

Jess Conway
13 February 2019
COMP 474 – Stone
Homework 1.1 – Software Testing, Part 1

a)

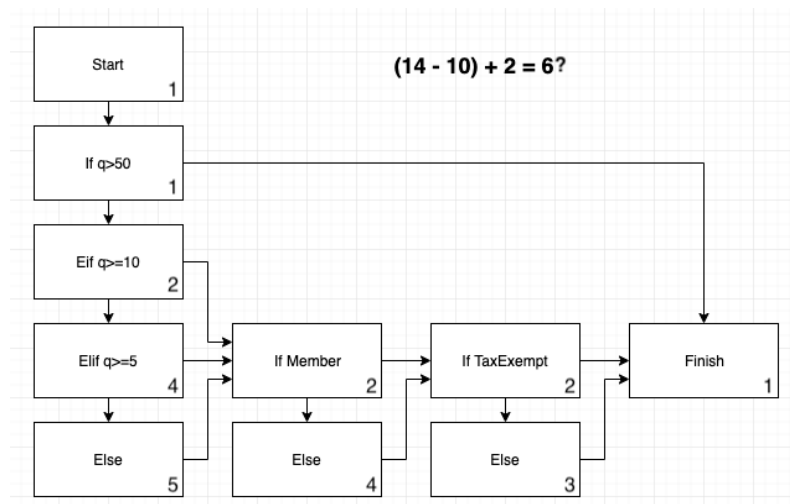
For full coverage, we need tests covering the following parameters:

1. Prices p such that $p > 0$
2. Quantities q such that $q > 50$, $50 \geq q > 10$, $10 \geq q > 5$, and $5 \geq q$
3. DiscountClub values dc such that $dc = \text{true}$, $dc = \text{false}$
4. TaxExempt values te such that $te = \text{true}$, $te = \text{false}$

If we wanted to do this the most efficient way possible, we would only need 6 tests (see flow graph for calculation). However, in practice it appears that 5 tests may be sufficient (the number in each box below corresponds to the number of the test that covers that path). This indicates an error in the graph or missed coverage in the tests below.

Define the checkout method as follows:

- IN: checkout(double price, int quantity, boolean discountMember, boolean taxExempt)
- OUT: netPurchase, netDiscount, taxAmount, netTotal



Sufficient tests for part (a) would be:

Test Number	Test Definition	Expected Output
1	checkout(83.10, 51, true, true)	TooManyItemsError
2	checkout(83.10, 35, true, true)	83.10, 15.79, 0, 67.31
3	checkout(83.10, 35, true, false)	83.10, 15.79, 3.03, 70.34
4	checkout(83.10, 7, false, true)	83.10, 4.16, 0, 78.94
5	checkout(83.10, 3, false, false)	83.10, 0, 3.74, 86.84

b)

We should also consider the following parameters for boundary analysis:

1. Quantities q=51, q=50, q=49, q=11, q=10, q=9, q=6, q=5, q=4, q=1

Test Number	Test Definition	Expected Output
6	checkout(83.10, 51, false, true)	TooManyItemsError
7	checkout(83.10, 50, false, true)	83.10, 8.31, 0, 74.79
8	checkout(83.10, 11, false, true)	83.10, 8.31, 0, 74.79
9	checkout(83.10, 10, false, true)	83.10, 8.31, 0, 74.79
10	checkout(83.10, 9, false, true)	83.10, 4.15, 0, 78.95
11	checkout(83.10, 6, false, true)	83.10, 4.15, 0, 78.95
12	checkout(83.10, 5, false, true)	83.10, 4.15, 0, 78.95
13	checkout(83.10, 4, false, true)	83.10, 0, 0, 83.10
14	checkout(83.10, 1, false, true)	83.10, 0, 0, 83.10

*Note that the price, membership, or tax parameters could be changed for any of these tests. This is because these test boundary values for quantity. None of these tests are concerned about price, membership or tax status.

c)

In order to check if the rounding method works correctly, we need to check the following:

1. Check that the outputs have exactly two decimals
2. Check that the decimal value is correctly rounded

In addition to this, we should check the following things during BVA and to examine edge cases:

1. Check values where third decimal of input is "5"
2. Check values where third decimal of input is "0"
3. Check values where input has less than 3 decimal places

Define the roundOff method as follows:

- IN: roundOff(double rounded)
- OUT: roundedNumberToTwoDecimals

Sufficient tests to cover part (c) include:

Test Number	Test Definition	Expected Output
15	roundOff(123.4567)	123.46
16	roundOff(98765.4321)	98765.43
17	roundOff(555.5555)	555.56
18	roundOff(10.004)	10.00
19	roundOff(5.52)	5.52
20	roundOff(9.4)	9.40
21	roundOff(7)	7.00

**See that tests 15-16 are general tests for coverage. Meanwhile, tests 17-18 check boundary cases and tests 19-21 test edge cases.