**GE02 Design Solution with Conditions**

**Purpose**: Learn how to approach problem solving to design a solution containing conditions.

Effort: Collaborate ([Academic Integrity](https://docs.google.com/document/d/1vfCpxFGn-7Ru5fplmD7lK1n_cSFvhtMtsVQg4SCs2OU/edit?usp=sharing)) if you have time

**Points:** 20 (See Rubric)

**Deliverables:** Upload this document as word or pdf

# Instructions

Read slides 1-10 on  [Problem Solving and the Software Development Life Cycle](https://docs.google.com/presentation/d/e/2PACX-1vSC4tM0BsOaYjRmd_emNeHtKrWaExKXfvvLxQC10rCt77CPT_WvF_s3W--o_yuj8XczYGo7qHxgjMIq/pub?start=false&loop=false&delayms=3000&slide=id.p1). **Note:** You are not writing any java code for this exploration but instead you are starting to problem solve by writing pseudocode.

# Problem Description

| **Requirements:**  Professors need a program to calculate a student’s final grade using the category weights. | **Acceptance Criteria:**   * Ask the professor for the final grade for each category. (Can contain decimal values) * If a category grade is not a value from 0 to 100 then give an error message and the program will end * If valid grades are entered for the category   + Calculate final grade percentage   + Displays final percentage for student grade   + Display final letter grade |
| --- | --- |

From the syllabus each category has a weighted percentage.

| **Grading Weights**   | Guided Exploration | 15% | | --- | --- | | Homework | 35% | | Quizzes\* | 10% | | Mid-term\* | 20% | | Final Exam \* | 20% | | **Grading Scale:**     | **A** | **90+** | | --- | --- | | **B** | **80 or more but less than 90** | | **C** | **70 or more but less than 80** | | **D** | **60 or more but less than 70** | | **F** | **Less than 60** | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |

## Part 1: Analysis

1.1 Look at the problem and break it into 3 tasks.

| Ask user for grade inputs, decide whether or not the input grades are valid, calculate grade using category weights |
| --- |

1.2 Why do you think it is important to break a problem into smaller pieces?

| breaking a problem down makes a task easier to digest and less daunting. |
| --- |

**1.3 Test Cases:** From the tasks below fill in the possible inputs and expected outputs.

Task 1: Get final grade for each category and if not in valid range end program.

| **Input** | **Output** |
| --- | --- |
| Enter grade for GE: 105 | 105 is not a valid GE grade. Program ending. |
| Enter grade for GE: 100  Enter grade for HW: 93  Enter grade for Quiz: -1 | -1 is not a valid Quiz grade. Program ending. |
| Enter grade for HW: -73 | -73 is not a valid Homework grade. Program ending. |
| Enter grade for GE: 100  Enter grade for HW: 93  Enter grade for Quiz: 80.5  Enter grade for Midterm: 75.7  Enter grade for Final Exam: 90.3 | Grades were entered correctly. The final grade will be calculated |

Task 2: Calculate final grade percentage given each category grade.

Final Grade = (ge \*.15) + (hw\*.35) + (qu\*.1) +(mid \* .2) + (fin \*.2)

| **Input** | **Output** |
| --- | --- |
| GE: 100  HW: 93  Quiz: 80.5  Midterm: 75.7  Final Exam: 90.3 | 88.8 |
| List another example of valid grades that could be entered for each category.  GE: 70  HW: 80  Quiz: 80  Midterm: 90  Final Exam: 50 | 74.5 |
| List another example of valid grades that could be entered for each category.  GE: 90  HW: 70  Quiz: 80  Midterm: 60  Final Exam: 90 | 76 |

Task 3: Display the correct letter grade for the final percentage.

| **Input** | **Output** |
| --- | --- |
| 90% | A |
| 69.9% | D |
| 85.4 | B |

1.4 Why do you think it is important to have an understanding of the inputs and output for each task (Test cases)?

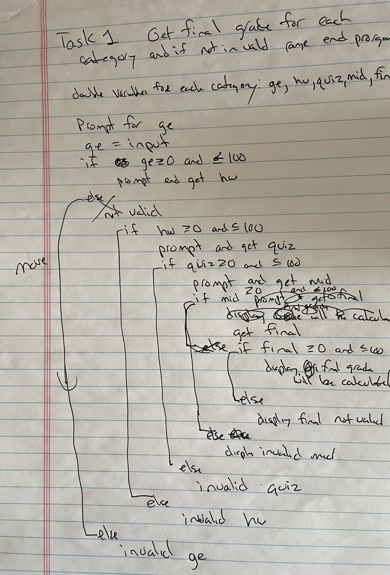
| It is important to understand the inputs and outputs of the tasks your code can perform so you can know if it is working properly or having issues later down the road. |
| --- |

## Part 2 Designing a Solution

Here you will create algorithms for each task by **writing the pseudocode to solve the problem for tasks 2 and 3 below on paper (task 1 is done for you)**. The pseudocode is not actual code but should still contain steps and conditions if/else. Include constants and variables you think will be needed for each task and their datatype. Take a picture or scan your design on paper and paste below. Review lecture 4 and chapter 3 for ideas on how to write the conditions.

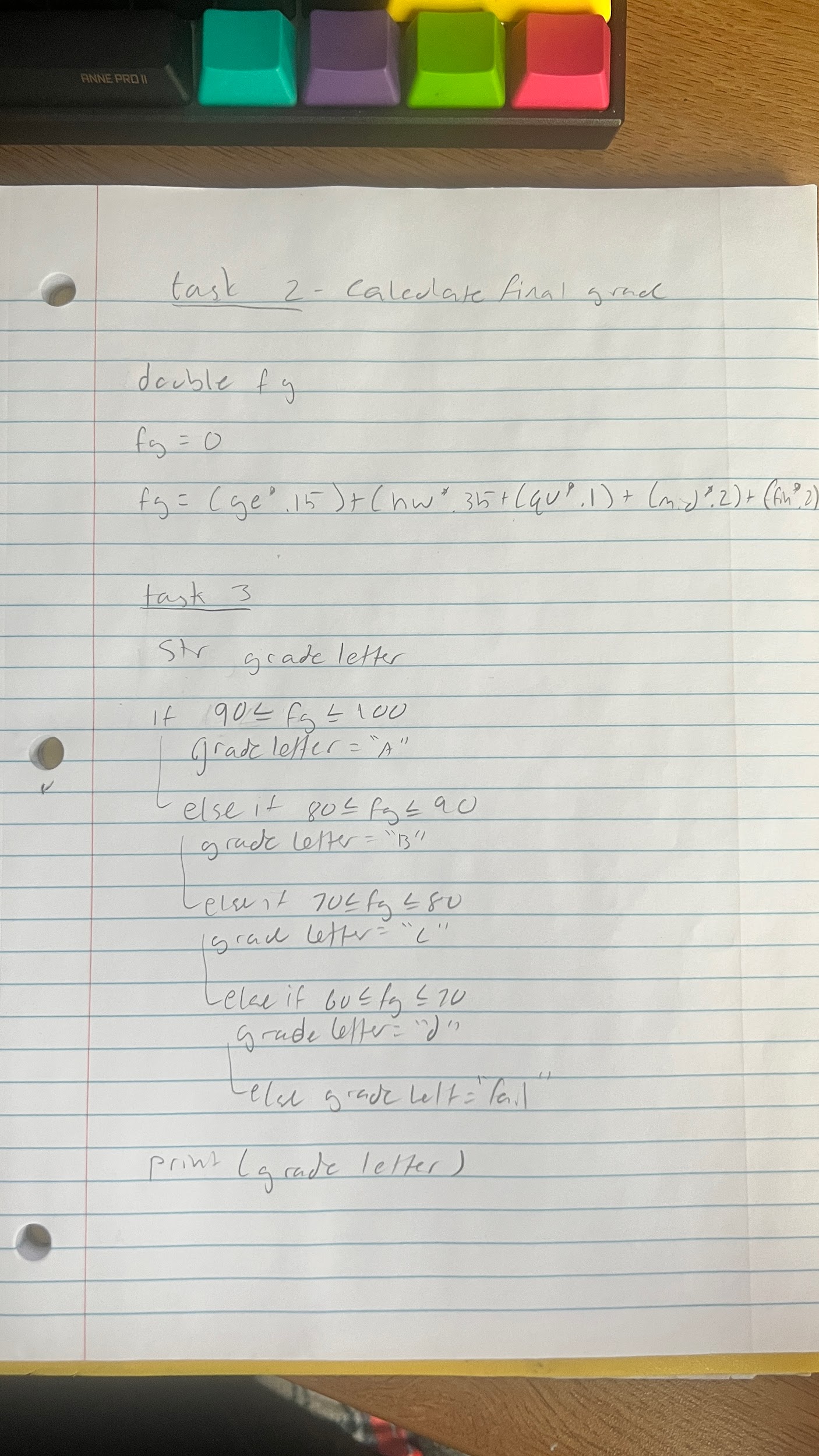
Task 1: Get final grade for each category and if not in valid range end program.

* As I think about the problem I see that a decision must be made for each grade inputted.
* After reviewing lecture 4 think nested if statements should be used so the program ends as soon as invalid data is entered for one of the grade categories.
* I indented to make the conditions more readable and drew lines to help me.
* Notice I cross things out and draw lines to help me. No need to rewrite if you make changes as this is your problem solving brainstorming stage.



2.1 Task 2: Calculate final grade percentage given each category grade. Think about what information would be best stored as a const.

Put image here of your pseudocode

* 

2.2 Task 3: Display the correct letter grade for the final percentage.

Put image here of your pseudocode

2.3 What is pseudocode?

| Pseudocode is a mixture of english and code that acts as a simpler, easier to quickly write version of your program. |
| --- |

2.4 How can writing pseudocode help solve a problem?

| Pseudocode can make it easier for you to understand the code you are writing. |
| --- |