Problem Set #7

MACS 30000, Dr. Evans
Keertana V. Chidambaram
(UCID: 12211266)

Problem 1: Unit Testing

(i) The edge cases for a general function that calculates the smallest factorial is `n` = 1, `n` = a non-prime number and `n` = a prime number. In this case, the error is caused because the range of values to be iterated on is calculated wrongly. This is because the upper bound given to the range function is not inclusive. For this particular case, the function would fail to produce the right answer when the square root of the number is its smallest factor. That is, when `n` is a square of a prime number. That test is also included in the test file and changes have been made to correct the upper bound so that the file passes the test.

(ii) To ensure complete coverage of the `month_len.py` file, cases have to be tested for every conditional statements. For the first if statement, one of `September, April, June,` or `November` has to be passed for the variable `month` for testing. For the second, `month` = `January, March, May, July, August, October,` or `December` has to be passed as argument. For the next set of 2 nested conditionals, `month` = `February` has to be passed through two tests, one with `leap_year` = `True` and the other with `leap_year` = `False`. Finally to check if `None` is correctly returned, an argument that is not the string representation of a month (the standard `foo` has been passed as a placeholder name) has to be passed for the variable `month`. Each of these cases have been checked in the test file, and 100% coverage has been achieved.

(iii) Four conditional statements can be covered by passing each of the four operands (`+, -, *, /`) as string in four different test cases. Additionally, we also need to check if appropriate error messages are generated. The first error message is generated at the first conditional statement. To invoke this error message we need to pass a non-string value for `oper`. The second and final error message is nested in the division part of the function. This part of the code can be covered by passing `b` = 0 and `oper` = `/`, i.e. trying to divide a number by zero. These are all the cases that have been used in the test file and 100% of the codes have been covered while testing.