- If you made assumptions about the values of parameters or you know that your function won't work with particular values, write a precondition to warn other programmers.

3.11 Exercises

Here are some exercises for you to try on your own. Solutions are available at http://pragprog.com/titles/gwpy2/practical-programming.

1. Two of Python's built-in functions are min and max. In the Python shell, execute the following function calls:

   a. `min(2, 3, 4)`
   b. `max(2, -3, 4, 7, -5)`
   c. `max(2, -3, min(4, 7), -5)`

2. For the following function calls, in what order are the subexpressions evaluated?

   a. `min(max(3, 4), abs(-5))`
   b. `abs(min(4, 6, max(2, 8)))`
   c. `round(max(5.572, 3.258), abs(-2))`

3. Following the function design recipe, define a function that has one parameter, a number, and returns that number tripled.

4. Following the function design recipe, define a function that has two parameters, both of which are numbers, and returns the absolute value of the difference of the two. Hint: Call built-in function abs.

5. Following the function design recipe, define a function that has one parameter, a distance in kilometers, and returns the distance in miles. (There are 1.6 kilometers per mile.)

6. Following the function design recipe, define a function that has three parameters, grades between 0 and 100 inclusive, and returns the average of those grades.

7. Following the function design recipe, define a function that has four parameters, all of them grades between 0 and 100 inclusive, and returns the average of the *best 3* of those grades. Hint: Call the function that you defined in the previous exercise.

8. Complete the examples in the docstring and then write the body of the following function: