CST8116 Pre-Lab Exercise 04

Revised 20 Oct 2023, DHH

Decisions

# Learning Resources

* Review the PowerPoint Slides, and Microsoft Word documents in lecture notes Week 9.
* Complete the required reading in Week 9.
  + Joyce Farrell. 2018. Programming Logic & Design Comprehensive. 9th Ed. Cengage Learning. Chapter 4 pp. 125 to 166  
    - All Sections
  + Cay Horstmann. 2019. Big Java Early Objects. 7th Ed. Wiley. Chapter 5 pp. 131 to 170  
    - All Sections

# Learning Objectives

1. Practice Problem Solving with Decision Structures
2. Gain experience using Boolean expressions to make a decision
3. Gain experience testing boundary cases

**General Instruction:** You should complete the following tasks before your scheduled In-Lab Exercise 04

# Task 1 – Problem Description, Pseudocode and Decisions

* A school wants a software program that permits a course number to be entered, and a letter grade. The output from the program is the course number, the percent grade range, the letter grade, and the numeric grade. One of the courses at the school, ABC333, has a requirement that if a student obtains a grade of D it is treated as an F. The program should handle this situation when generating the output.
* The grading system is summarized here:

Percent Grade Letter Grade Numeric Grade

80 – 100 A 4.0

70 – 79 B 3.0

60 – 69 C 2.0

50 – 59 D 1.0

0 – 49 F 0.0

* A junior programmer at the company you are working for has provided some starter code, and UML Class diagrams you can use as a reference, however, method createReport of class ShoeSizeConverter is incomplete. Focus your efforts on this method. (See Appendix for diagrams)
  + Review the starter design and starter code provided and write pseudocode for the createReport method of class GradingSystem.
  + Your pseudocode must demonstrate both a case statement structure and an if-then-else-endif statement structure in the worker method createReport.
  + Ensure that you include a default output if none of the decisions equate to true – see sample program runs.
* Do not add any “extra” functionality to your solution such as repetition or error checking the input.

# Task 2 – Testing

* Test the program with each valid input value, and one invalid input of a letter grade.
* Substitute your name as it appears in ACSIS in “Program by <your name>”
* This is an example, you will have more test cases.

|  |  |  |  |
| --- | --- | --- | --- |
| External Input (user) | Expected Output | Actual Output | Description |
| “SomeCourse”  A | Please enter course number  Please enter letter grade  Course SomeCourse: Entered Grade A, Percent Grade 80 - 100, Number Grade 4.0  Program by <your name> | Please enter course number  Please enter letter grade  Course SomeCourse: Entered Grade A, Percent Grade 80 - 100, Number Grade 4.0  Program by Stanley Pieda | Matches, algorithm is not required to verify course number. |
| “ABC333”  D | Please enter course number  Please enter letter grade  Course ABC333: Entered Grade D, Actual Grade F, Percent Grade 0 - 49, Number Grade 0.0  Program by <your name> | Please enter course number  Please enter letter grade  Course ABC333: Entered Grade D, Actual Grade F, Percent Grade 0 - 49, Number Grade 0.0  Program by Stanley Pieda | Matches, algorithm handles case of ABC333. |

# Task 3 – Plan, Create, Compile, Run and Test a Java Program using Decisions

* Using Eclipse, create a project named PreLab Ex 04 within your CST8116Workspace folder. (See Hybrid 1)
* Copy the provided .java starter files into the project so that you can edit them in Eclipse.
* For **Program.java**, include a course standard file comment header with student full name present, as well as placing your student full name on the program output.
* For **GradingSystem.java**, follow Java programming conventions for identifiers, indentation, provide a course standard file comment header and programmer comments, and implement the worker method **createReport**, as detailed earlier in this document.
* Your implementation of your pseudocode must demonstrate the use of both a case statement structure and an if-then-else-endif statement structure in the worker method **createReport.**
* Take screen shots of your program running, minimally three screen shots based on your test plan:
  + one with an invalid input for a letter grade;
  + one with ABC333 with D, and;
  + one with any other course and any valid letter grade
* Ensure the Eclipse console window and your full name is visible as part of the program output in each screen shot.

**Note:** As specified in the course CSI, “Regardless of rubric, any lab exercise, lab exam, or lab assignment submitted without all required .java files will receive a maximum grade of 20%.”

# Grading

|  |  |  |  |
| --- | --- | --- | --- |
| Criteria | Missing / Incorrect (0) | Below Expectations (1) | Meets Expectations (3) |
| Algorithm: pseudocode | Missing or incorrect. | Pseudocode is included in MS Word document, however there are logic mistakes and or missing either if or switch structure(s). | Pseudocode is included in MS Word document, provides a working algorithm that meets the requirements. |
| Test Plan for Program | Missing or poorly done or is only an unchanged copy of the provided algorithm test plan. | May not have correct format, does not verify that the program outputs match expectations. | Has correct format as shown in the lab handout, verifies that the program outputs match, and documents variations in output including samples of invalid inputs. |

|  |  |  |  |
| --- | --- | --- | --- |
| Criteria | Missing / Incorrect (0) | Below Expectations (1) | Meets Expectations (2) |
| Screen Shot of Program Execution in Eclipse Console Window | Screen shot in Eclipse missing Console Window or incorrect or student name not part of program output. | Screen shot in Eclipse Console Window shows student full name as in ACSIS, but program might not have compiled or run properly. | Screen shot in Eclipse Console Window shows student full name as in ACSIS. Program runs properly and valid input(s) and expected output(s) are shown. |
| Source Code: \*.java file(s) Comments and Conventions | Missing or poorly done. | Missing a comment-header from one or more of class declaration and / or method main declaration. Code loosely follows Java coding conventions for identifiers, indentation. | File comment header with student full name is present. Class and method declarations have comment headers. Code closely follows Java coding conventions for identifiers, indentation. |
| Source Code:  \*.java file(s) program structure and logic. | Missing or poorly done or program does not follow from the pseudocode. | Program may have small syntax mistakes and will not compile, and / or produces incorrect output when run. Program loosely follows the student’s pseudocode and flowchart(s). | Program has correct syntax and program logic that produces correct output. Program closely follows the student’s pseudocode. |

# Appendix: Sample Program Run(s) (Each block of output is a separate program run)

* User input is indicated using **bold font with yellow highlighting**.

Please enter course number **tuna**

Please enter letter grade **fish**

Invalid letter grade entered

Program by Stanley Pieda

Please enter course number **ABC333**

Please enter letter grade **tuna**

Invalid letter grade entered

Program by Stanley Pieda

Please enter course number **12345**

Please enter letter grade **A**

Course 12345: Entered Grade A, Percent Grade 80 - 100, Number Grade 4.0

Program by Stanley Pieda

Please enter course number **ABC333**

Please enter letter grade **D**

Course ABC333: Entered Grade D, Actual Grade F, Percent Grade 0 - 49, Number Grade 0.0

Program by Stanley Pieda

Please enter course number **ABC333**

Please enter letter grade **A**

Course ABC333: Entered Grade A, Percent Grade 80 - 100, Number Grade 4.0

Program by Stanley Pieda

Please enter course number **aaa**

Please enter letter grade **A**

Course aaa: Entered Grade A, Percent Grade 80 - 100, Number Grade 4.0

Program by Stanley Pieda

Please enter course number **aaa**

Please enter letter grade **B**

Course aaa: Entered Grade B, Percent Grade 70 - 79, Number Grade 3.0

Program by Stanley Pieda

Please enter course number **aaa**

Please enter letter grade **C**

Course aaa: Entered Grade C, Percent Grade 60 - 69, Number Grade 2.0

Program by Stanley Pieda

Please enter course number **aaa**

Please enter letter grade **D**

Course aaa: Entered Grade D, Percent Grade 50 - 59, Number Grade 1.0

Program by Stanley Pieda

Please enter course number **aaa**

Please enter letter grade **F**

Course aaa: Entered Grade F, Percent Grade 0 - 49, Number Grade 0.0

Program by Stanley Pieda

Please enter course number **ABC333**

Please enter letter grade **F**

Course ABC333: Entered Grade F, Percent Grade 0 - 49, Number Grade 0.0

Program by Stanley Pieda

Please enter course number **aaa**

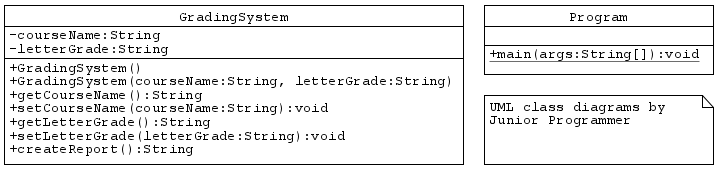
Please enter letter grade **a**

Invalid letter grade entered

Program by Stanley Pieda

# Appendix: Starter Diagrams

## UML Class Diagram (via UMLet)



## Flowchart (via Open Office Draw)

