**Assignment 5**

**Rasika Mohod**

rmohod@gmu.edu

G01044774

The missed exception: The exception of duplicate value insertions made in the Tree

Consider one of the test case run through the code, where all the eight values taken as input from user are same.

**Test Case 4: Input: All integers inserted same**

**Output:**

Please enter a random integer : 1

Please enter a random integer : 1

Please enter a random integer : 1

Please enter a random integer : 1

Please enter a random integer : 1

Please enter a random integer : 1

Please enter a random integer : 1

Please enter a random integer : 1

Values in order = 1 8

As the output shows the final output of the Tree has only two values: 1 & 8.

8 was inserted by in code implementation and the 1 was given as input by user. Here, consider the add(int val) method of Tree class as below:

**public** **void** add(**int** val)

{

//EFFECTS: Initializes the tree node value

//MODIFIES: Modifies this (the value in the tree, so as to maintain the sorted order: mLeft < mval < mRight)

**if** (val < mVal)

{

**if** (mLeft == **null**)

mLeft = **new** Tree(val);

**else**

mLeft.add(val);

}

**else** **if** (val > mVal)

{

**if** (mRight == **null**)

mRight = **new** Tree(val);

**else**

mRight.add(val);

}

}

This method does validation check of the input variable val: (value to be added in Tree) in two conditions: **if** (val < mVal) and **else** **if** (val > mVal), however this method misses out the condition of new input value being equal to the existing node value of tree.

Thus, **here user client should have been notified by using exception – duplicateValueException** which would have prompted user to inset discrete values each time.

The exception can be included as follows (see the grey shaded part of code):

**public** **void** add(**int** val) **throws** Exception

{

//EFFECTS: Initializes the tree node value

//MODIFIES: Modifies this (the value in the tree, so as to maintain the sorted order: mLeft < mval < mRight)

**if** (val < mVal)

{

**if** (mLeft == **null**)

mLeft = **new** Tree(val);

**else**

mLeft.add(val);

}

**else** **if** (val > mVal)

{

**if** (mRight == **null**)

mRight = **new** Tree(val);

**else**

mRight.add(val);

}

**else**

{

Exception e = **new** Exception ("Duplicate value exception.");

**throw** e;

}

}