Design Document and Test Plan

Name of team members who collaborated on the design and test plan:

1. Name (*first last*): Dylan Harvey
2. Name (*first last*): Janaki Bhosale
3. Name (*first last*): Angie Diaz

Name of programming challenge for which you submit this document: Problem 3: Retirement Calculator

# Pseudocode

RS = P \* (1 + r)t

Prompt user to enter P

Prompt user to enter r as a decimal

Prompt user to enter t

Call function to calculate savings using formula (RS = P x (1 + r)t aka rs = p \* pow((1 + r), t))

Display output after calculation. (Your retirement savings will be $\_)

# Test Plan

*(See Ch. 5.13 in our textbook for an example of how to write a test plan)*

|  |  |  |  |
| --- | --- | --- | --- |
| **Test #** | **Purpose** | **Input** | **Expected Output** |
| 1 | Test correct input | 1000 0.025  20 | Your retirement savings will be $1638.62 |
| 2 | Test correct input | 1000  0.025  40 | Your retirement savings will be $2685.06 |
| 3 | Test correct input | 1500  0.035  25 | Your retirement savings will be $3544.86 |
| 4 | Test correct input | 2000  0.015  40 | Your retirement savings will be $3628.03 |
| 5 | Test correct input | 1000  0.100  20 | Your retirement savings will be $6727.50 |
| … | *(Feel free to add more test cases)* |  |  |

Note: since this one in particular is not input validated, these tests only assume correct input.