

```

public class Test {
    int sum;
    public int y;
    public void methodA() {
        int x = 0, y = 0;
        y = y + 7;
        x = y + 11;
        sum = x + y;
        System.out.println(x + " " + y + " " + sum);
    }
    public void methodB() {
        int x = 0;
        y = y + 11;
        x = x + 33 + y;
        sum = sum + x + y;
        System.out.println(x + " " + y + " " + sum);
    }
    // Trace the outputs
    public static void main(String[] args) {
        Test t = new Test();
        t.methodA();
        t.methodB();
    }
}

```

states	local x	local y	instance sum
initial	0	0	0
y = y + 7	0	7	0
x = y + 11	18	7	0
Sum = x + y	18	7	25
Print	18	7	25

states	local x	instance y	instance sum
initial	0	0	25
y = y + 11	0	11	25
x = x + 33 + y	44	11	25
sum = sum + x + y	44	11	80
Print	44	11	80

```

public class Q3 {
    public static void main(String[] args) {
        String test = "";
        int i = 5, j = 0, k = 15;
        while (i < 10) {
            k -= 1;
            j = k;
            while (j > 10) {
                if (j % 2 == 0) {
                    test = "<--";
                    test = test + i + 2 + "-->" + (j / 2);
                } else {
                    test = "-->";
                    test = "-->" + (i / 2) + test + j;
                }
                System.out.println(test);
                --j;
            }
            i++;
        }
    }
}

```

The diagram illustrates the execution flow of the Q3 program through four state tables. Red arrows connect the tables sequentially, showing the progression of variable values (i, j, k) and the state of the program (initial, while loops, if statements, prints).

states	i	j	k
initial	5	0	15

states	i	j	k
while	7	12	12
while		12	
if		j even	
print	<--72-->6		
while		11	
else		j odd	
print	-->3-->11		

states	i	j	k
while	5	14	14
while		14	
if	j even		
print	<--52-->7		
while		13	
else	j odd		
print	-->2-->13		
while		12	
if	j even		
print	<--62-->6		
while		11	
else	j odd		
print	-->3-->11		

states	i	j	k
while	6	13	13
while		13	
else	j odd		
print	-->3-->13		
while		12	
if	j even		
print	<--62-->6		
while		11	
else	j odd		
print	-->3-->11		

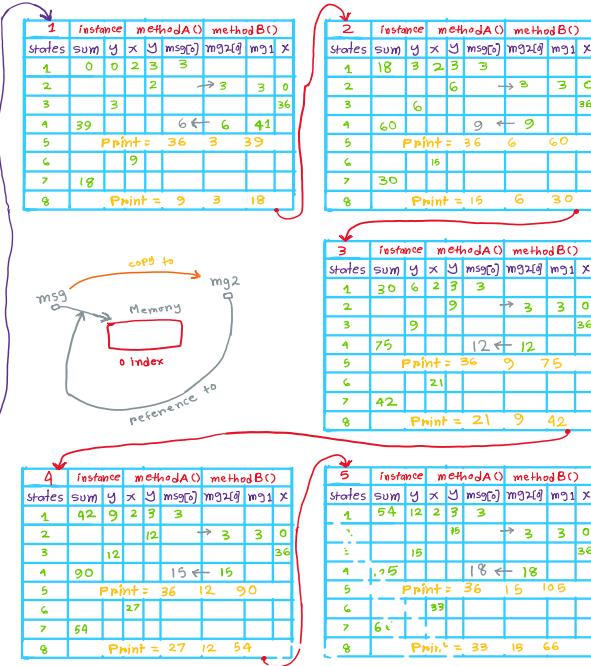
states	i	j	k
while	9	10	10
while	10		
End of program			

```

public class Test3 {
    public int sum;
    public int y;
    public void methodA() {
        int x = 2, y = 3;
        int[] msg = new int[1];
        msg[0] = 1;
        2. msg[0] = x + msg[0];
        methodB(msg, msg[0]);
        x = this.y + msg[0];
        sum = x + y + msg[0];
        System.out.println(x + " " + y + " " + sum);
    }
    private void methodB(int[] mg2, int mg1) {
        int x = 0;
        3. y = this.y + mg2[0];
        x = x + 33 + mg1;
        sum = y + mg1;
        mg2[0] = y + mg1;
        mg1 = mg1 + x + 2;
        System.out.println(x + " " + y + " " + sum);
    }
}

// Tracing the outputs
public static void main(String[] args) {
    Test3 t = new Test3();
    for (int i = 0; i < 5; ++i) {
        t.methodA();
    }
}

```



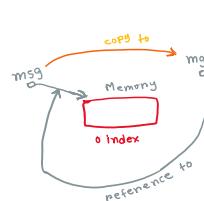
```

public class Test4 {
    public static int sum;
    public int y;
    public void methodA() {
        int x = 0 + 0;
        int[] msg = new int[1];
        msg[0] = 5;
        y = x + y * methodB(msg);
        X = y * y + methodB(msg, msg[0]);
        sum = x + y + sum;
        System.out.println(x + " " + y + " " + sum);
    }
    private int methodB(int mrgl2), int mg1) {
        int x = 0;
        y = y * y + 20;
        x = x + 33 + mg1;
        sum = sum + x + y;
        mrgl2[0] = y + mg1;
        y = mg1 + x + 2;
        System.out.println(x + " " + y + " " + sum);
        return sum;
    }
    private int methodB(int mg1) {
        int x = 0;
        int y = 0;
        y = y * mg1;
        x = x + 33 + mg1;
        sum = sum + x + y;
        this.y = mg1 + x + 2;
        System.out.println(x + " " + y + " " + sum);
        return sum;
    }
}

// Tracing the outputs
public static void main(String[] args) {
    Test4 t = new Test4();
    for (int i = 0; i < 4; ++i) {
        t.methodA();
    }
}

```

1	instance	methodA()	methodB()	methodC()
stories	sum	y	x	msg[0]
1	0	0	0	5
2				5 0 0
3				38
4	43 45			
5		printf = 38	5	43
6		5		
7			→ 5	5 0
8		50		28
9	131		55 ← 55	
10				45
11		printf = 38 50		131
12		136		
13	196			
14		printf = 136 5		196

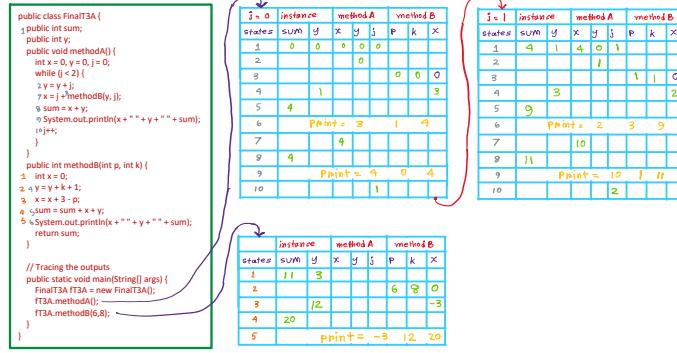


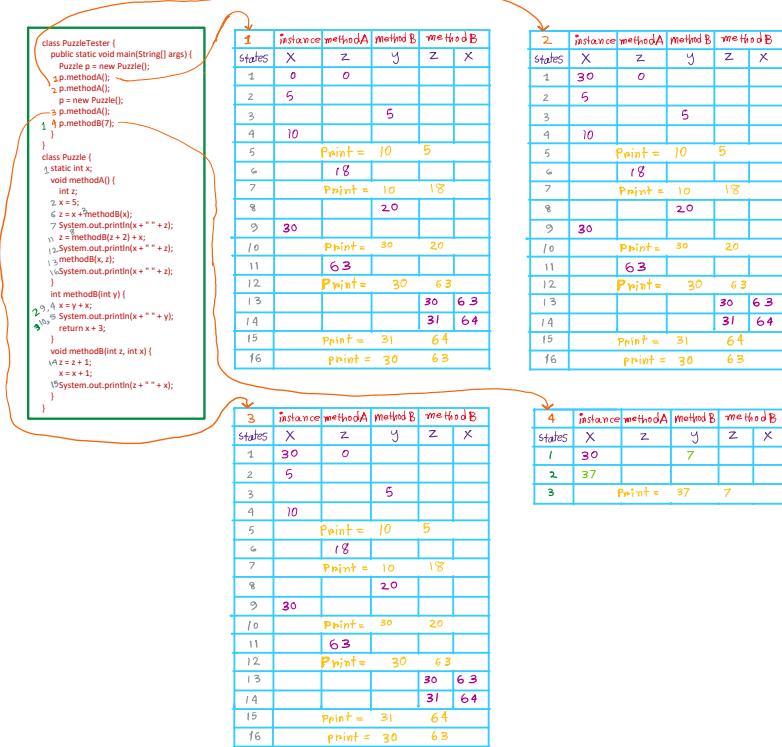
9	instance	methodA()	methodB()	methodB
shares	sum	y	x	msg
1	588	50	0	5
2				5 0
3				38
4	631	95		
5		Print = 38	5	631
6		5		
7			→ 5	5 0
8	50			38
9	719		55 ← 55	
10				45
11		Print = 38	50	
12		724		
13	784			
14		Print = 724	5	784

	2	Resistance	methodA()	methodB()	methodC()
states	sum	$y \times y$	$m[0]$	$m[2] \times m[1]$	$m[2] \times x$
1	196	50	0	0	5
2					0
3					38
4	239	45			
5			Print = 38	5	239
6			5		
7				5	0
8		50			38
9	327		55	55	
10					45
11			PRINT = 38	50	327
12		332			
13	392				
14			Print = 332	5	392

3	instance	methodA()	methodB()	methodC()
states	sum	y X msg[0]	m32[0] m32[1] m31	X msg[1] msg[2]
1	392	50	0 0	5
2				5 0
3				38
4	135	45		
5			Print = 38	5 435
6		5		
7			→ 5	5 0
8		50		38
9	523		55 ← 55	
10				45
11		Print = 38	50	523
12		528		
13	588			
14		Print = 528	5	588

S	Resistance	methodA()	methodB()	methodB
states	Sum Y	X Y msqY	msqZ	msq Y
1	789450	0 0	5	
2				5 0
3				38
4	827495			
5		Print = 38	5	827
6		5		
7			→ 5	5 0
8		50		38
9	915		55 ← 55	
10				45
11		Print = 38	50	915
12		920		
13	980			
14		Print = 920	5	980





Puzzle	X	Z	Y	Z	X	Method A	Method B	Method C	Method D	Maze	States	X	m	Y	Z	X
States																
1	30		7							8	0	0				
2	37									9	5					
3			Point = 37	7						10			5			
										11	25					
4	10									12	Point = 25	5				
5										13	33					
6	18									14	Point = 25	33				
7			Point = 10	18						15		30				
8										16	900					
9	20									17	Point = 900	30				
10	30									18	1803					
11			Point = 30	63						19	Point = 900	1803				
12										20			900	1803		
13										21			998	1803		
14										22	Point = 998	1803				
15										23	Point = 900	1803900				
16																