Homework 1

Below are four faulty programs. Each includes test inputs that result in failure. Answer the following questions about each program.

```
'''
Find last index of element

@param x array to search
@param y value to look for
@return last index of y in x; -1 if absent
@TypeError if x is None or …
'''
def find_last(x, y):
    i = len(x) - 1
    while i > 0:
        if x[i] == y:
            return i
        i -= 1
    return -1
# test: x = [2, 3, 5]; y = 2; Expected = 0
```

```
'''
Find last index of zero

@param x array to search

@return last index of 0 in x; -1 if absent
@TypeError if x is None or …
'''
def last_zero(x):
    i = 0
    while i < len(x):
        if x[i] == 0:
            return i
        i += 1
    return -1
# test: x = [0, 1, 0]; Expected = 2
```

```
'''
Count positive elements

@param x array to search
@return count of positive elements in x
@ TypeError if x is None or …
'''
def count_positive(x):
    count = 0
    i = 0
    while i < len(x):
        if x[i] >= 0:
            count += 1
        i += 1
    return count
# test: x = [-4, 2, 0, 2]; Expcted = 2
```

```
'''
Count odd or postive elements

@param x array to search
@return count of odd/positive values in x
@ TypeError if x is None or …
'''
def odd_or_pos(x):
    count = 0
    i = 0
    while i < len(x):
        if x[i] > 0 or x[i] % 2 == -1:
            count += 1
        i += 1
    return count
# test: x = [-3, -2, 0, 1, 4]; Expected = 3
```

(a) Explain what is wrong with the given code. Describe the fault precisely by proposing a modification to the code.
(b) If possible, give a test case that does not execute the fault. If not, briefly explain why not. (You need to give the same number of arguments.)
(c) If possible, give a test case that executes the fault, but does not result in an error state. If not, briefly explain why not. (You also need to answer expected and actual output.)
(d) If possible, give a test case that results in an error state, but not a failure. If not, briefly explain why not. (You also need to answer expected and actual output.)
(e) For the given test case in (d), describe the first error state. Be sure to describe the complete state.

Fault:
* A static defect in software.
* Parts of source code that are incorrect.

Error State:
* An incorrect internal state.
* State information: variable values (including return value).

Failure:
* External, incorrect behavior with respect to the expected behavior.