Phase1 - Public Test Cases

0.1 Test Case 3

0.1.1 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
    }
```

0.1.2 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

0.1.3 CFG

1

0.1.4 Output

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

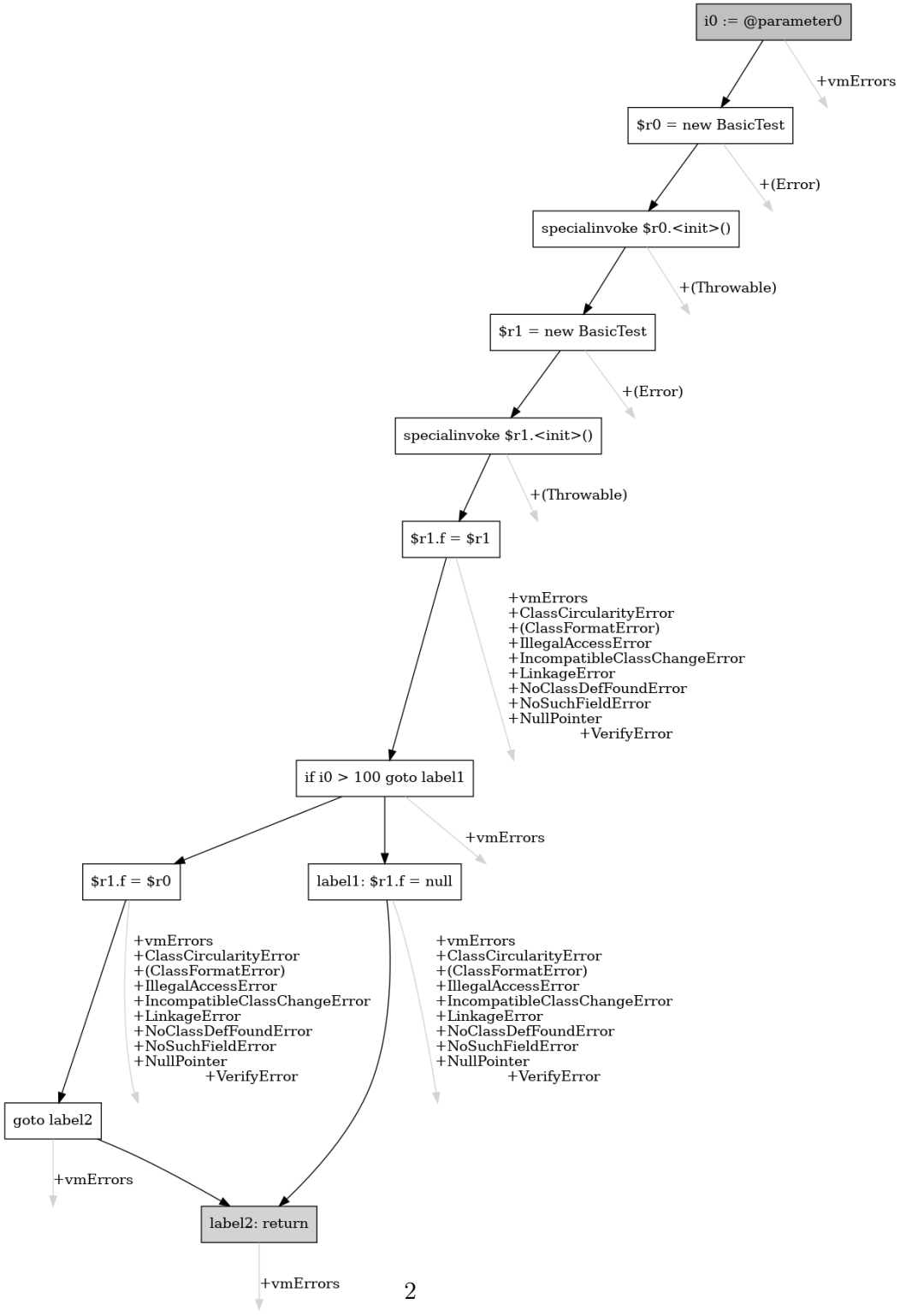
i0 := @parameter0
+vmErrors
$r0 = new BasicTest
+(Error)
specialinvoke $r0.<init>()
+(Throwable)
$r1 = new BasicTest
+(Error)
specialinvoke $r1.<init>()
+(Throwable)
$r1.f = $r1
+vmErrors
+ClassCircularityError
+(ClassFormatError)
+IllegalAccessError
+IncompatibleClassChangeError
+LinkageError
+NoClassDefFoundError
+NoSuchFieldError
+NullPointer
+VerifyError
if i0 > 100 goto label1
+vmErrors
$r1.f = $r0
label1: $r1.f = null
+vmErrors
+ClassCircularityError
+(ClassFormatError)
+IllegalAccessError
+IncompatibleClassChangeError
+LinkageError
+NoClassDefFoundError
+NoSuchFieldError
+NullPointer
+VerifyError
+vmErrors
+ClassCircularityError
+(ClassFormatError)
+IllegalAccessError
+IncompatibleClassChangeError
+LinkageError
+NoClassDefFoundError
+NoSuchFieldError
+NullPointer
+VerifyError
goto label2
+vmErrors
label2: return
+vmErrors
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: {new01, new03}
BasicTest.fun3: in09: $r0: {new01}
BasicTest.fun3: in09: $r1: {new03}
BasicTest.fun3: in09: new03.f: {new03}
BasicTest.fun3: in10: $r0: {new01}
BasicTest.fun3: in10: $r1: {new03}
BasicTest.fun3: in10: new03.f: {new01, new03, null}

0.2 Test Case 4

0.2.1 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;
}
```

0.2.2 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

0.2.3 CFG

2

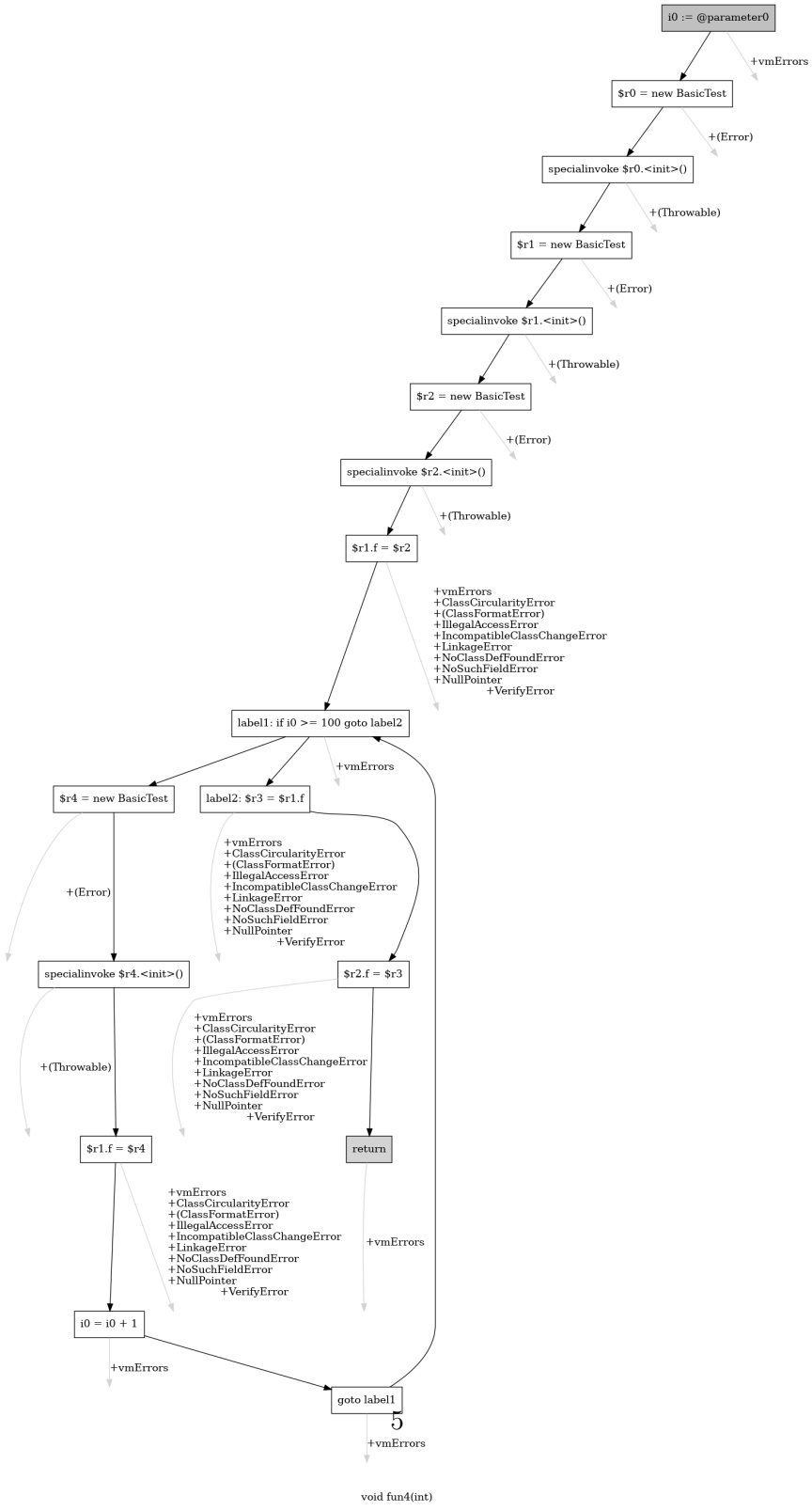
i0 := @parameter0
+vmErrors
$r0 = new BasicTest
+(Error)
specialinvoke $r0.<init>()
+(Throwable)
$r1 = new BasicTest
+(Error)
specialinvoke $r1.<init>()
+(Throwable)
$r2 = new BasicTest
+(Error)
specialinvoke $r2.<init>()
+(Throwable)
$r1.f = $r2
+vmErrors
+ClassCircularityError
+(ClassFormatError)
+IllegalAccessError
+IncompatibleClassChangeError
+LinkageError
+NoClassDefFoundError
+NoSuchFieldError
+NullPointer
+VerifyError
label1: if i0 >= 100 goto label2
+vmErrors
$r4 = new BasicTest
label2: $r3 = $r1.f
+(Error)
+vmErrors
+ClassCircularityError
+(ClassFormatError)
+IllegalAccessError
+IncompatibleClassChangeError
+LinkageError
+NoClassDefFoundError
+NoSuchFieldError
+NullPointer
+VerifyError
specialinvoke $r4.<init>()
$r2.f = $r3
+(Throwable)
+vmErrors
+ClassCircularityError
+(ClassFormatError)
+IllegalAccessError
+IncompatibleClassChangeError
+LinkageError
+NoClassDefFoundError
+NoSuchFieldError
+NullPointer
+VerifyError
$r1.f = $r4
return
+vmErrors
+ClassCircularityError
+(ClassFormatError)
+IllegalAccessError
+IncompatibleClassChangeError
+LinkageError
+NoClassDefFoundError
+NoSuchFieldError
+NullPointer
+VerifyError
+vmErrors
i0 = i0 + 1
+vmErrors
goto label1
+vmErrors
void fun4(int)

Figure 2: TC-4

0.2.4 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}

0.3 Test Case 5

0.3.1 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
}
```

0.3.2 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

0.3.3 CFG

3

0.3.4 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}

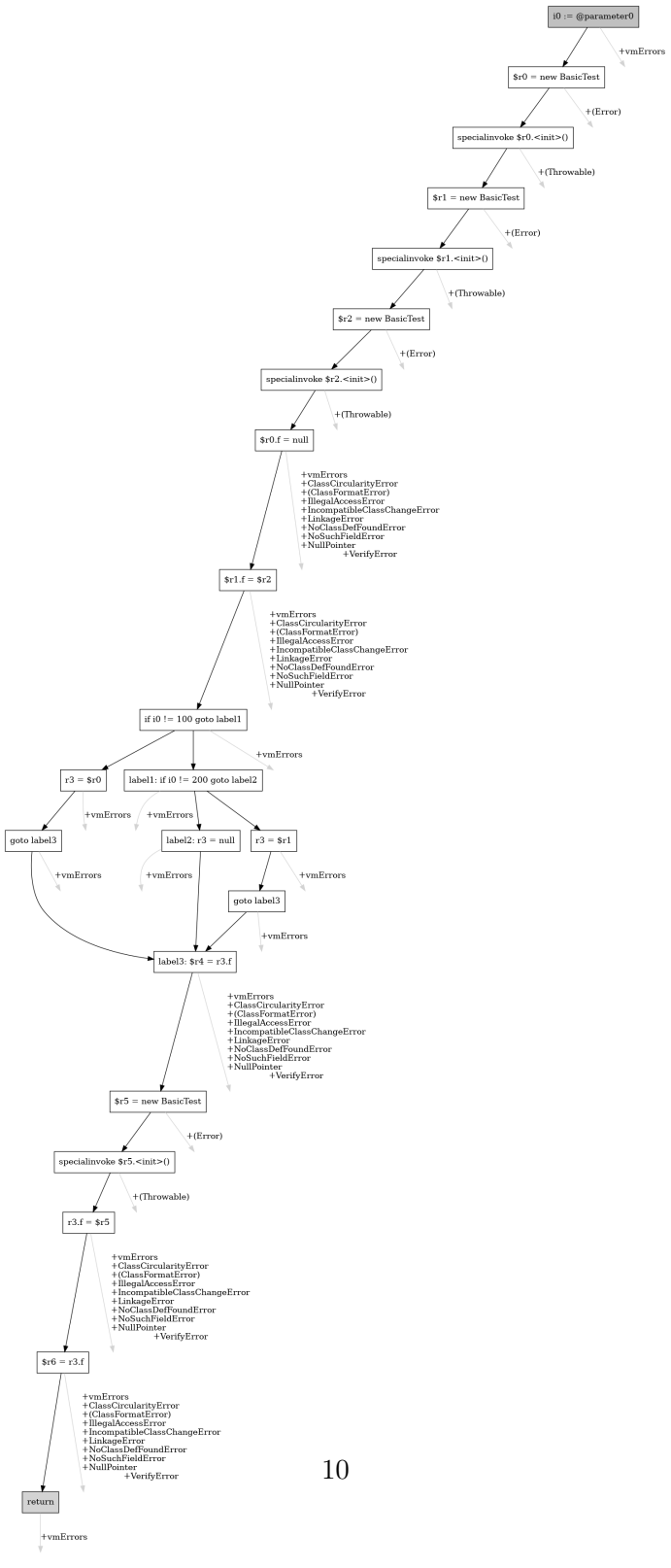
i0 := @parameter0
+vmErrors
$r0 = new BasicTest
+(Error)
specialinvoke $r0.<init>()
+(Throwable)
$r1 = new BasicTest
+(Error)
specialinvoke $r1.<init>()
+(Throwable)
$r2 = new BasicTest
+(Error)
specialinvoke $r2.<init>()
+(Throwable)
$r0.f = null
+vmErrors
+ClassCircularityError
+(ClassFormatError)
+IllegalAccessError
+IncompatibleClassChangeError
+LinkageError
+NoClassDefFoundError
+NoSuchFieldError
+NullPointer
+VerifyError
$r1.f = $r2
+vmErrors
+ClassCircularityError
+(ClassFormatError)
+IllegalAccessError
+IncompatibleClassChangeError
+LinkageError
+NoClassDefFoundError
+NoSuchFieldError
+NullPointer
+VerifyError
if i0 != 100 goto label1
+vmErrors
r3 = $r0
label1: if i0 != 200 goto label2
+vmErrors
+vmErrors
goto label3
label2: r3 = null
r3 = $r1
+vmErrors
+vmErrors
+vmErrors
goto label3
+vmErrors
label3: $r4 = r3.f
+vmErrors
+ClassCircularityError
+(ClassFormatError)
+IllegalAccessError
+IncompatibleClassChangeError
+LinkageError
+NoClassDefFoundError
+NoSuchFieldError
+NullPointer
+VerifyError
$r5 = new BasicTest
+(Error)
specialinvoke $r5.<init>()
+(Throwable)
r3.f = $r5
+vmErrors
+ClassCircularityError
+(ClassFormatError)
+IllegalAccessError
+IncompatibleClassChangeError
+LinkageError
+NoClassDefFoundError
+NoSuchFieldError
+NullPointer
+VerifyError
$r6 = r3.f
+vmErrors
+ClassCircularityError
+(ClassFormatError)
+IllegalAccessError
+IncompatibleClassChangeError
+LinkageError
+NoClassDefFoundError
+NoSuchFieldError
+NullPointer
+VerifyError
return
+vmErrors
void fun5(int)

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: new01.f: {null}
BasicTest.fun5: in18: new03.f: {new05}
BasicTest.fun5: in18: r3: {new01, new03, null}
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}

0.4 Test Case 6

0.4.1 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
}
```

0.4.2 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

0.4.3 CFG

4

0.4.4 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}

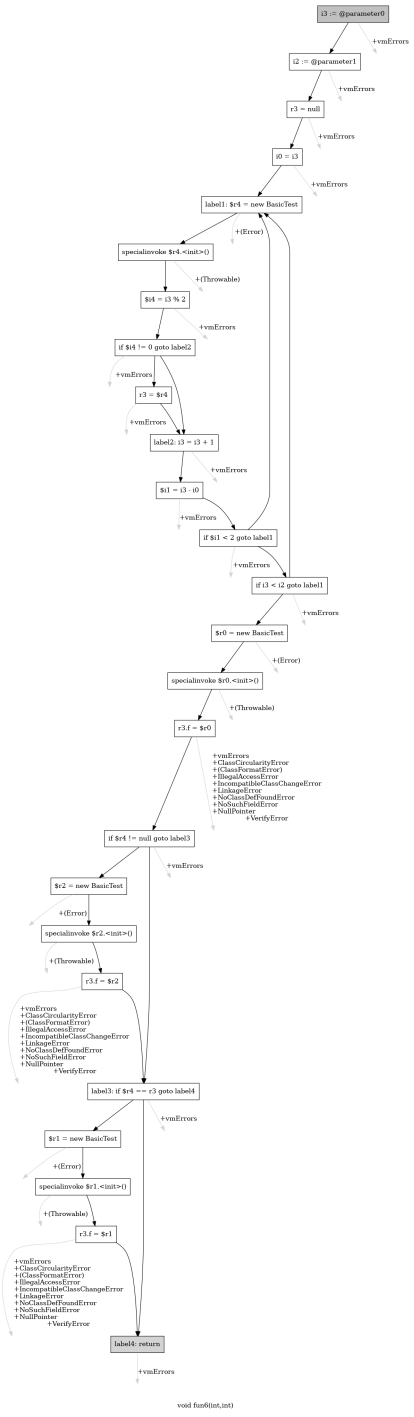
i3 := @parameter0
+vmErrors
i2 := @parameter1
+vmErrors
r3 = null
+vmErrors
i0 = i3
+vmErrors
label1: $r4 = new BasicTest
+(Error)
specialinvoke $r4.<init>()
+(Throwable)
$i4 = i3 % 2
+vmErrors
if $i4 != 0 goto label2
+vmErrors
r3 = $r4
+vmErrors
label2: i3 = i3 + 1
+vmErrors
$i1 = i3 - i0
+vmErrors
if $i1 < 2 goto label1
+vmErrors
if i3 < i2 goto label1
+vmErrors
$r0 = new BasicTest
+(Error)
specialinvoke $r0.<init>()
+(Throwable)
r3.f = $r0
+vmErrors
+ClassCircularityError
+(ClassFormatError)
+IllegalAccessError
+IncompatibleClassChangeError
+LinkageError
+NoClassDefFoundError
+NoSuchFieldError
+NullPointer
+VerifyError
if $r4 != null goto label3
+vmErrors
$r2 = new BasicTest
+(Error)
specialinvoke $r2.<init>()
+(Throwable)
r3.f = $r2
+vmErrors
+ClassCircularityError
+(ClassFormatError)
+IllegalAccessError
+IncompatibleClassChangeError
+LinkageError
+NoClassDefFoundError
+NoSuchFieldError
+NullPointer
+VerifyError
label3: if $r4 == r3 goto label4
+vmErrors
$r1 = new BasicTest
+(Error)
specialinvoke $r1.<init>()
+(Throwable)
r3.f = $r1
+vmErrors
+ClassCircularityError
+(ClassFormatError)
+IllegalAccessError
+IncompatibleClassChangeError
+LinkageError
+NoClassDefFoundError
+NoSuchFieldError
+NullPointer
+VerifyError
label4: return
+vmErrors
void fun6(int,int)

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: in18: $r2: {new17}
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}

Phase-2 TestCase 1

```
public class PubTest
{
PubTest f;

    // treat each test* method below as an entry point, and
    // hence analyze it under ‘@’ (i.e,, epsilon) context

    static void test1() {
        // example 1 starts here
        PubTest k1 = new PubTest();   // first object
        PubTest k2 = new PubTest();   // second object
        k1.f = k2;
        PubTest k3 = new PubTest();   // third object
        foo1(k1,k3);
        // first object’s f points to second object, second object’s
        // f will point to third object here,
        // and third object’s f points to fourth object,
        // and fourth object’s f points to null
    }

    static void foo1(PubTest k1, PubTest k3) {
        // called from a single context. In this context,
        // k1 points to first object, k3 points to
        third object, and first object’s f points to second object
        PubTest k2 = k1.f;
        k2.f = k3;
        PubTest t = new PubTest(); // fourth object
        k3.f = t;
        t.f = null;
    }
}
```

0.5 Intermediate Interpretation

test1

```
00:         $r0 = new PubTest
01:         specialinvoke $r0.<PubTest: void <init>()>()
02:         $r1 = new PubTest
03:         specialinvoke $r1.<PubTest: void <init>()>()
04:         $r0.<PubTest: PubTest f> = $r1
05:         $r2 = new PubTest
06:         specialinvoke $r2.<PubTest: void <init>()>()
07:         staticinvoke <PubTest: void foo1(PubTest,PubTest)>($r0, $r2)
08:         return
```

foo1

```
00:         r0 := @parameter0: PubTest
01:         r2 := @parameter1: PubTest
02:         r1 = r0.<PubTest: PubTest f>
03:         r1.<PubTest: PubTest f> = r2
04:         $r3 = new PubTest
05:         specialinvoke $r3.<PubTest: void <init>()>()
06:         r2.<PubTest: PubTest f> = $r3
07:         $r3.<PubTest: PubTest f> = null
08:         return
```

0.6 CFG

5 6

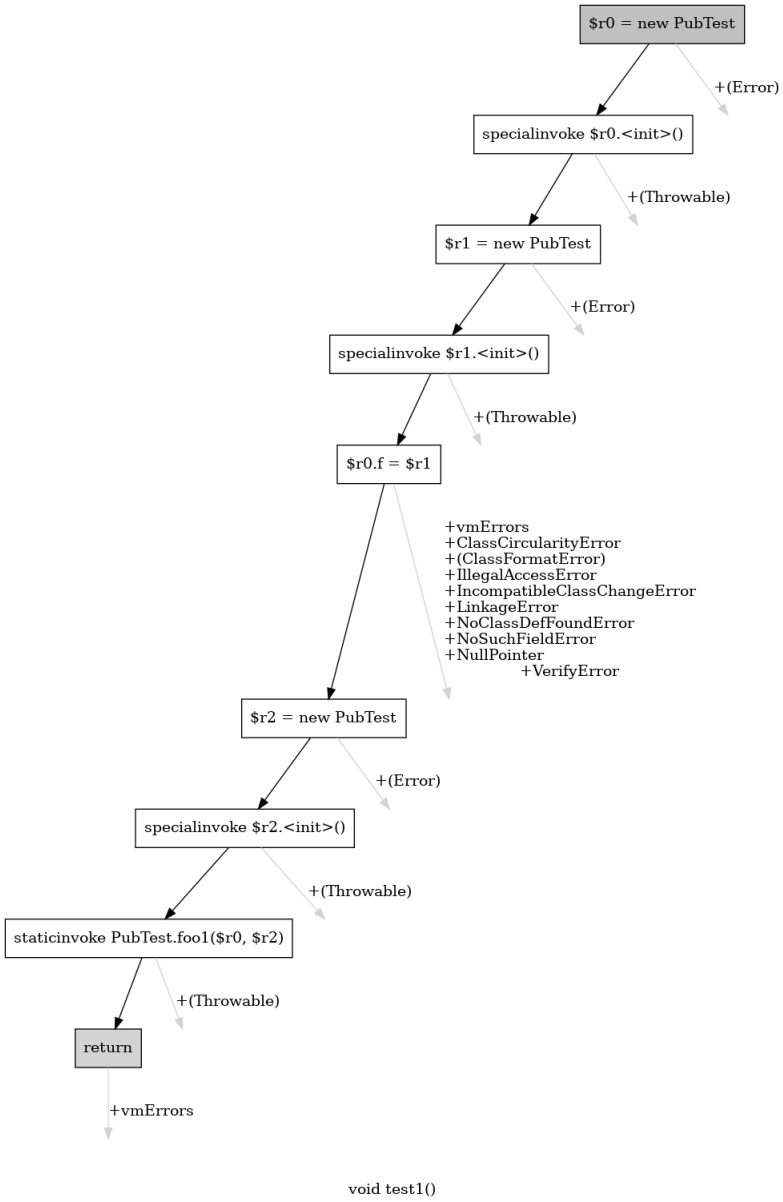

Figure 5: CFG for test1

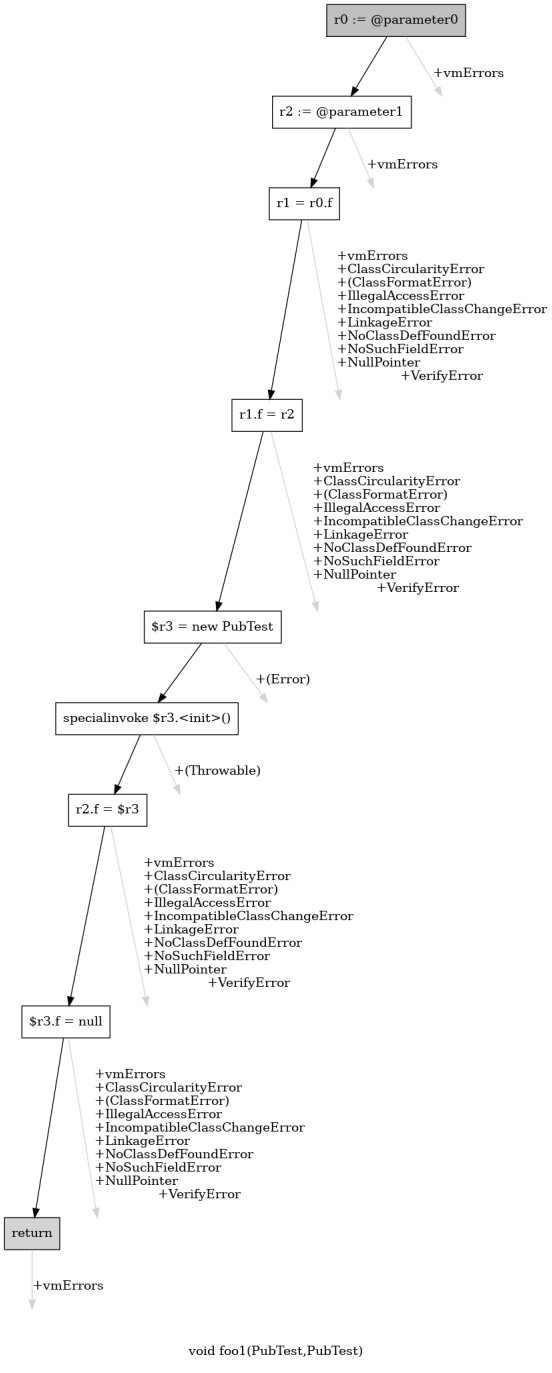

Figure 6: CFG for foo1

0.6.1 Output

```
PubTest.foo1: in01: [PubTest.test1.in07] => r0: {test1.new00}
PubTest.foo1: in01: [PubTest.test1.in07] => test1.new00.f: {test1.new02}
PubTest.foo1: in02: [PubTest.test1.in07] => r0: {test1.new00}
PubTest.foo1: in02: [PubTest.test1.in07] => r2: {test1.new05}
PubTest.foo1: in02: [PubTest.test1.in07] => test1.new00.f: {test1.new02}
PubTest.foo1: in03: [PubTest.test1.in07] => r0: {test1.new00}
PubTest.foo1: in03: [PubTest.test1.in07] => r1: {test1.new02}
PubTest.foo1: in03: [PubTest.test1.in07] => r2: {test1.new05}
PubTest.foo1: in03: [PubTest.test1.in07] => test1.new00.f: {test1.new02}
PubTest.foo1: in04: [PubTest.test1.in07] => r0: {test1.new00}
PubTest.foo1: in04: [PubTest.test1.in07] => r1: {test1.new02}
PubTest.foo1: in04: [PubTest.test1.in07] => r2: {test1.new05}
PubTest.foo1: in04: [PubTest.test1.in07] => test1.new00.f: {test1.new02}
PubTest.foo1: in04: [PubTest.test1.in07] => test1.new02.f: {test1.new05}
PubTest.foo1: in05: [PubTest.test1.in07] => $r3: {foo1.new04}
PubTest.foo1: in05: [PubTest.test1.in07] => r0: {test1.new00}
PubTest.foo1: in05: [PubTest.test1.in07] => r1: {test1.new02}
PubTest.foo1: in05: [PubTest.test1.in07] => r2: {test1.new05}
PubTest.foo1: in05: [PubTest.test1.in07] => test1.new00.f: {test1.new02}
PubTest.foo1: in05: [PubTest.test1.in07] => test1.new02.f: {test1.new05}
PubTest.foo1: in06: [PubTest.test1.in07] => $r3: {foo1.new04}
PubTest.foo1: in06: [PubTest.test1.in07] => r0: {test1.new00}
PubTest.foo1: in06: [PubTest.test1.in07] => r1: {test1.new02}
PubTest.foo1: in06: [PubTest.test1.in07] => r2: {test1.new05}
PubTest.foo1: in06: [PubTest.test1.in07] => test1.new00.f: {test1.new02}
PubTest.foo1: in06: [PubTest.test1.in07] => test1.new02.f: {test1.new05}
PubTest.foo1: in07: [PubTest.test1.in07] => $r3: {foo1.new04}
PubTest.foo1: in07: [PubTest.test1.in07] => r0: {test1.new00}
PubTest.foo1: in07: [PubTest.test1.in07] => r1: {test1.new02}
PubTest.foo1: in07: [PubTest.test1.in07] => r2: {test1.new05}
PubTest.foo1: in07: [PubTest.test1.in07] => test1.new00.f: {test1.new02}
PubTest.foo1: in07: [PubTest.test1.in07] => test1.new02.f: {test1.new05}
PubTest.foo1: in07: [PubTest.test1.in07] => test1.new05.f: {foo1.new04}
PubTest.foo1: in08: [PubTest.test1.in07] => $r3: {foo1.new04}
PubTest.foo1: in08: [PubTest.test1.in07] => foo1.new04.f: {null}
PubTest.foo1: in08: [PubTest.test1.in07] => r0: {test1.new00}
PubTest.foo1: in08: [PubTest.test1.in07] => r1: {test1.new02}
PubTest.foo1: in08: [PubTest.test1.in07] => r2: {test1.new05}
```

```
PubTest.foo1: in08: [PubTest.test1.in07] => test1.new00.f: {test1.new02}
PubTest.foo1: in08: [PubTest.test1.in07] => test1.new02.f: {test1.new05}
PubTest.foo1: in08: [PubTest.test1.in07] => test1.new05.f: {foo1.new04}
PubTest.test1: in01: @ => $r0: {test1.new00}
PubTest.test1: in02: @ => $r0: {test1.new00}
PubTest.test1: in03: @ => $r0: {test1.new00}
PubTest.test1: in03: @ => $r1: {test1.new02}
PubTest.test1: in04: @ => $r0: {test1.new00}
PubTest.test1: in04: @ => $r1: {test1.new02}
PubTest.test1: in05: @ => $r0: {test1.new00}
PubTest.test1: in05: @ => $r1: {test1.new02}
PubTest.test1: in05: @ => test1.new00.f: {test1.new02}
PubTest.test1: in06: @ => $r0: {test1.new00}
PubTest.test1: in06: @ => $r1: {test1.new02}
PubTest.test1: in06: @ => $r2: {test1.new05}
PubTest.test1: in06: @ => test1.new00.f: {test1.new02}
PubTest.test1: in07: @ => $r0: {test1.new00}
PubTest.test1: in07: @ => $r1: {test1.new02}
PubTest.test1: in07: @ => $r2: {test1.new05}
PubTest.test1: in07: @ => test1.new00.f: {test1.new02}
PubTest.test1: in08: @ => $r0: {test1.new00}
PubTest.test1: in08: @ => $r1: {test1.new02}
PubTest.test1: in08: @ => $r2: {test1.new05}
PubTest.test1: in08: @ => test1.new00.f: {test1.new02}
PubTest.test1: in08: @ => test1.new02.f: {test1.new05}
PubTest.test1: in08: @ => test1.new05.f: {foo1.new04}
PubTest.test1: in08: @ => foo1.new04.f: {null}
```

1 Phase-2 TestCase 2

1.1 Intermediate Representation

test2

```
00:         $r0 = new PubTest
01:         specialinvoke $r0.<PubTest: void <init>()>()
02:         staticinvoke <PubTest: PubTest foo2(PubTest)>($r0)
03:         $r1 = new PubTest
04:         specialinvoke $r1.<PubTest: void <init>()>()
05:         staticinvoke <PubTest: PubTest foo2(PubTest)>($r1)
```

```
06:         return
```

foo2

```
00:         r0 := @parameter0: PubTest
01:         $r1 = staticinvoke <PubTest: PubTest bar2(PubTest)>(r0)
02:         return $r1
```

bar2

```
00:         r0 := @parameter0: PubTest
01:         $r1 = staticinvoke <PubTest: PubTest bar3(PubTest)>(r0)
02:         return $r1
```

bar3

```
00:         r0 := @parameter0: PubTest
01:         return r0
```

1.2 CFG

7 8 9 10

1.3 Output

2 Phase-2 TestCase 3

2.1 Intermediate Representation

test3

```
00:         $r0 = new PubTest
01:         specialinvoke $r0.<PubTest: void <init>()>()
02:         staticinvoke <PubTest: PubTest baz2(PubTest)>($r0)
03:         $r1 = new PubTest
04:         specialinvoke $r1.<PubTest: void <init>()>()
05:         staticinvoke <PubTest: PubTest baz2(PubTest)>($r1)
06:         return
```

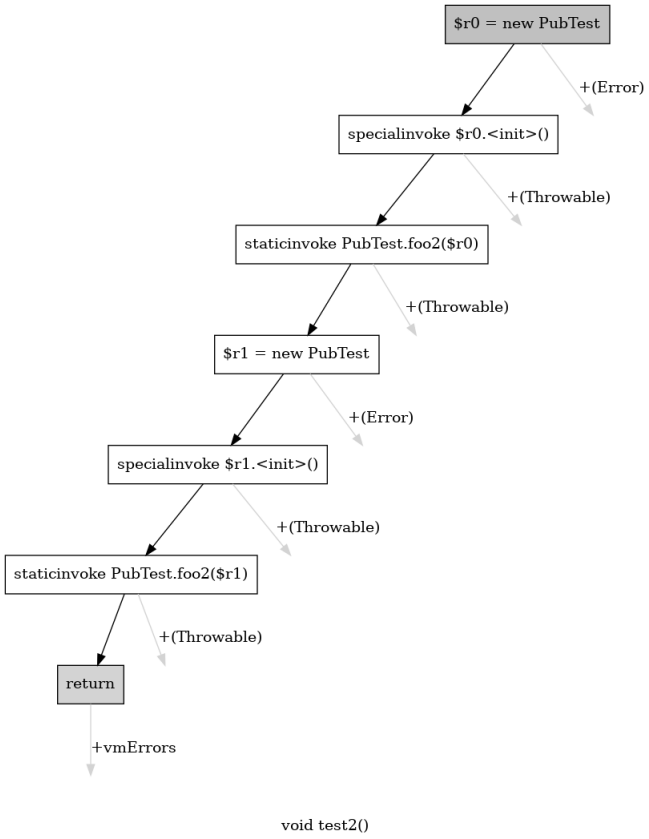

Figure 7: CFG for test2

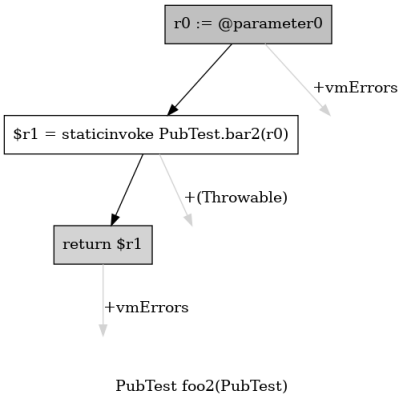

Figure 8: CFG for foo2

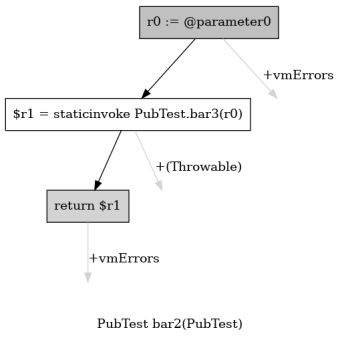

Figure 9: CFG for bar2

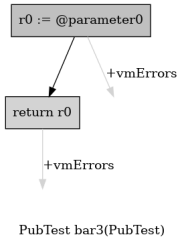

Figure 10: CFG for bar3

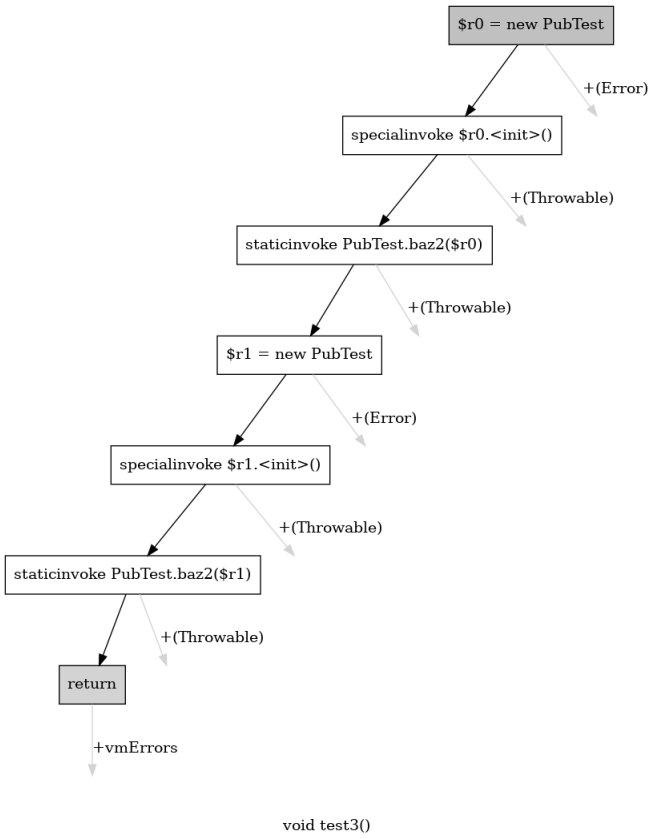

Figure 11: CFG for test3

baz2

```
00:         r0 := @parameter0: PubTest
01:         $r1 = staticinvoke <PubTest: PubTest baz3(PubTest)>(r0)
02:         return $r1
```

baz3

```
00:         r0 := @parameter0: PubTest
01:         return r0
```

2.2 CFG

11 12 13

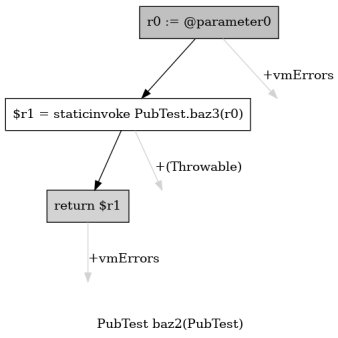

Figure 12: CFG for baz2

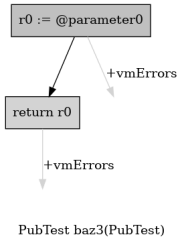

Figure 13: CFG for baz3

2.3 Output