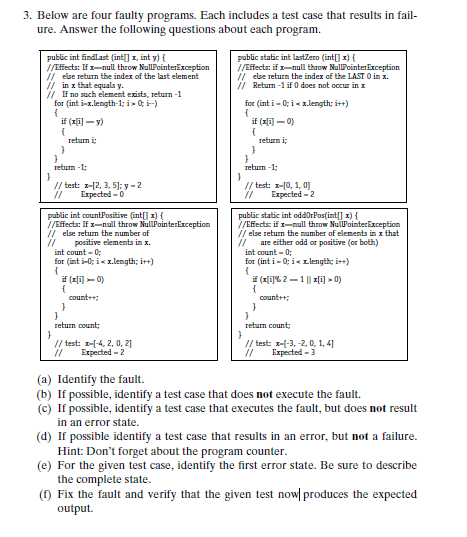
Name: Natnael Kebede

ID: 1001149004



**1** a) the fault in the **findLast** method is that the first element of the test case or 2 is never checked.

b) If we consider a test case where x = null and y = 2 the fault won’t be executed.

c) If we consider a test case where x = [2, 3, 5] and y = 5 the fault will be executed but it won’t result in an error state.

d) If we consider a test case where x = [2, 3, 5] and y = 0 it results in an error but not in a failure.

e) The first error state for the given test case is x = [2, 3, 5], y = 2, x.length = 3, i = 0 and PC = return - 1

f) To fix the fault we have to change the condition i > 0 to i >= 0. This will result in the expected output for the test case given in the question.

**2** a) the fault in the **LastZero** method is that the program returns the index of the first 0 in x but not the last zero in x.

b) If we consider a test case where x = null the fault won’t be executed.

c) If we consider a test case where x = [0] the fault will be executed but it won’t result in an error state.

d) If we consider a test case where x = [0, 1, 1] it results in an error but not in a failure.

e) The first error state for the given test case is x = [0, 1, 0], x.length = 3, i = 0, PC = just after i = 0

f) To fix the fault we have to change the for loop to be for (int i = x.length -1; i >= 0; i--) This will result in the expected output for the test case given in the question.

**3** a) the fault in the **countPositive** method is that the count includes 0 but zero is not positive in x.

b) If we consider a test case where x = null the fault won’t be executed.

c) If we consider a test case where x = [2] the fault will be executed but it won’t result in an error state.

d) It seems impossible to find a test case that results in an error but not in a failure. That is, every input that results in error also results in failure.

e) The first error state for the given test case is x = [-4, 2, 0, 2], x.length = 4, i = 2, count = 0,

PC = if

f) To fix the fault we have to change the if statement inside the for loop to be if (x[i] > 0). This will result in the expected output for the test case given in the question.

**4** a) the fault in the **oddOrPos** method is that the code doesn’t consider non-negative odd numbers.

b) If we consider a test case where x = null the fault won’t be executed.

c) If we consider a test case where x = [3, 2, 1] the fault will be executed but it won’t result in an error state.

d) It seems impossible to find a test case that results in an error but not in a failure. That is, every input that results in error also results in failure.

e) The first error state for the given test case is x = [-3, -2, 0, 1, 4], x.length = 5, i = 0, count = 0, PC = if

f) To fix the fault we have to change the if statement inside the for loop to be if (x[i]%2 == 1 || x[i] > 0 || x[i]%2 == -1). This will result in the expected output for the test case given in the question.