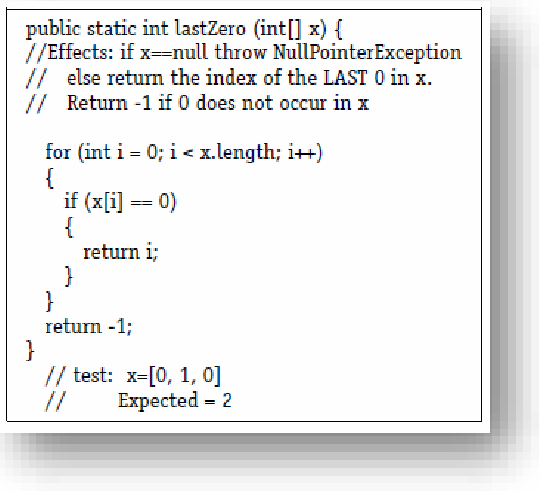
Software Testing & Quality Assurance
Homework 2

Name: Venkata Lakshmi Sasank Tipparaju
Student ID: 700738838
SE 5930 - CRN 31139

Type your answers in this word file and submit it in PDF format.

I. Question 1 (10 points)



```
public static int lastZero (int[] x) {
//Effects: if x==null throw NullPointerException
//   else return the index of the LAST 0 in x.
//   Return -1 if 0 does not occur in x

  for (int i = 0; i < x.length; i++)
  {
    if (x[i] == 0)
    {
      return i;
    }
  }
  return -1;
}
  // test:  x=[0, 1, 0]
  //        Expected = 2
```

1. Identify the fault

   Answer: In for loop $for(int\ i\ =\ 0;\ i < x.length;\ i++)$ will produce the index of first zero in x. The correct code in for loop condition is $for(int\ i = x.length - 1;\ i \geq 0;\ i--)$

2. Identify a test case (test inputs) that results in no error. (Provide an answer that is different from the answer in the slides.)

   Answer: 'Fault but no Error' Occurs when there is no execution of for loop. i.e.,
   when array is empty – we get NullPointerException (or)
   when array has only 1 element then loop executes only once. : i=0 is same for both in initial state
   There is fault but results in no error in above case even if for loop is executed.
   $: x = [15]\ \ expected - 1\ and\ returns - 1$
   $: x = [0]\ \ expected\ 0\ and\ returns\ 0$
   $: x = []$ NullPointerException

3. Identify a test case that results in an error, but not a failure. (Provide an answer that is different from the answer in the slides.)

Answer: error but not a failure: $x = [2,5,0,1,6]$ return 2 (or) $x = [12,25,65,0,85]$ returns 3 Even though for loop is executed with fault that results in error state, failure is not occurred as it produces the expected output.

Error & Failure: $x = [2,5,0,1,0,6]$ – i should be 4 but returns 2 -so error and failure

II. Question 2 (40 points)

One of the fields on a form contains a text box that accepts an exam grade (out of 100).

1. greater than or equal to 90, but less than or equal to 100 – output 'A'
2. greater than or equal to 80, but less than 90 – output 'B'
3. greater than or equal to 70, but less than 80 - output 'C'
4. greater than or equal to 0, but less than 70 - output 'D'
5. All other inputs -output ‘invalid’

Design **7** test cases using Equivalence partitioning. (Define valid and invalid classes(partition) first)

Partitions:

- **Valid:** $\{x|\ \boldsymbol{x \geq 0\ and\ x \leq 100}\}$
- **Invalid:** $\{x|\ \boldsymbol{x < 0\ or\ x > 100}\}$
- **Invalid**: $\{x|\ \boldsymbol{x\ is\ not\ a\ number}\}$

Valid Partition:

$\boldsymbol{\geq 0\ and < 70}$	D
$\boldsymbol{\geq 70\ and < 80}$	**C**
$\boldsymbol{\geq 80\ and < 90}$	**B**
$\boldsymbol{\geq 90\ and \leq 100}$	**A**

Invalid Partition:

$\boldsymbol{< 0}$	**Invalid**
$\boldsymbol{> 100}$	**Invalid**
Not a Number	**Invalid**

Test Case ID	Input	Partition	Output
1	**-85**	< 0	**Invalid**
2	**55**	$\geq 0\ and < 70$	**D**
3	**75**	$\geq 70\ and < 80$	**C**
4	**80**	$\geq 80\ and < 90$	**B**
5	**99**	$\geq 90\ and \leq 100$	**A**
6	**125**	> 100	**Invalid**
7	**Hundred**	**Not a Number**	**Invalid**