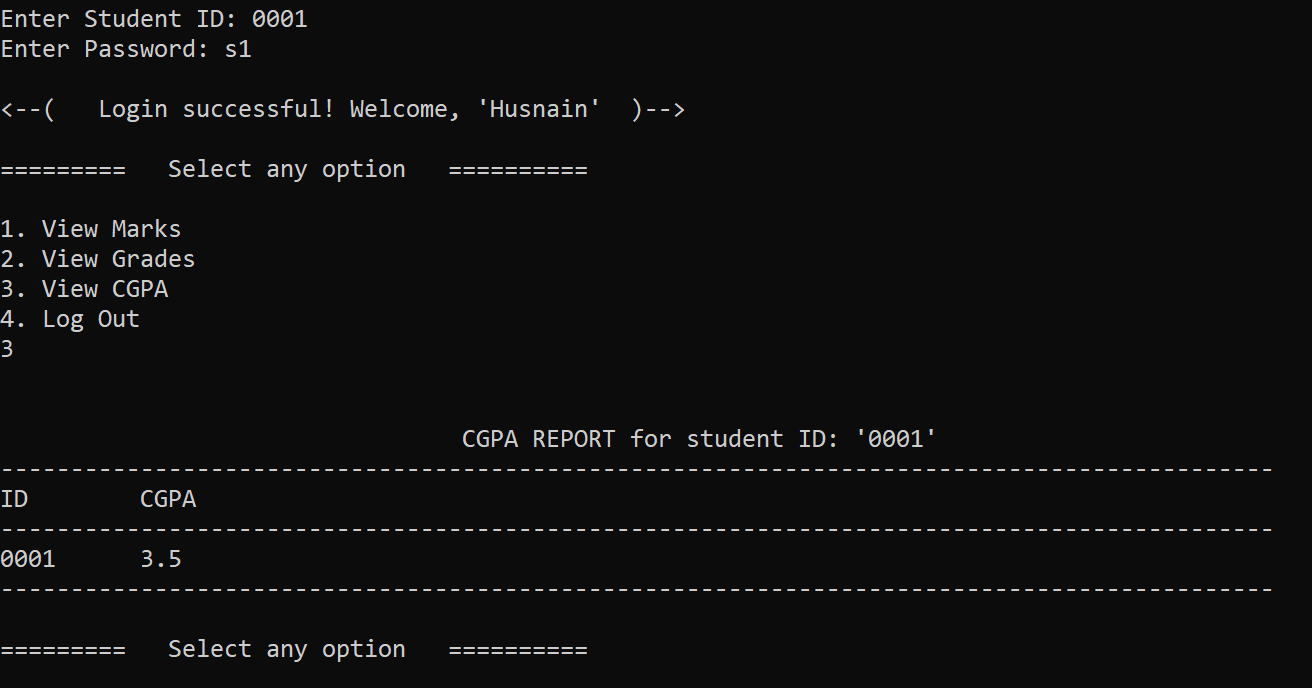
**Viewing Grades:**



**Viewing CGPA:**



**When logged in by other student:**



**END**