Shubham Arya 1001650536

CSE 4321-900
Software testing and maintenance
Homework 2

3.

a) Consider for the following list [1] with only one element. In this case, 1 is both the first entry of the list as well as the last entry of the list. Since this list comes under both Block 1 and Block 2, the location of the element in list fails disjoint property.
Example [3,4,3], e = 3. Element will be in the first entry as well as the last entry.
Or [3], e = 3. Element will be both the first and the last entry.

b) Consider an empty list []. This list does not have a first entry or a last entry. This list does not come under any of the three blocks. This means that the location of element in list fails the completeness property. This also fails if the element is not present in a list.
The problem is that e may not be in the list. Eg [1,2,3], e = 4

c) Block 4: element is the only entry in list.
Block 5: element is empty or null.
Block 6: element is not in the list.
Whether e is first in the list: true or false
Whether e is last in the list: true or false
Whether e is in the list: true or false

4.

a) Input variables can be all types of characters, from numbers to alphabet to special character sets. If a character set is used for another functionality in GenericStack() then that character may not be included.
The null variable is a state variable that will trigger isEmpty() to be true. Anything else would result in false.
Any other data type can be a state variable for the push(Object X). If isEmpty is true, then nothing will be pushed in the stack.
Similarly, any data type can be popped. But if isEmpty() is true, then nothing will be popped.

b) Input variable can be any data type or be of type Object. Input variable can also be a null type with no value.

d) Block1 – integer values like -1, 2, 0. Floating point- 1.2, -3.5. String- “Hello”, “”. Characters- ‘a’,’’,etc.
Block2- null value.

a) There are four testable units here. The constructor and the three functions but there is a substantial overlap between the characteristics relevant for each one. For the three methods, the implicit parameter is the state of the Generic stack. The only explicit input is the Object x parameter in Push(). The constructor has nether inputs nor implicit parameters.

Characteristics for the implicit state are

- Whether the stack is empty
  - True- stack = []
  - False – stack = [“a”], [“b”]
- The size of the stack
  - 0- empty stack
  - 1- stack = [“a”], [null]
  - More than 1= [“a”], [“b”], [null]
- Whether stack contains null entries
  - True- stack = [null], [null, “cat”, “bat”]
  - False- stack = [“cat”, “bat”]
- Whether x is null
  - True- x = null
  - False- x = “a”

There are also characteristics that involve the combination of Object X and the stack state.
Does object x appear in the stack?

- True
- False