Test Plan

Nelson Villatoro

CMSC115

Chapter 7, Project 1

December 03, 2024

**Program Goals & Objectives**

This program aims to read a list of student scores, determine the best score, and assign letter grades to each student based on their score relative to the best score. The program prompts the user to enter the total number of students and collects all their scores. It then computes the highest score among them. Using the logic of the program, grades are assigned as follows::

• **A**: score ≥ best - 10

• **B**: score ≥ best - 20

• **C**: score ≥ best - 30

• **D**: score ≥ best - 40

• **F**: if other conditions are not satisfied

**Program Functional Requirements**

1. User Input
   1. The program should prompt the user to enter the number of students (an integer).
   2. The program should prompt the user to enter the scores for each student.
2. Data Processing
   1. Store the scores in an integer array.
   2. Determine the best score among the entered scores.
   3. For each student’s score, calculate the grade based on the grading scheme:
      1. If the student’s score is greater than or equal to best - 10, assign grade ‘A’.
      2. ii. If the student’s score is greater than or equal to best - 20, assign grade ‘B’.
      3. iii. If the student’s score is greater than or equal to best - 30, assign grade ‘C’.
      4. iv. If the student’s score is greater than or equal to best - 40, assign grade ‘D’.
      5. v. If the student’s score does not meet any of the above conditions, assign grade ‘F’.
3. Output
   1. Display each student’s score and the corresponding letter grade in the format:

**Program Pseudocode**

START

Prompt user to enter number of students

Get number of students from user

Create array to store scores

Prompt user to enter all student scores

FOR each student position in array

Get score from user and store in array

ENDFOR

Set best score to first score in array

FOR each remaining score in array

IF current score is greater than best score

Update best score to current score

ENDIF

ENDFOR

FOR each student position in array

IF score >= best - 10

Assign grade 'A'

ELSE IF score >= best - 20

Assign grade 'B'

ELSE IF score >= best - 30

Assign grade 'C'

ELSE IF score >= best - 40

Assign grade 'D'

ELSE

Assign grade 'F'

ENDIF

Display student number, score, and grade

ENDFOR

Close input scanner

END

**Program Flowchart**

**I was unable to complete the flowchart.**

**Table 1 – Traceability Matrix**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test Case | Input/Output | Expected Result | Actual Result | Outcome  (Pass/Fail) |
| 1a | Enter number of students: 5  Enter 5 scores: 70  30  60  80  55 | Student 0 score is 70 and grade is A  Student 1 score is 30 and grade is F  Student 2 score is 60 and grade is B  Student 3 score is 80 and grade is A  Student 4 score is 55 and grade is C | Student 0 score is 70 and grade is A  Student 1 score is 30 and grade is F  Student 2 score is 60 and grade is B  Student 3 score is 80 and grade is A  Student 4 score is 55 and grade is C | Pass |
| 2a | Enter number of students: ten | Handle invalid input correctly | Exception in thread "main" java.util.InputMismatchException | Fail |
| 3a | Enter number of students: 0 | Handle invalid input correctly | OutOfBoundsException | Fail |