TR

L0

[1]

[2]

[5]

[3]

[4]

L1

[1,2]

[1,5]

[2,1]

[2,3]

[3,4]

[3,2]

[4,2]

L2

[1,2,1]

[1,2,3]

[1,5]

[2,1,2]

[2,1,5]

[2,3,4]

[2,3,2]

[3,4,2]

[3,2,1]

[3,2,3]

[4,2,1]

[4,2,3]

Simple Paths

[1,2]

[1,5]

[2,1]

[2,3]

[3,4]

[3,2]

[4,2]

[1,2,1]

[1,2,3]

[2,1,2]

[2,1,5]

[2,3,4]

[2,3,2]

[3,4,2]

[3,2,1]

[3,2,3]

[4,2,1]

[4,2,3]

[1,2,3,4]

[2,3,4,2]

[3,4,2,1]

[3,4,2,3]

[3,2,1,5]

[4,2,1,5]

[4,2,3,4]

[3,4,2,1,5]

Prime Path

[3,4,2,1,5]

[2,3,4,2]

[1,2,3,4]

[3,4,2,3]

[4,2,3,4]

[3,2,1,5]

[2,1,2]

[1,2,1]

[2,3,2]

[3,2,3]

Test Paths for Prime Path coverage

Test Paths Test Requirements

[1,2,3,2,3,2,1,5] [2,1,2] [1,2,1]

[1,2,3,4,2,3,4,2,1,5] [3,4,2,1,5] [2,3,4,2] [1,2,3,4] [3,4,2,3] [4,2,3,4]

[1,2,3,4,2,3,2,1,5] [2,3,4,2] [1,2,3,4] [3,4,2,3] [3,2,1,5] [2,3,2]

[1,2,3,2,3,2,1,5] [3,2,1,5] [2,3,2] [3,2,3]

BNF Grammar

Stream ::= action\*

Action ::= actS

actS ::= B Sequence

Sequence ::= num (“,”num)\* |float(“,”float)\*|string(“,”string)\*|char(“,”char)\*

num ::= digit+

float = digit+.digit+

digit ::= “0” | “1” | “2” | “3” | “4” | “5” | “6” | “7” | “8” | “9”

string ::= char+

char ::= "a" | "b" | "c" | "d" | "e" | "f" | "g" | "h" | "i" | "j" | "k" | "l" | "m" | "n" | "o" | "p" | "q" | "r" | "s" | "t" | "u" | "v" | "w" | "x" | "y" | "z" | "A" | "B" | "C" | "D" | "E" | "F" | "G" | "H" | "I" | "J" | "K" | "L" | "M" | "N" | "O" | "P" | "Q" | "R" | "S" | "T" | "U" | "V" | "W" | "X" | "Y" | "Z"

B 5,4,3,2,1

B a,e,c,v,t,u,i

B Test, test, world, hello, java, Curtin

public static <T extends Comparable<T>> void bubbleSort(T[] arr) {

for (int i = 0; i < arr.length - 1; i++) {

∆1 for (int i = 0; i < arr.length - 1; i+=2) {

for (int j = 1; j < arr.length; j++) {

∆2 for (int j = 1; j < arr.length; j+=2) {

if (arr[j].compareTo(arr[j - 1]) < 0) {

∆3 if (arr[j].compareTo(arr[j - 1]) > 0) {

T temp = arr[j];

arr[j] = arr[j - 1];

arr[j - 1] = temp;

∆4 T temp = arr[j - 1]; // Swap positions

arr[j - 1] = arr[j];

arr[j] = temp;

}

}

}

}

Mutant 1

Reachability: True

Infection: i++ != i+=2

Propagation:

Full Test Specification:

Mutant 2

Reachability:

Infection:

Propagation:

Full Test Specification:

Mutant 3

Reachability:

Infection:

Propagation:

Full Test Specification:

Mutant 4

Reachability:

Infection:

Propagation:

Full Test Specification: