

Test Report: TestReport

**Date:** Sun Aug 18 05:54:38 IST 2019

**Report generated by:** Melan Rashitha Dias

This is basically a sample report.

## Report Table :-

Line Number	Program Statement	Cs Value	CtC Value
CnC Value	Ci Value	TW Value	Cps Value
Cr Value	1.	1	/*
0	0	0	2
2	0	0	2.
2	* To change this license header, choose License Headers in Project Properties.	0	0
0	2	2	0
0	3.	3	* To change this template file, choose Tools   Templates
0	0	0	2
2	0	0	4.
4	* and open the template in the editor.	0	0
0	2	2	0
0	5.	5	*/
0	0	0	2
2	0	0	6.
6	package cnC_calculation;	0	0
0	2	2	0
0	7.	7	
0	0	0	2
2	0	0	8.
8	import java.util.ArrayList;	0	0
0	2	2	0
0	9.	9	import java.util.LinkedHashMap;
0	0	0	2
2	0	0	10.
10	import java.util.regex.Pattern;	0	0
0	2	2	0
0	11.	11	import java.util.regex.Matcher;
0	0	0	2
2	0	0	12.
12		0	0
0	2	2	0
0	13.	13	/**
0	0	0	2
2	0	0	14.
14	*	0	0
0	2	2	0
0	15.	15	* @author melan
0	0	0	2
2	0	0	16.
16	*/	0	0
0	2	2	0
0	17.	17	public class CnCCalculation {
0	0	0	2
2	0	0	18.
18		0	0
0	2	2	0
0	19.	19	//var base
0	0	0	2
2	0	0	20.
20	ArrayList<String> lineOfCode;	0	0
0	2	2	0
0	21.	21	ArrayList<Integer> cncPoints;
0	0	0	2
2	0	0	22.
22	static LinkedHashMap<Integer, Integer> cncLinePoint = new LinkedHashMap<Integer, Integer>();	0	0
0	2	2	0

Line Number	Program Statement	Cs Value	CtC Value
0	23.	23	int bracketState = 0; //initially bracket state at globle value map
0	0	0	2
2	0	0	24.
24		0	0
0	2	2	0
0	25.	25	//REGX block
0	0	0	2
2	0	0	26.
26	String openBracketsREGX = "\\{";	0	0
0	2	2	0
0	27.	27	String closedBracketREGX = "\\}";
0	0	0	2
2	0	0	28.
28	String bracketsREGX = "\\b((if while for do)(\\s+ \\( \\) \\. \\s \\s*))";	0	0
0	2	2	0
0	29.	29	String singleLineREGX = "^((\\s* \\s \\s*) ^\\s*)\$";
0	0	0	2
2	0	0	30.
30		0	0
0	2	2	0
0	31.	31	public CncCalculation(ArrayList<String> lineOfCode) {
0	0	0	2
2	0	0	32.
32	this.lineOfCode = lineOfCode;	0	0
0	2	2	0
0	33.	33	cncPoints = new ArrayList<Integer>(lineOfCode.size()) ;
0	0	0	2
2	0	0	34.
34	}	0	0
0	2	2	0
0	35.	35	
0	0	0	2
2	0	0	36.
36	public void incrementBracketState() {	0	0
0	2	2	0
0	37.	37	System.out.println("increment state CNC");
0	0	0	2
2	0	0	38.
38	this.bracketState++;	0	0
0	2	2	0
0	39.	39	}
0	0	0	2
2	0	0	40.
40		0	0
0	2	2	0
0	41.	41	public void decrementBracketState() {
0	0	0	2
2	0	0	42.
42		0	0
0	2	2	0
0	43.	43	System.out.println("Decrement state CNC");
0	0	0	2
2	0	0	44.
44	this.bracketState--;	0	0
0	2	2	0

Line Number	Program Statement	Cs Value	CtC Value
0	45.	45	
0	0	0	2
2	0	0	46.
46	}	0	0
0	2	2	0
0	47.	47	
0	0	0	2
2	0	0	48.
48	public void coreBracketMapper() {	0	0
0	2	2	0
0	49.	49	Pattern bracketsPattern = Pattern.compile(bracketsREGX);
0	0	0	2
2	0	0	50.
50	Pattern bracketsSingle = Pattern.compile(singleLineREGX);	0	0
0	2	2	0
0	51.	51	Pattern bracketsClosed = Pattern.compile(closedBracketREGX);
0	0	0	2
2	0	0	52.
52		0	0
0	2	2	0
0	53.	53	// Check each line for Cnc
0	0	0	2
2	0	0	54.
54	for (int i = 0; i < lineOfCode.size(); i++) {	0	2
1	2	5	0
0	55.	55	int count = 0;
0	0	1	2
3	0	0	56.
56	String line = lineOfCode.get(i);	0	0
1	2	3	0
0	57.	57	// check for conditions and loops
0	0	1	2
3	0	0	58.
58	Matcher matcher = bracketsPattern.matcher(line);	0	0
1	2	3	0
0	59.	59	while (matcher.find()) {
0	2	2	2
6	0	0	60.
60	incrementBracketState();	0	0
2	2	4	0
0	61.	61	}
0	0	0	2
2	0	0	62.
62	count = bracketState;	0	0
1	2	3	0
0	63.	63	
0	0	0	2
2	0	0	64.
64	Matcher close_m = bracketsClosed.matcher(line);	0	0
1	2	3	0
0	65.	65	while (close_m.find()) {
0	2	2	2
6	0	0	66.
66	decrementBracketState();	0	0
2	2	4	0
0	67.	67	}
0	0	0	2
2	0	0	68.

Line Number	Program Statement	Cs Value	CtC Value
68	// check for lines with brackets or empty line	0	0
1	2	3	0
0	69.	69	Matcher singleline = bracketsSingle.matcher(line);
0	0	1	2
3	0	0	70.
70	if (singleline.find()) {	0	1
2	2	5	0
0	71.	71	count = 0;
0	0	2	2
4	0	0	72.
72	}	0	0
0	2	2	0
0	73.	73	cncPoints.add(count);
0	0	1	2
3	0	0	74.
74	}	0	0
0	2	2	0
0	75.	75	}
0	0	0	2
2	0	0	76.
76		0	0
0	2	2	0
0	77.	77	public ArrayList<Integer> getCnc() {
0	0	0	2
2	0	0	78.
78	coreBracketMapper();	0	0
0	2	2	0
0	79.	79	return cncPoints;
0	0	0	2
2	0	0	80.
80	}	0	0
0	2	2	0
0	81.	81	
0	0	0	2
2	0	0	82.
82	public int getTotalCncPoints() {	0	0
0	2	2	0
0	83.	83	int total = 0;
0	0	0	2
2	0	0	84.
84		0	0
0	2	2	0
0	85.	85	for (int i = 0; i < cncPoints.size(); i++) {
0	2	1	2
5	0	0	86.
86	total += cncPoints.get(i);	0	0
1	2	3	0
0	87.	87	}
0	0	0	2
2	0	0	88.
88		0	0
0	2	2	0
0	89.	89	return total;
0	0	0	2
2	0	0	90.
90	}	0	0
0	2	2	0
0	91.	91	
0	0	0	2
2	0	0	92.

Line Number	Program Statement	Cs Value	CtC Value
92	}	0	0
0	2	2	0
0	93.	93	
0	0	0	2