



HOME WORK 4

Sened Abraham Desalegn
Studid: 20082

Introduction

The GradeBookApp is a command-line interface program designed to assist instructors in managing student grades efficiently. It allows instructors to set up grading policies, add student records, and input and manage grades for assignments, tests, and final exams. The application supports sorting student information by name or ID and provides functionalities to calculate final grades based on predefined grading weights. All data, including student details and grading policies, are saved and managed through system files, making it a comprehensive tool for academic grading management.

Test Cases and Results

Test Case Structure:


- Each test case should include:
 - **Description:** A brief overview of what the test case is testing.
 - **Expected Outcome:** What should happen when the test case is executed.
 - **Actual Outcome:** The real result of executing the test case.
 - **Screenshots** showing the test case execution.


Documenting Test Cases:


Below I have created several test cases for the major functionality of the application. You may reproduce each test case as specified below.

Gradebook Instructor System

1. Test Case ID: TC001

 **Description:** Set up a new semester in the grade book system.

 **Preconditions:** The application is running.

 **Test Steps:**


Navigate to the main menu of the GradeBookApp.


Select the option to set up a new semester ('S').


Enter the number of programming assignments, tests, and final exams.


Enter the weight percentages for assignments, tests, and final exams.

Confirm the setup.

 **Expected Outcome:** The new semester setup is confirmed and saved in the system.

 **Actual Outcome:** The new semester setup is confirmed and saved in the system.

 **Status:** Success

 **Screenshots:**

```
===== MENU =====
(S) Set up new semester
(A) Add a student
(P) Record programming assignment
(T) Record test grades for all students
(F) Record Final exam score for all students
(C) Change a grade
(G) Calculate final score
(O) Output the grade data
(Q) Quit
Enter your choice: 
```

```
Enter your choice: s
Please enter the following info

Enter the number of programming assignments (0-6): 2
Enter the number of tests (0-4): 2
Enter the number of final exams (0-1): 1
Enter the weight of programming assignments (0-100): 40
Enter the weight of tests (0-100): 30
Enter the weight of final exams (0-100): 30
New semester setup completed.
===== MENU =====
```

The information is saved in the polic.dat file as shown:

```
policy.dat
1 {"num_assignments": 2, "num_tests": 2, "num_finals": 1, "weights": {"assignments": 40.0, "tests": 30.0, "final_exam": 30.0}}
```

For visibility:

```
policy.dat
1 {"num_assignments": 2, "num_tests": 2, "num_finals": 1,
2 "weights": {"assignments": 40.0, "tests": 30.0, "final_exam": 30.0}}
```

🚦 **Test Case ID:** TC002

🚦 **Description:** Add a new student to the grade book.

🚦 **Preconditions:** The semester has been set up.

🚦 **Test Steps:**

Navigate to the main menu of the GradeBookApp.

Select the option to add a student ('A').

Input the student's ID, first name, and last name.

Confirm the addition of the student.

🚦 **Expected Outcome:** The new student is added and displayed in the list of students.

🚦 **Actual Outcome:** The new student is added and displayed in the list of students.

🚦 **Status:** Success

```
===== MENU =====
(S) Set up new semester
(A) Add a student
(P) Record programming assignment
(T) Record test grades for all students
(F) Record Final exam score for all students
(C) Change a grade
(G) Calculate final score
(O) Output the grade data
(Q) Quit
Enter your choice: 
```

```
Enter your choice: a
Enter the student's ID (1 - 9999): 2345
Enter the student's last name (max 20 characters): abraham
Enter the student's first name (max 20 characters): sened
Student added successfully.
```

```
===== MENU =====
```

```
Enter your choice: a
Enter the student's ID (1 - 9999): 1234
Enter the student's last name (max 20 characters): berhe
Enter the student's first name (max 20 characters): hiyab
Student added successfully.
```

```
===== MENU =====
```

🚩 **Test Case ID:** TC003

🚩 **Description:** Record grades for a programming assignment.

🚩 **Preconditions:** At least one student has been added to the system.

🚩 **Test Steps:**

Navigate to the main menu of the GradeBookApp.

Select the option to record programming assignment grades ('P').

Enter the assignment number for which grades are to be recorded. (We had two assignments two option (1,2):

Input grades for each student.

Confirm the recording of grades.

🚩 **Highlighted point:** The maximum score for an assignment is calculated to ensure that the score entered for each student does not exceed the proportionate value that each assignment contributes towards the final grade, maintaining consistency across all assignments.

🚩 **Expected Outcome:** Grades are recorded and updated in the students' records.

🚩 **Actual Outcome:** Grades are recorded and updated in the students' records.

🚩 **Status:** Success

🚩 **Screenshots:** (I have only added one student earlier in Test Case TC002, for elaboration I will add another student following the same step as TC002):

```

===== MENU =====
(S) Set up new semester
(A) Add a student
(P) Record programming assignment
(T) Record test grades for all students
(F) Record Final exam score for all students
(C) Change a grade
(G) Calculate final score
(O) Output the grade data
(Q) Quit
Enter your choice:

```

```

Enter your choice: p
Enter the assignment number to record scores for: 1
Enter score for sened abraham (ID: 2345), maximum score is 20.0: 18
Enter score for hiyab berhe (ID: 1234), maximum score is 20.0: 15
Scores recorded successfully.
===== MENU =====

```

The info is also saved in “Grades.dat”

```

# grades.dat
1  ("stuid": 2345, "first_name": "sened", "last_name": "abraham", "grades": [{"Assignment": 18.0}], "final_score": 0.0)
2  ("stuid": 1234, "first_name": "hiyab", "last_name": "berhe", "grades": [{"Assignment": 15.0}], "final_score": 0.0)

```

More Clarified:

```

"last_name": "abraham", "grades": [{"Assignment": 18.0}],
"last_name": "berhe", "grades": [{"Assignment": 15.0}], "

```

Test Case ID: TC004

 **Description:** Record grades for a test.


 **Preconditions:** At least one student has been added to the system.

Test Steps:

Navigate to the main menu of the GradeBookApp.

Select the option to record test grades ('T').

Enter the test number for which grades are to be recorded. (We had two assignments two option (1,2)

 **Highlighted point:** The maximum score for an assignment is calculated to ensure that the score entered for each student does not exceed the proportionate value that each assignment contributes towards the final grade, maintaining consistency across all assignments.

Input grades for each student.

Confirm the recording of grades.

 **Expected Outcome:** Test grades are recorded and updated in the students' records.

- Actual Outcome: Test grades are recorded and updated in the students' records.
- Status: Success

```
===== MENU =====
(S) Set up new semester
(A) Add a student
(P) Record programming assignment
(T) Record test grades for all students
(F) Record Final exam score for all students
(C) Change a grade
(G) Calculate final score
(O) Output the grade data
(Q) Quit
Enter your choice: 
```

```
Enter your choice: t
Enter the test number to record scores for: 1
Enter score for sened abraham (ID: 2345), maximum score is 15.0: 10
Enter score for hiyab berhe (ID: 1234), maximum score is 15.0: 13
Test scores recorded successfully.
```

```
Enter your choice: t
Enter the test number to record scores for: 2
Enter score for sened abraham (ID: 2345), maximum score is 15.0: 15
Enter score for hiyab berhe (ID: 1234), maximum score is 15.0: 12
Test scores recorded successfully.
```

Then they are also updated in the “grades.dat”

```
# grades.dat
1 5, "first_name": "sened", "last_name": "abraham", "grades": [{"Assignment", 18.0}, {"Test", 10.0}, {"Test", 15.0}], "final"
2 4, "first_name": "hiyab", "last_name": "berhe", "grades": [{"Assignment", 15.0}, {"Test", 13.0}, {"Test", 12.0}], "final"
3
```

More clarified pic:

```
"": [{"Assignment", 18.0}, {"Test", 10.0}, {"Test", 15.0}], "final"
[["Assignment", 15.0}, {"Test", 13.0}, {"Test", 12.0}], "final"
```

- Test Case ID: TC005
- Description: Record grades for a Final Exam.
- Preconditions: At least one student has been added to the system.
- Test Steps:

Navigate to the main menu of the GradeBookApp.

Select the option to record test grades ('F').

Enter the test number for which grades are to be recorded. (We had two assignments 1 option (1)

- ✚ **Highlighted point:** The maximum score for an assignment is calculated to ensure that the score entered for each student does not exceed the proportionate value that each assignment contributes towards the final grade, maintaining consistency across all assignments.

Input grades for each student.

Confirm the recording of grades.

- ✚ **Expected Outcome:** Test grades are recorded and updated in the students' records.
- ✚ **Actual Outcome:** Test grades are recorded and updated in the students' records.
- ✚ **Status:** Success

```
===== MENU =====
(S) Set up new semester
(A) Add a student
(P) Record programming assignment
(T) Record test grades for all students
(F) Record Final exam score for all students
(C) Change a grade
(G) Calculate final score
(O) Output the grade data
(Q) Quit
Enter your choice: f
```

```
Enter your choice: f
Recording scores for the Final Exam:
Enter final exam score for sened abraham (ID: 2345), maximum score is 30.0: 26
Enter final exam score for hiyab berhe (ID: 1234), maximum score is 30.0: 24
Final exam scores recorded successfully.
```

Also saved to "grades.dat"

```
# grades.dat
1 med, "last_name": "abraham", "grades": [{"Assignment", 10.0}, {"Test", 10.0}, {"Test", 15.0}, {"FinalExam", 26.0}], "f
2 lyab, "last_name": "berhe", "grades": [{"Assignment", 15.0}, {"Test", 13.0}, {"Test", 12.0}, {"FinalExam", 24.0}], "f
3
```

More Clarified:

```
0.0}, {"Test", 15.0}, {"FinalExam", 26.0}],
], {"Test", 12.0}, {"FinalExam", 24.0}], "f
```

- ✚ **Test Case ID:** TC006
- ✚ **Description:** Change a student's grade.
- ✚ **Preconditions:** Grades have been recorded for assignments or tests.
- ✚ **Test Steps:**

Navigate to the main menu of the GradeBookApp.

Select the option to change a grade ('C').

Input the student's ID and the type of grade to change.

Enter the new grade and the specific assignment/test number.

Confirm the grade change.

🚦 **Expected Outcome:** The grade change is reflected in the student's record.

🚦 **Actual Outcome:** The grade change is reflected in the student's record.

🚦 **Status:** Success

--Testing to change Assignments

```
===== MENU =====
(S) Set up new semester
(A) Add a student
(P) Record programming assignment
(T) Record test grades for all students
(F) Record Final exam score for all students
(C) Change a grade
(G) Calculate final score
(O) Output the grade data
(Q) Quit
Enter your choice: 
```

```
Enter your choice: c
Enter the student ID for which the grade needs to be changed: 2345
Enter the type of grade to change (P for programming assignment, T for test, F for final exam): p
Enter the new score: 18.67
Enter the p number: 1
Grade updated successfully.
```

Successfully, updated in the file.

```
grades.dat
1  "first_name": "sened", "last_name": "abraham", "grades": [{"Assignment", 18.67}, {"Test", 10.0}],
2  "first_name": "hiyab", "last_name": "berhe", "grades": [{"Assignment", 15.0}, {"Test", 13.0}, {"Te
```

```
abraham", "grades": [{"Assignment", 18.67}, {"Test", 10.0}],
berhe", "grades": [{"Assignment", 15.0}, {"Test", 13.0}, {"Te
```

Testing to update Tests:


```

Enter your choice: c
Enter the student ID for which the grade needs to be changed: 2345
Enter the type of grade to change (P for programming assignment, T for test, F for final exam): t
Enter the new score: 6
Enter the t number: 1
Grade updated successfully.

```

Updates successfully in the file “grades.dat”

```

grades.dat
1  {"stuid": 2345, "first_name": "sened", "last_name": "abraham", "grades": [{"Assignment", 19.0}, {"Test", 6.0},
2  {"stuid": 1234, "first_name": "hiyab", "last_name": "berhe", "grades": [{"Assignment", 15.0}, {"Test", 8.0}, [
3  |

```

```

les": [{"Assignment", 18.67}, {"Test", 6.0}, {"Test", 15.0}, {"FinalExam"
s": [{"Assignment", 15.0}, {"Test", 13.0}, {"Test", 12.0}, {"FinalExam",

```

Testing to update Finals:

```

(Q) Quit
Enter your choice: c
Enter the student ID for which the grade needs to be changed: 2345
Enter the type of grade to change (P for programming assignment, T for test, F for final exam): f
Enter the new score: 29
Final exam grade updated successfully.
===== MENU =====

```

Updates successfully in the file “grades.dat”

```

grades.dat
"last_name": "abraham", "grades": [{"Assignment", 18.67}, {"Test", 6.0}, {"Test", 15.0}, {"FinalExam", 29.0}], "
"last_name": "berhe", "grades": [{"Assignment", 15.0}, {"Test", 13.0}, {"Test", 12.0}, {"FinalExam", 24.0}], "fi


```


```

grades.dat
1  ], {"Test", 6.0}, {"Test", 15.0}, {"FinalExam", 29.0}], "
2  [{"Test", 13.0}, {"Test", 12.0}, {"FinalExam", 24.0}], "fi
3  |

```

 **Test Case ID:** TC007

 **Description:** Calculate final scores for all students.


 **Preconditions:** All necessary grades (assignments, tests, final exam) have been entered.


 **Test Steps:**

Navigate to the main menu of the GradeBookApp.

Select the option to calculate final scores ('G').

Confirm the calculation.

 **Expected Outcome:** Final scores are calculated based on defined weights and updated in the students' records.

 **Actual Outcome:** Final scores are calculated based on defined weights and updated in the students' records.

Status: Success





```
===== MENU =====
(S) Set up new semester
(A) Add a student
(P) Record programming assignment
(T) Record test grades for all students
(F) Record Final exam score for all students
(C) Change a grade
(G) Calculate final score
(O) Output the grade data
(Q) Quit
Enter your choice: 
```

```
Enter your choice: g
Final scores calculated and updated for all students.
===== MENU =====
(S) Set up new semester
```

Updates successfully in the grades.dat file

```
grades.dat
1  ["Assignment", 18.67], ["Test", 6.0], ["Test", 15.0], ["FinalExam", 29.0]], "final_score": 87.34}
2  ["Assignment", 15.0], ["Test", 13.0], ["Test", 12.0], ["FinalExam", 24.0]], "final_score": 79.0}
3
```




```
, ["FinalExam", 29.0]], "final_score": 87.34}
["FinalExam", 24.0]], "final_score": 79.0}
```

-  **Test Case ID:** TC008
-  **Description:** Output the grade data.
-  **Preconditions:** Final scores have been calculated.
-  **Test Steps:**

 Navigate to the main menu of the GradeBookApp.

 Select the option to output grade data ('O').

 Displays output both on ID and By name based on the selected option.

-  **Expected Outcome:** Grade data is output in the desired format and displayed.
-  **Actual Outcome:** Grade data is outputted in the desired format and displayed.
-  **Status:** Success

```
===== MENU =====
(S) Set up new semester
(A) Add a student
(P) Record programming assignment
(T) Record test grades for all students
(F) Record Final exam score for all students
(C) Change a grade
(G) Calculate final score
(O) Output the grade data
(Q) Quit
Enter your choice: █
```

It sorts the Names based on the last names.

```
Enter your choice: o
Choose sorting method: Enter 'N' for name or 'I' for ID: N
-----Sorting by Name-----
StudentID: 2345
First_Name:sened
Last_Name:abraham
FinalExams:[Score:29.0]Tests:Score:6.0
Score:15.0
Assignments:Score:18.67
FinalScore:87.34

StudentID: 1234
First_Name:hiyab
Last_Name:berhe
FinalExams:[Score:24.0]Tests:Score:13.0
Score:12.0
Assignments:Score:15.0
FinalScore:79.0
```

Sorting by Id:

```
Enter your choice: o
Choose sorting method: Enter 'N' for name or 'I' for ID: i
-----Sorting By ID-----
StudentID: 1234
First_Name:hiyab
Last_Name:berhe
FinalExams:[Score:24.0]Tests:Score:13.0
Score:12.0
Assignments:Score:15.0
FinalScore:79.0

StudentID: 2345
First_Name:sened
Last_Name:abraham
FinalExams:[Score:29.0]Tests:Score:6.0
Score:15.0
Assignments:Score:18.67
FinalScore:87.34
```

- 🚦 **Test Case ID:** TC009
- 🚦 **Description:** Quit the application.
- 🚦 **Preconditions:** The application is running.
- 🚦 **Test Steps:**

Navigate to the main menu of the GradeBookApp.

Select the option to quit ('Q').

- 🚦 **Expected Outcome:** The application saves any changes and closes.
- 🚦 **Actual Outcome:** The application saves any changes and closes.
- 🚦 **Status:** Success

```
===== MENU =====
(S) Set up new semester
(A) Add a student
(P) Record programming assignment
(T) Record test grades for all students
(F) Record Final exam score for all students
(C) Change a grade
(G) Calculate final score
(O) Output the grade data
(Q) Quit
Enter your choice: █
```

```
(Q) QUIT
Enter your choice: q
Quitting ... 😊
PS C:\Users\sened\Desktop\hw4>
```

Some Important Cases Handled:

1. When there is no policy, no commands can be executed. They should be set first.

```
Enter your choice: a
Error: Please set up the semester first.
===== MENU =====
```

2. When the user enters the command “S”, all data in the files will be ignored. And the new policy will be included for one semester.
3. While running, is the instructor finished and want to set for another semester. They will receive prompt as follows below:

```
Enter your choice: s
Setup for the current semester has already been completed.
Do you want to reset the setup? (yes/no):
```

4. There are many validations to check if the data inputs are correct. For example, while adding a student the ID should be between 1-9999. If it is out of range, it throws exception through the error handling.

```
Enter your choice: a
Enter the student's ID (1 - 9999): 9999999
Error: Student ID must be in the range 1 - 9999.
===== MENU =====
```