

1 Public Test Cases

1.1 Test Case 3

1.1.1 Program

```
static void fun3(int value)
{
    BasicTest v1 = new BasicTest();
    BasicTest v2 = new BasicTest();
        v2.f = v2;
        if(value <= 100)
        {
            v2.f = v1;
        }
        else
            v2.f = null;
        // Situation 4 partially -- union for an object field
    }
```

1.1.2 Intermediate Representation

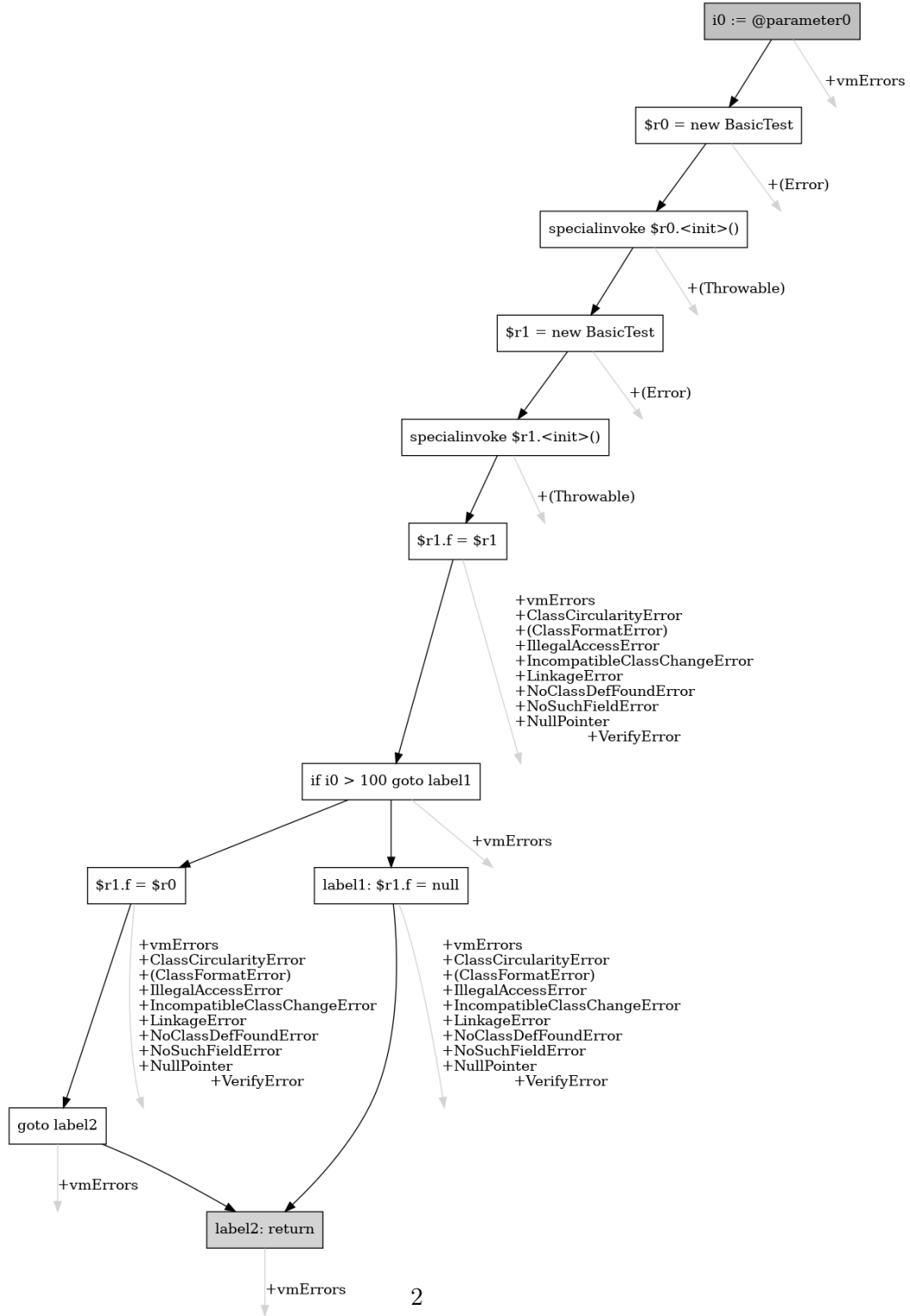
```
00: i0 := @parameter0: int
01: $r0 = new BasicTest
02: specialinvoke $r0.<BasicTest: void <init>()>()
03: $r1 = new BasicTest
04: specialinvoke $r1.<BasicTest: void <init>()>()
05: $r1.<BasicTest: BasicTest f> = $r1
06: if i0 > 100 goto label1
07: $r1.<BasicTest: BasicTest f> = $r0
08: goto label2
09: $r1.<BasicTest: BasicTest f> = null
10: return
```

1.1.3 CFG

1

1.1.4 Output

```
BasicTest.fun3: in02: $r0: {new01}
BasicTest.fun3: in03: $r0: {new01}
```



2

void fun3(int)

Figure 1: TC-3

BasicTest.fun3: in04: \$r0: {new01}
BasicTest.fun3: in04: \$r1: {new03}
BasicTest.fun3: in05: \$r0: {new01}
BasicTest.fun3: in05: \$r1: {new03}
BasicTest.fun3: in06: \$r0: {new01}
BasicTest.fun3: in06: \$r1: {new03}
BasicTest.fun3: in06: new03.f: {new03}
BasicTest.fun3: in07: \$r0: {new01}
BasicTest.fun3: in07: \$r1: {new03}
BasicTest.fun3: in07: new03.f: {new03}
BasicTest.fun3: in08: \$r0: {new01}
BasicTest.fun3: in08: \$r1: {new03}
BasicTest.fun3: in08: new03.f: {new03, new01}
BasicTest.fun3: in09: \$r0: {new01}
BasicTest.fun3: in09: \$r1: {new03}
BasicTest.fun3: in09: new03.f: {new03}
BasicTest.fun3: in10: \$r1: {new03}
BasicTest.fun3: in10: \$r0: {new01}
BasicTest.fun3: in10: new03.f: {new03, new01, null}

1.2 Test Case 4

1.2.1 Program

```
static void fun4(int value)
{
    BasicTest v1 = new BasicTest();
    BasicTest v2 = new BasicTest();
    BasicTest v3 = new BasicTest();
    v2.f = v3;
    while(value < 100){
        v2.f = new BasicTest();
        value += 1;    // Situation 5a partially (object field only)
    }
    v3.f = v2.f;
}
```

1.2.2 Intermediate Representation

```
00: i0 := @parameter0: int
01: $r0 = new BasicTest
02: specialinvoke $r0.<BasicTest: void <init>()>()
03: $r1 = new BasicTest
04: specialinvoke $r1.<BasicTest: void <init>()>()
05: $r2 = new BasicTest
06: specialinvoke $r2.<BasicTest: void <init>()>()
07: $r1.<BasicTest: BasicTest f> = $r2
08: if i0 >= 100 goto label2
09: $r4 = new BasicTest
10: specialinvoke $r4.<BasicTest: void <init>()>()
11: $r1.<BasicTest: BasicTest f> = $r4
12: i0 = i0 + 1
13: goto label1
14: $r3 = $r1.<BasicTest: BasicTest f>
15: $r2.<BasicTest: BasicTest f> = $r3
16: return
```

1.2.3 CFG

2

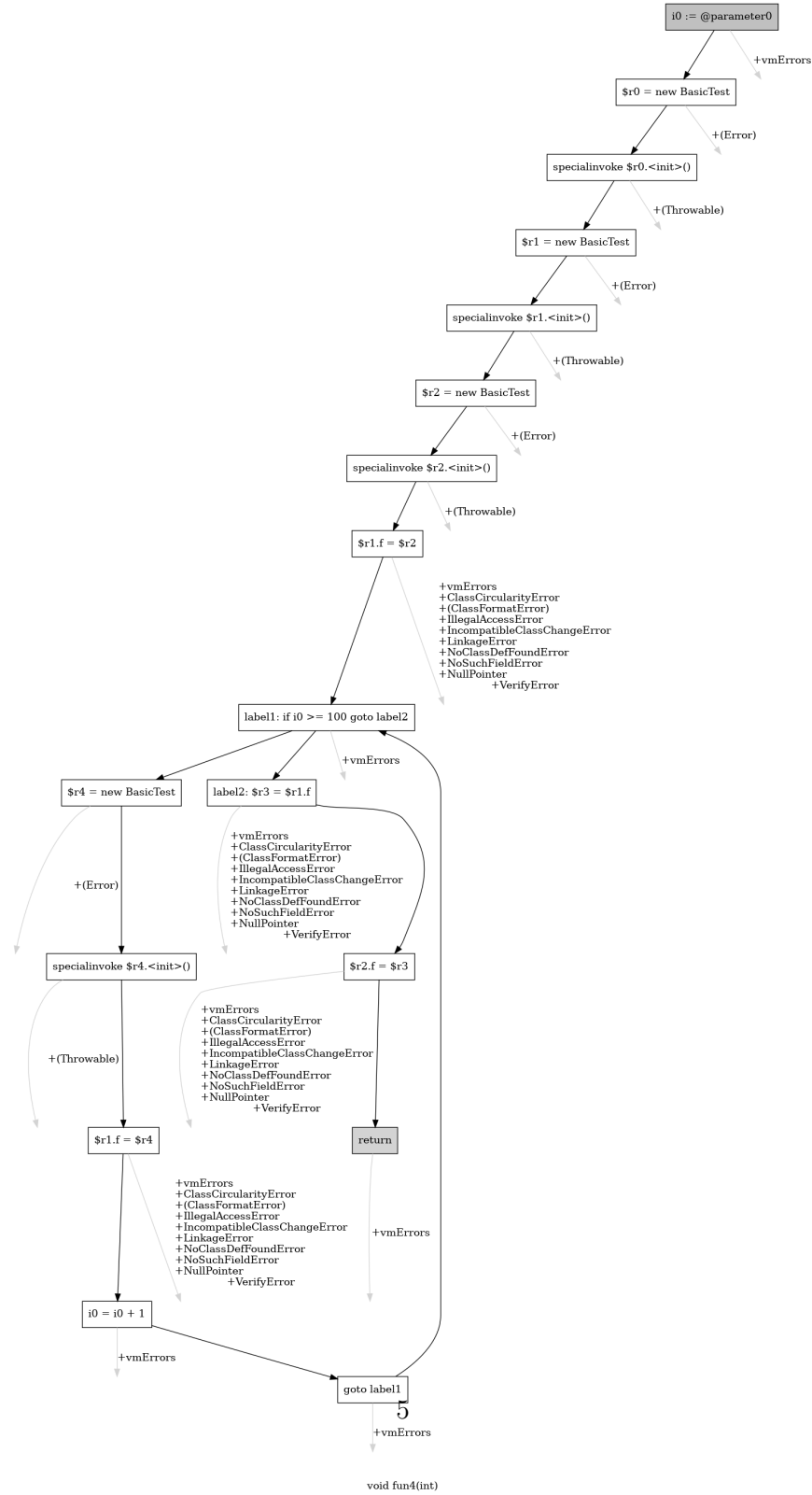


Figure 2: TC-4

1.2.4 Output

```
BasicTest.fun4: in02: $r0: {new01}
BasicTest.fun4: in03: $r0: {new01}
BasicTest.fun4: in04: $r0: {new01}
BasicTest.fun4: in04: $r1: {new03}
BasicTest.fun4: in05: $r0: {new01}
BasicTest.fun4: in05: $r1: {new03}
BasicTest.fun4: in06: $r0: {new01}
BasicTest.fun4: in06: $r1: {new03}
BasicTest.fun4: in06: $r2: {new05}
BasicTest.fun4: in07: $r0: {new01}
BasicTest.fun4: in07: $r1: {new03}
BasicTest.fun4: in07: $r2: {new05}
BasicTest.fun4: in08: $r0: {new01}
BasicTest.fun4: in08: $r1: {new03}
BasicTest.fun4: in08: $r2: {new05}
BasicTest.fun4: in08: $r4: {new09}
BasicTest.fun4: in08: new03.f: {new05, new09}
BasicTest.fun4: in09: $r0: {new01}
BasicTest.fun4: in09: $r1: {new03}
BasicTest.fun4: in09: $r2: {new05}
BasicTest.fun4: in09: $r4: {new09}
BasicTest.fun4: in09: new03.f: {new05, new09}
BasicTest.fun4: in10: $r0: {new01}
BasicTest.fun4: in10: $r1: {new03}
BasicTest.fun4: in10: $r2: {new05}
BasicTest.fun4: in10: $r4: {new09}
BasicTest.fun4: in10: new03.f: {new05, new09}
BasicTest.fun4: in11: $r0: {new01}
BasicTest.fun4: in11: $r1: {new03}
BasicTest.fun4: in11: $r2: {new05}
BasicTest.fun4: in11: $r4: {new09}
BasicTest.fun4: in11: new03.f: {new05, new09}
BasicTest.fun4: in12: $r0: {new01}
BasicTest.fun4: in12: $r1: {new03}
BasicTest.fun4: in12: $r2: {new05}
BasicTest.fun4: in12: $r4: {new09}
BasicTest.fun4: in12: new03.f: {new05, new09}
BasicTest.fun4: in13: $r0: {new01}
```

BasicTest.fun4: in13: \$r1: {new03}
BasicTest.fun4: in13: \$r2: {new05}
BasicTest.fun4: in13: \$r4: {new09}
BasicTest.fun4: in13: new03.f: {new05, new09}
BasicTest.fun4: in14: \$r0: {new01}
BasicTest.fun4: in14: \$r1: {new03}
BasicTest.fun4: in14: \$r2: {new05}
BasicTest.fun4: in14: \$r4: {new09}
BasicTest.fun4: in14: new03.f: {new05, new09}
BasicTest.fun4: in15: \$r0: {new01}
BasicTest.fun4: in15: \$r1: {new03}
BasicTest.fun4: in15: \$r2: {new05}
BasicTest.fun4: in15: \$r3: {new05, new09}
BasicTest.fun4: in15: \$r4: {new09}
BasicTest.fun4: in15: new03.f: {new05, new09}
BasicTest.fun4: in16: \$r0: {new01}
BasicTest.fun4: in16: \$r1: {new03}
BasicTest.fun4: in16: \$r2: {new05}
BasicTest.fun4: in16: \$r3: {new05, new09}
BasicTest.fun4: in16: \$r4: {new09}
BasicTest.fun4: in16: new03.f: {new05, new09}
BasicTest.fun4: in16: new05.f: {new05, new09}

1.3 Test Case 5

1.3.1 Program

```
static void fun5(int value)
{
    BasicTest v1;
    BasicTest v2 = new BasicTest();
    BasicTest v3 = new BasicTest();
    BasicTest v4 = new BasicTest();

    v2.f = null;
    v3.f = v4;

    if(value == 100)
    {
        v1 = v2;          // Situation 3 (strong update)
    }
    else if (value == 200)
    {
        v1 = v3;          // Situation 3 (strong update)
    }
    else {
        v1 = null;
    }
    // Situation 4 partially -- union for a variable
    BasicTest v5 = v1.f; // Situation 6, Situation 8
    BasicTest v6 = new BasicTest();

    v1.f = v6;           // Situation 1, Situation 5b, Situation 7
    v5 = v1.f;           // Situation 6
}
```

1.3.2 Intermediate Representation

```
00: i0 := @parameter0: int
01: $r0 = new BasicTest
02: specialinvoke $r0.<BasicTest: void <init>()>()
03: $r1 = new BasicTest
04: specialinvoke $r1.<BasicTest: void <init>()>()
05: $r2 = new BasicTest
```



```

06: specialinvoke $r2.<BasicTest: void <init>()>()
07: $r0.<BasicTest: BasicTest f> = null
08: $r1.<BasicTest: BasicTest f> = $r2
09: if i0 != 100 goto label1
10: r3 = $r0
11: goto label3
12: if i0 != 200 goto label2
13: r3 = $r1
14: goto label3
15: r3 = null
16: $r4 = r3.<BasicTest: BasicTest f>
17: $r5 = new BasicTest
18: specialinvoke $r5.<BasicTest: void <init>()>()
19: r3.<BasicTest: BasicTest f> = $r5
20: $r6 = r3.<BasicTest: BasicTest f>
21: return

```

1.3.3 CFG

3

1.3.4 Output

```

BasicTest.fun5: in02: $r0: {new01}
BasicTest.fun5: in03: $r0: {new01}
BasicTest.fun5: in04: $r0: {new01}
BasicTest.fun5: in04: $r1: {new03}
BasicTest.fun5: in05: $r0: {new01}
BasicTest.fun5: in05: $r1: {new03}
BasicTest.fun5: in06: $r0: {new01}
BasicTest.fun5: in06: $r1: {new03}
BasicTest.fun5: in06: $r2: {new05}
BasicTest.fun5: in07: $r0: {new01}
BasicTest.fun5: in07: $r1: {new03}
BasicTest.fun5: in07: $r2: {new05}
BasicTest.fun5: in08: $r0: {new01}
BasicTest.fun5: in08: $r1: {new03}
BasicTest.fun5: in08: $r2: {new05}
BasicTest.fun5: in08: new01.f: {null}

```

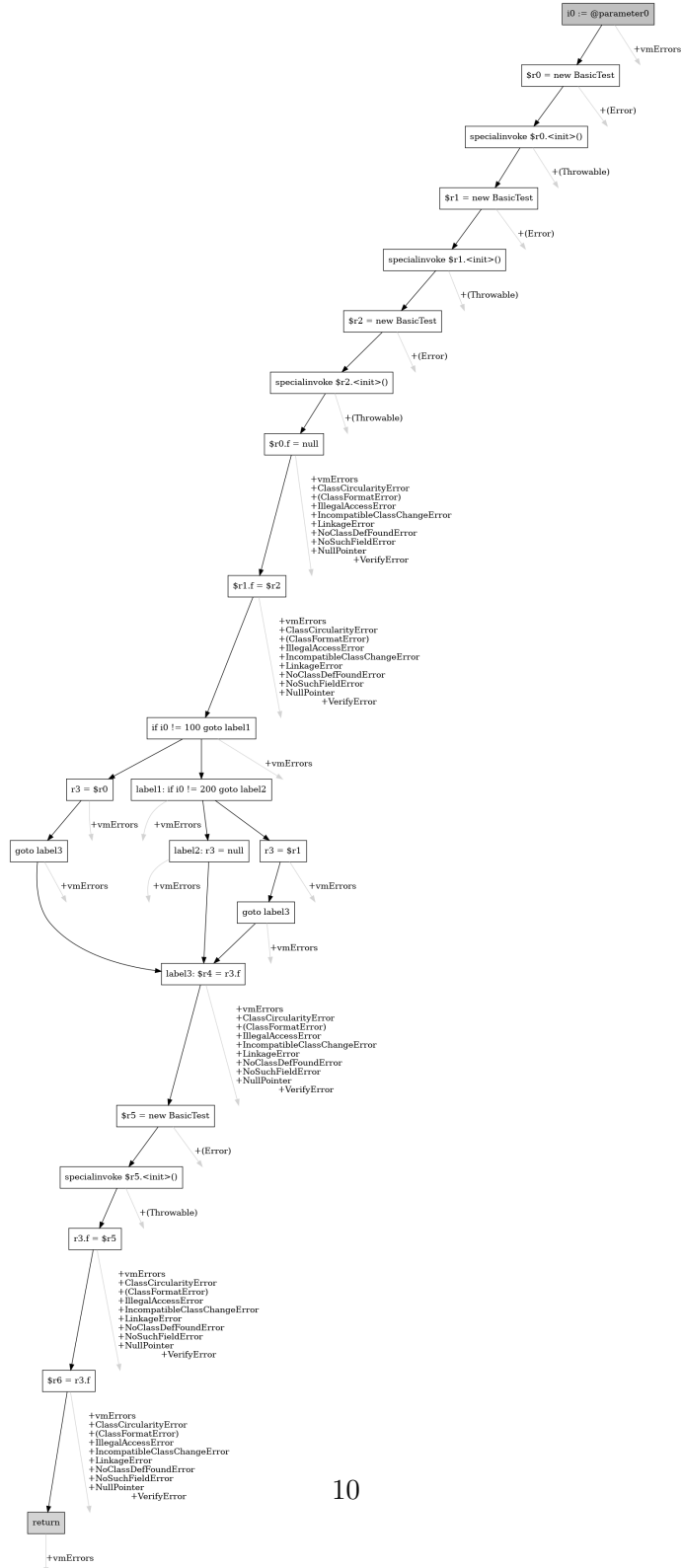


Figure 3: TC-5

BasicTest.fun5: in09: \$r0: {new01}
BasicTest.fun5: in09: \$r1: {new03}
BasicTest.fun5: in09: \$r2: {new05}
BasicTest.fun5: in09: new01.f: {null}
BasicTest.fun5: in09: new03.f: {new05}
BasicTest.fun5: in10: \$r0: {new01}
BasicTest.fun5: in10: \$r1: {new03}
BasicTest.fun5: in10: \$r2: {new05}
BasicTest.fun5: in10: new01.f: {null}
BasicTest.fun5: in10: new03.f: {new05}
BasicTest.fun5: in11: \$r0: {new01}
BasicTest.fun5: in11: \$r1: {new03}
BasicTest.fun5: in11: \$r2: {new05}
BasicTest.fun5: in11: new01.f: {null}
BasicTest.fun5: in11: new03.f: {new05}
BasicTest.fun5: in11: r3: {new01}
BasicTest.fun5: in12: \$r0: {new01}
BasicTest.fun5: in12: \$r1: {new03}
BasicTest.fun5: in12: \$r2: {new05}
BasicTest.fun5: in12: new01.f: {null}
BasicTest.fun5: in12: new03.f: {new05}
BasicTest.fun5: in13: \$r0: {new01}
BasicTest.fun5: in13: \$r1: {new03}
BasicTest.fun5: in13: \$r2: {new05}
BasicTest.fun5: in13: new01.f: {null}
BasicTest.fun5: in13: new03.f: {new05}
BasicTest.fun5: in14: \$r0: {new01}
BasicTest.fun5: in14: \$r1: {new03}
BasicTest.fun5: in14: \$r2: {new05}
BasicTest.fun5: in14: new01.f: {null}
BasicTest.fun5: in14: new03.f: {new05}
BasicTest.fun5: in14: r3: {new03}
BasicTest.fun5: in15: \$r0: {new01}
BasicTest.fun5: in15: \$r1: {new03}
BasicTest.fun5: in15: \$r2: {new05}
BasicTest.fun5: in15: new01.f: {null}
BasicTest.fun5: in15: new03.f: {new05}
BasicTest.fun5: in16: \$r0: {new01}
BasicTest.fun5: in16: \$r1: {new03}
BasicTest.fun5: in16: \$r2: {new05}

BasicTest.fun5: in16: new01.f: {null}
 BasicTest.fun5: in16: new03.f: {new05}
 BasicTest.fun5: in16: r3: {new01, new03, null}
 BasicTest.fun5: in17: \$r0: {new01}
 BasicTest.fun5: in17: \$r1: {new03}
 BasicTest.fun5: in17: \$r2: {new05}
 BasicTest.fun5: in17: \$r4: {new05, null}
 BasicTest.fun5: in17: new01.f: {null}
 BasicTest.fun5: in17: new03.f: {new05}
 BasicTest.fun5: in17: r3: {new01, new03, null}
 BasicTest.fun5: in18: \$r0: {new01}
 BasicTest.fun5: in18: \$r1: {new03}
 BasicTest.fun5: in18: \$r2: {new05}
 BasicTest.fun5: in18: \$r4: {new05, null}
 BasicTest.fun5: in18: \$r5: {new17}
 BasicTest.fun5: in18: r3: {new01, new03, null}
 BasicTest.fun5: in18: new01.f: {null}
 BasicTest.fun5: in18: new03.f: {new05}
 BasicTest.fun5: in19: \$r0: {new01}
 BasicTest.fun5: in19: \$r1: {new03}
 BasicTest.fun5: in19: \$r2: {new05}
 BasicTest.fun5: in19: \$r4: {new05, null}
 BasicTest.fun5: in19: \$r5: {new17}
 BasicTest.fun5: in19: new01.f: {null}
 BasicTest.fun5: in19: new03.f: {new05}
 BasicTest.fun5: in19: r3: {new01, new03, null}
 BasicTest.fun5: in20: \$r0: {new01}
 BasicTest.fun5: in20: \$r1: {new03}
 BasicTest.fun5: in20: \$r2: {new05}
 BasicTest.fun5: in20: \$r4: {new05, null}
 BasicTest.fun5: in20: \$r5: {new17}
 BasicTest.fun5: in20: new01.f: {new17, null}
 BasicTest.fun5: in20: new03.f: {new05, new17}
 BasicTest.fun5: in20: r3: {new01, new03, null}
 BasicTest.fun5: in21: \$r0: {new01}
 BasicTest.fun5: in21: \$r1: {new03}
 BasicTest.fun5: in21: \$r2: {new05}
 BasicTest.fun5: in21: \$r4: {new05, null}
 BasicTest.fun5: in21: \$r5: {new17}
 BasicTest.fun5: in21: \$r6: {new05, new17, null}

BasicTest.fun5: in21: new01.f: {new17, null}
BasicTest.fun5: in21: new03.f: {new05, new17}
BasicTest.fun5: in21: r3: {new01, new03, null}

1.4 Test Case 6

1.4.1 Program

```
static void fun6(int value, int n)
{
    BasicTest v1, v2 = null;
    int old = value;
    do {
        v1 = new BasicTest();
        if (value % 2 == 0) {
            v2 = v1;
        }
        value++;
    } while (value - old < 2 || value < n);
    v2.f = new BasicTest();
    if (v1 == null)
        v2.f = new BasicTest(); // unreachable
    if (v1 != v2)
        v2.f = new BasicTest(); // reachable
}
```

1.4.2 Intermediate Representation

```
00: i3 := @parameter0: int
01: i2 := @parameter1: int
02: r3 = null
03: i0 = i3
04: $r4 = new BasicTest
05: specialinvoke $r4.<BasicTest: void <init>()>()
06: $i4 = i3 % 2
07: if $i4 != 0 goto label2
08: r3 = $r4
09: i3 = i3 + 1
10: $i1 = i3 - i0
11: if $i1 < 2 goto label1
12: if i3 < i2 goto label1
13: $r0 = new BasicTest
14: specialinvoke $r0.<BasicTest: void <init>()>()
15: r3.<BasicTest: BasicTest f> = $r0
16: if $r4 != null goto label3
```

```

17: $r2 = new BasicTest
18: specialinvoke $r2.<BasicTest: void <init>()>()
19: r3.<BasicTest: BasicTest f> = $r2
20: if $r4 == r3 goto label4
21: $r1 = new BasicTest
22: specialinvoke $r1.<BasicTest: void <init>()>()
23: r3.<BasicTest: BasicTest f> = $r1
24: return

```

1.4.3 CFG

4

1.4.4 Output

```

BasicTest.fun6: in03: r3: {null}
BasicTest.fun6: in04: $r4: {new04}
BasicTest.fun6: in04: r3: {new04, null}
BasicTest.fun6: in05: $r4: {new04}
BasicTest.fun6: in05: r3: {new04, null}
BasicTest.fun6: in06: $r4: {new04}
BasicTest.fun6: in06: r3: {new04, null}
BasicTest.fun6: in07: $r4: {new04}
BasicTest.fun6: in07: r3: {new04, null}
BasicTest.fun6: in08: $r4: {new04}
BasicTest.fun6: in08: r3: {new04, null}
BasicTest.fun6: in09: $r4: {new04}
BasicTest.fun6: in09: r3: {new04, null}
BasicTest.fun6: in10: $r4: {new04}
BasicTest.fun6: in10: r3: {new04, null}
BasicTest.fun6: in11: $r4: {new04}
BasicTest.fun6: in11: r3: {new04, null}
BasicTest.fun6: in12: $r4: {new04}
BasicTest.fun6: in12: r3: {new04, null}
BasicTest.fun6: in13: $r4: {new04}
BasicTest.fun6: in13: r3: {new04, null}
BasicTest.fun6: in14: $r0: {new13}
BasicTest.fun6: in14: $r4: {new04}
BasicTest.fun6: in14: r3: {new04, null}

```

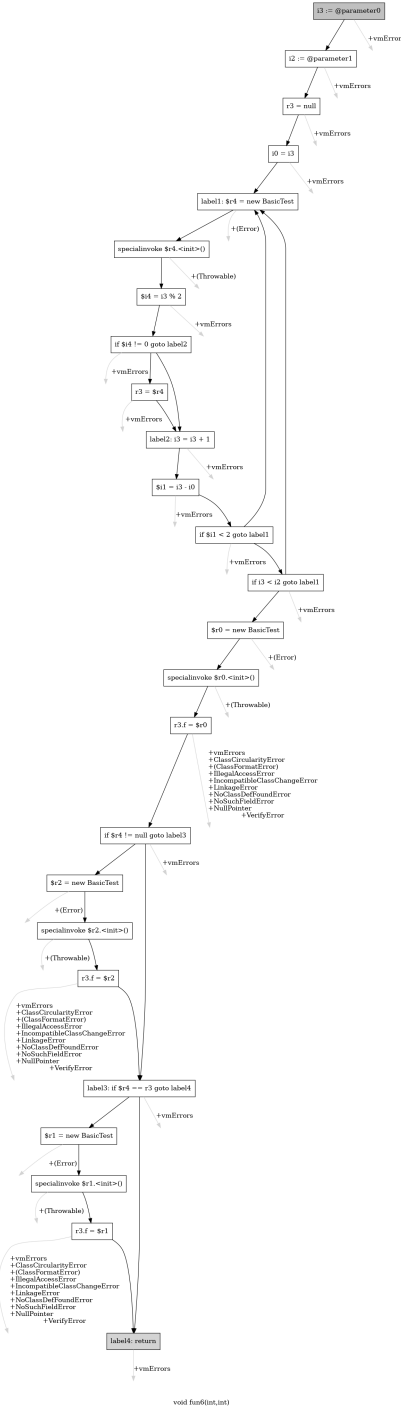


Figure 4: TC-6

BasicTest.fun6: in15: \$r0: {new13}
 BasicTest.fun6: in15: \$r4: {new04}
 BasicTest.fun6: in15: r3: {new04, null}
 BasicTest.fun6: in16: \$r0: {new13}
 BasicTest.fun6: in16: \$r4: {new04}
 BasicTest.fun6: in16: new04.f: {new13}
 BasicTest.fun6: in16: r3: {new04, null}
 BasicTest.fun6: in19: \$r2: {new17}
 BasicTest.fun6: in20: \$r0: {new13}
 BasicTest.fun6: in20: \$r2: {new17}
 BasicTest.fun6: in20: \$r4: {new04}
 BasicTest.fun6: in20: new04.f: {new13}
 BasicTest.fun6: in20: r3: {new04, null}
 BasicTest.fun6: in21: \$r0: {new13}
 BasicTest.fun6: in21: \$r2: {new17}
 BasicTest.fun6: in21: \$r4: {new04}
 BasicTest.fun6: in21: new04.f: {new13}
 BasicTest.fun6: in21: r3: {new04, null}
 BasicTest.fun6: in22: \$r0: {new13}
 BasicTest.fun6: in22: \$r1: {new21}
 BasicTest.fun6: in22: \$r2: {new17}
 BasicTest.fun6: in22: \$r4: {new04}
 BasicTest.fun6: in22: new04.f: {new13}
 BasicTest.fun6: in22: r3: {new04, null}
 BasicTest.fun6: in23: \$r0: {new13}
 BasicTest.fun6: in23: \$r1: {new21}
 BasicTest.fun6: in23: \$r2: {new17}
 BasicTest.fun6: in23: \$r4: {new04}
 BasicTest.fun6: in23: new04.f: {new13}
 BasicTest.fun6: in23: r3: {new04, null}
 BasicTest.fun6: in24: \$r0: {new13}
 BasicTest.fun6: in24: \$r1: {new21}
 BasicTest.fun6: in24: \$r2: {new17}
 BasicTest.fun6: in24: \$r4: {new04}
 BasicTest.fun6: in24: new04.f: {new13, new21}
 BasicTest.fun6: in24: r3: {new04, null}