```
TEST CASE test001: Unary Negation
     STUDENT:
--------------------------------------------------------------------------
Class name: In
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 1, scan_int, None, public, static, int
Method Parameters:
Variable Table:
Method Body:
Block([])
METHOD: 2, scan_float, None, public, static, float
Method Parameters:
Variable Table:
Method Body:
Block([])
--------------------------------------------------------------------------
Class name: Out
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 3, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, i, formal, int
Method Body:
Block([])
METHOD: 4, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, f, formal, float
Method Body:
Block([])
METHOD: 5, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, b, formal, boolean
Method Body:
Block([])
METHOD: 6, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, s, formal, string
Method Body:
Block([])
--------------------------------------------------------------------------
Class name: test001
Superclass Name:
Fields:
```

```
Constructors:
Methods:
METHOD: 7, main, test001, public, static, void
Method Parameters:
Variable Table:
VARIABLE 1, a, local, boolean
VARIABLE 2, b, local, boolean
Method Body:
Block([
Expr(Assign(Variable(1), Constant(True)))
, Expr(Assign(Variable(2), Unnary(Variable(-1), neg)))
])
---------------------------------------------------------------------------


     STUDENT ERROR:

     SHOULD BE: Check print out of AST


Press Enter to continue...
```

```
TEST CASE test002: Unary Minus

     STUDENT:
--------------------------------------------------------------------------
Class name: In
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 1, scan_int, None, public, static, int
Method Parameters:
Variable Table:
Method Body:
Block([])
METHOD: 2, scan_float, None, public, static, float
Method Parameters:
Variable Table:
Method Body:
Block([])
--------------------------------------------------------------------------
Class name: Out
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 3, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, i, formal, int
Method Body:
Block([])
METHOD: 4, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, f, formal, float
Method Body:
Block([])
METHOD: 5, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, b, formal, boolean
Method Body:
Block([])
METHOD: 6, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, s, formal, string
Method Body:
Block([])
--------------------------------------------------------------------------
Class name: test002
Superclass Name:
```

```
Fields:
Constructors:
Methods:
METHOD: 7, main, test002, public, static, void
Method Parameters:
Variable Table:
VARIABLE 1, a, local, int
VARIABLE 2, b, local, int
Method Body:
Block([
Expr(Assign(Variable(1), Constant(Integer-constant(17))))
, Expr(Assign(Variable(2), Unnary(Variable(-1), uminus)))
])
--------------------------------------------------------------------------


     STUDENT ERROR:

     SHOULD BE: Check print out of AST


Press Enter to continue...
```

```
TEST CASE test003: Unary Plus

     STUDENT:
--------------------------------------------------------------------------
Class name: In
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 1, scan_int, None, public, static, int
Method Parameters:
Variable Table:
Method Body:
Block([])
METHOD: 2, scan_float, None, public, static, float
Method Parameters:
Variable Table:
Method Body:
Block([])
--------------------------------------------------------------------------
Class name: Out
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 3, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, i, formal, int
Method Body:
Block([])
METHOD: 4, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, f, formal, float
Method Body:
Block([])
METHOD: 5, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, b, formal, boolean
Method Body:
Block([])
METHOD: 6, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, s, formal, string
Method Body:
Block([])
--------------------------------------------------------------------------
Class name: test003
Superclass Name:
```

```
Fields:
Constructors:
Methods:
METHOD: 7, main, test003, public, static, void
Method Parameters:
Variable Table:
VARIABLE 1, a, local, int
VARIABLE 2, b, local, int
Method Body:
Block([
Expr(Assign(Variable(1), Constant(Integer-constant(19))))
, Expr(Assign(Variable(2), Variable(1)))
])
--------------------------------------------------------------------------


     STUDENT ERROR:

     SHOULD BE: Check print out of AST


Press Enter to continue...
```

```
TEST CASE test004: Binary GTE

     STUDENT:
--------------------------------------------------------------------------
Class name: In
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 1, scan_int, None, public, static, int
Method Parameters:
Variable Table:
Method Body:
Block([])
METHOD: 2, scan_float, None, public, static, float
Method Parameters:
Variable Table:
Method Body:
Block([])
--------------------------------------------------------------------------
Class name: Out
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 3, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, i, formal, int
Method Body:
Block([])
METHOD: 4, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, f, formal, float
Method Body:
Block([])
METHOD: 5, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, b, formal, boolean
Method Body:
Block([])
METHOD: 6, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, s, formal, string
Method Body:
Block([])
--------------------------------------------------------------------------
Class name: test004
Superclass Name:
```

```
Fields:
Constructors:
Methods:
METHOD: 7, main, test004, public, static, void
Method Parameters:
Variable Table:
VARIABLE 1, a, local, boolean
Method Body:
Block([
Expr(Assign(Variable(1), Binary(geq, Constant(Integer-constant(17)),
Constant(Integer-constant(14)))))
])
--------------------------------------------------------------------------


     STUDENT ERROR:

     SHOULD BE: Check print out of AST


Press Enter to continue...
```

```
TEST CASE test005: Binary GT

     STUDENT:
---------------------------------------------------------------------------
Class name: In
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 1, scan_int, None, public, static, int
Method Parameters:
Variable Table:
Method Body:
Block([])
METHOD: 2, scan_float, None, public, static, float
Method Parameters:
Variable Table:
Method Body:
Block([])
---------------------------------------------------------------------------
Class name: Out
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 3, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, i, formal, int
Method Body:
Block([])
METHOD: 4, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, f, formal, float
Method Body:
Block([])
METHOD: 5, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, b, formal, boolean
Method Body:
Block([])
METHOD: 6, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, s, formal, string
Method Body:
Block([])
---------------------------------------------------------------------------
Class name: test005
Superclass Name:
```

```
Fields:
Constructors:
Methods:
METHOD: 7, main, test005, public, static, void
Method Parameters:
Variable Table:
VARIABLE 1, a, local, boolean
Method Body:
Block([
Expr(Assign(Variable(1), Binary(gt, Constant(Integer-constant(17)),
Constant(Integer-constant(14)))))
])
---------------------------------------------------------------------------


     STUDENT ERROR:

     SHOULD BE: Check print out of AST


Press Enter to continue...
```

```
TEST CASE test006: Binary LTE

     STUDENT:
--------------------------------------------------------------------------
Class name: In
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 1, scan_int, None, public, static, int
Method Parameters:
Variable Table:
Method Body:
Block([])
METHOD: 2, scan_float, None, public, static, float
Method Parameters:
Variable Table:
Method Body:
Block([])
--------------------------------------------------------------------------
Class name: Out
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 3, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, i, formal, int
Method Body:
Block([])
METHOD: 4, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, f, formal, float
Method Body:
Block([])
METHOD: 5, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, b, formal, boolean
Method Body:
Block([])
METHOD: 6, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, s, formal, string
Method Body:
Block([])
--------------------------------------------------------------------------
Class name: test006
Superclass Name:
```

```
Fields:
Constructors:
Methods:
METHOD: 7, main, test006, public, static, void
Method Parameters:
Variable Table:
VARIABLE 1, a, local, boolean
Method Body:
Block([
Expr(Assign(Variable(1), Binary(leq, Constant(Integer-constant(17)),
Constant(Integer-constant(14)))))
])
---------------------------------------------------------------------------


     STUDENT ERROR:

     SHOULD BE: Check print out of AST


Press Enter to continue...
```

```
TEST CASE test007: Binary LT

     STUDENT:
--------------------------------------------------------------------------
Class name: In
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 1, scan_int, None, public, static, int
Method Parameters:
Variable Table:
Method Body:
Block([])
METHOD: 2, scan_float, None, public, static, float
Method Parameters:
Variable Table:
Method Body:
Block([])
--------------------------------------------------------------------------
Class name: Out
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 3, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, i, formal, int
Method Body:
Block([])
METHOD: 4, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, f, formal, float
Method Body:
Block([])
METHOD: 5, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, b, formal, boolean
Method Body:
Block([])
METHOD: 6, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, s, formal, string
Method Body:
Block([])
--------------------------------------------------------------------------
Class name: test007
Superclass Name:
```

```
Fields:
Constructors:
Methods:
METHOD: 7, main, test007, public, static, void
Method Parameters:
Variable Table:
VARIABLE 1, a, local, boolean
Method Body:
Block([
Expr(Assign(Variable(1), Binary(lt, Constant(Integer-constant(17)),
Constant(Integer-constant(14)))))
])
--------------------------------------------------------------------------


     STUDENT ERROR:

     SHOULD BE: Check print out of AST


Press Enter to continue...
```

```
TEST CASE test008: Binary NEQ

     STUDENT:
--------------------------------------------------------------------------
Class name: In
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 1, scan_int, None, public, static, int
Method Parameters:
Variable Table:
Method Body:
Block([])
METHOD: 2, scan_float, None, public, static, float
Method Parameters:
Variable Table:
Method Body:
Block([])
--------------------------------------------------------------------------
Class name: Out
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 3, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, i, formal, int
Method Body:
Block([])
METHOD: 4, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, f, formal, float
Method Body:
Block([])
METHOD: 5, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, b, formal, boolean
Method Body:
Block([])
METHOD: 6, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, s, formal, string
Method Body:
Block([])
--------------------------------------------------------------------------
Class name: test008
Superclass Name:
```

```
Fields:
Constructors:
Methods:
METHOD: 7, main, test008, public, static, void
Method Parameters:
Variable Table:
VARIABLE 1, a, local, boolean
Method Body:
Block([
Expr(Assign(Variable(1), Binary(neq, Constant(Integer-constant(17)),
Constant(Integer-constant(14)))))
])
--------------------------------------------------------------------------


     STUDENT ERROR:

     SHOULD BE: Check print out of AST


Press Enter to continue...
```

```
TEST CASE test009: Binary EQ

     STUDENT:
---------------------------------------------------------------------------
Class name: In
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 1, scan_int, None, public, static, int
Method Parameters:
Variable Table:
Method Body:
Block([])
METHOD: 2, scan_float, None, public, static, float
Method Parameters:
Variable Table:
Method Body:
Block([])
---------------------------------------------------------------------------
Class name: Out
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 3, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, i, formal, int
Method Body:
Block([])
METHOD: 4, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, f, formal, float
Method Body:
Block([])
METHOD: 5, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, b, formal, boolean
Method Body:
Block([])
METHOD: 6, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, s, formal, string
Method Body:
Block([])
---------------------------------------------------------------------------
Class name: test009
Superclass Name:
```

```
Fields:
Constructors:
Methods:
METHOD: 7, main, test009, public, static, void
Method Parameters:
Variable Table:
VARIABLE 1, a, local, boolean
Method Body:
Block([
Expr(Assign(Variable(1), Binary(eq, Constant(Integer-constant(17)),
Constant(Integer-constant(14)))))
])
--------------------------------------------------------------------------


     STUDENT ERROR:

     SHOULD BE: Check print out of AST


Press Enter to continue...
```

```
TEST CASE test010: Binary OR

     STUDENT:
--------------------------------------------------------------------------
Class name: In
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 1, scan_int, None, public, static, int
Method Parameters:
Variable Table:
Method Body:
Block([])
METHOD: 2, scan_float, None, public, static, float
Method Parameters:
Variable Table:
Method Body:
Block([])
--------------------------------------------------------------------------
Class name: Out
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 3, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, i, formal, int
Method Body:
Block([])
METHOD: 4, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, f, formal, float
Method Body:
Block([])
METHOD: 5, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, b, formal, boolean
Method Body:
Block([])
METHOD: 6, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, s, formal, string
Method Body:
Block([])
--------------------------------------------------------------------------
Class name: test010
Superclass Name:
```

```
Fields:
Constructors:
Methods:
METHOD: 7, main, test010, public, static, void
Method Parameters:
Variable Table:
VARIABLE 1, a, local, boolean
Method Body:
Block([
Expr(Assign(Variable(1), Binary(or, Constant(True), Constant(False))))
])
--------------------------------------------------------------------------


     STUDENT ERROR:

     SHOULD BE: Check print out of AST


Press Enter to continue...
```

```
TEST CASE test011: Binary AND

     STUDENT:
--------------------------------------------------------------------------
Class name: In
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 1, scan_int, None, public, static, int
Method Parameters:
Variable Table:
Method Body:
Block([])
METHOD: 2, scan_float, None, public, static, float
Method Parameters:
Variable Table:
Method Body:
Block([])
--------------------------------------------------------------------------
Class name: Out
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 3, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, i, formal, int
Method Body:
Block([])
METHOD: 4, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, f, formal, float
Method Body:
Block([])
METHOD: 5, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, b, formal, boolean
Method Body:
Block([])
METHOD: 6, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, s, formal, string
Method Body:
Block([])
--------------------------------------------------------------------------
Class name: test011
Superclass Name:
```

```
Fields:
Constructors:
Methods:
METHOD: 7, main, test011, public, static, void
Method Parameters:
Variable Table:
VARIABLE 1, a, local, boolean
Method Body:
Block([
Expr(Assign(Variable(1), Binary(and, Constant(True), Constant(False))))
])
--------------------------------------------------------------------------


     STUDENT ERROR:

     SHOULD BE: Check print out of AST


Press Enter to continue...
```

```
TEST CASE test012: Binary DIVISION

     STUDENT:
--------------------------------------------------------------------------
Class name: In
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 1, scan_int, None, public, static, int
Method Parameters:
Variable Table:
Method Body:
Block([])
METHOD: 2, scan_float, None, public, static, float
Method Parameters:
Variable Table:
Method Body:
Block([])
--------------------------------------------------------------------------
Class name: Out
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 3, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, i, formal, int
Method Body:
Block([])
METHOD: 4, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, f, formal, float
Method Body:
Block([])
METHOD: 5, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, b, formal, boolean
Method Body:
Block([])
METHOD: 6, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, s, formal, string
Method Body:
Block([])
--------------------------------------------------------------------------
Class name: test012
Superclass Name:
```

```
Fields:
Constructors:
Methods:
METHOD: 7, main, test012, public, static, void
Method Parameters:
Variable Table:
VARIABLE 1, a, local, int
Method Body:
Block([
Expr(Assign(Variable(1), Binary(div, Constant(Integer-constant(18)),
Constant(Integer-constant(6)))))
])
--------------------------------------------------------------------------


     STUDENT ERROR:

     SHOULD BE: Check print out of AST


Press Enter to continue...
```

```
TEST CASE test013: Binary MULTIPLICATION

     STUDENT:
--------------------------------------------------------------------------
Class name: In
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 1, scan_int, None, public, static, int
Method Parameters:
Variable Table:
Method Body:
Block([])
METHOD: 2, scan_float, None, public, static, float
Method Parameters:
Variable Table:
Method Body:
Block([])
--------------------------------------------------------------------------
Class name: Out
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 3, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, i, formal, int
Method Body:
Block([])
METHOD: 4, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, f, formal, float
Method Body:
Block([])
METHOD: 5, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, b, formal, boolean
Method Body:
Block([])
METHOD: 6, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, s, formal, string
Method Body:
Block([])
--------------------------------------------------------------------------
Class name: test013
Superclass Name:
```

```
Fields:
Constructors:
Methods:
METHOD: 7, main, test013, public, static, void
Method Parameters:
Variable Table:
VARIABLE 1, a, local, int
Method Body:
Block([
Expr(Assign(Variable(1), Binary(mul, Constant(Integer-constant(18)),
Constant(Integer-constant(6)))))
])
---------------------------------------------------------------------------


     STUDENT ERROR:

     SHOULD BE: Check print out of AST


Press Enter to continue...
```

```
TEST CASE test014: Binary SUBTRACTION

     STUDENT:
--------------------------------------------------------------------------
Class name: In
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 1, scan_int, None, public, static, int
Method Parameters:
Variable Table:
Method Body:
Block([])
METHOD: 2, scan_float, None, public, static, float
Method Parameters:
Variable Table:
Method Body:
Block([])
--------------------------------------------------------------------------
Class name: Out
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 3, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, i, formal, int
Method Body:
Block([])
METHOD: 4, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, f, formal, float
Method Body:
Block([])
METHOD: 5, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, b, formal, boolean
Method Body:
Block([])
METHOD: 6, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, s, formal, string
Method Body:
Block([])
--------------------------------------------------------------------------
Class name: test014
Superclass Name:
```

```
Fields:
Constructors:
Methods:
METHOD: 7, main, test014, public, static, void
Method Parameters:
Variable Table:
VARIABLE 1, a, local, int
Method Body:
Block([
Expr(Assign(Variable(1), Binary(sub, Constant(Integer-constant(18)),
Constant(Integer-constant(6)))))
])
--------------------------------------------------------------------------


     STUDENT ERROR:

     SHOULD BE: Check print out of AST


Press Enter to continue...
```

```
TEST CASE test015: Binary ADDITION

     STUDENT:
--------------------------------------------------------------------------
Class name: In
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 1, scan_int, None, public, static, int
Method Parameters:
Variable Table:
Method Body:
Block([])
METHOD: 2, scan_float, None, public, static, float
Method Parameters:
Variable Table:
Method Body:
Block([])
--------------------------------------------------------------------------
Class name: Out
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 3, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, i, formal, int
Method Body:
Block([])
METHOD: 4, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, f, formal, float
Method Body:
Block([])
METHOD: 5, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, b, formal, boolean
Method Body:
Block([])
METHOD: 6, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, s, formal, string
Method Body:
Block([])
--------------------------------------------------------------------------
Class name: test015
Superclass Name:
```

```
Fields:
Constructors:
Methods:
METHOD: 7, main, test015, public, static, void
Method Parameters:
Variable Table:
VARIABLE 1, a, local, int
Method Body:
Block([
Expr(Assign(Variable(1), Binary(add, Constant(Integer-constant(18)),
Constant(Integer-constant(6)))))
])
---------------------------------------------------------------------------


     STUDENT ERROR:

     SHOULD BE: Check print out of AST


Press Enter to continue...
```

TEST CASE test016: INTEGER CONST Declaration and Assignment

```
     STUDENT:
--------------------------------------------------------------------------
Class name: In
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 1, scan_int, None, public, static, int
Method Parameters:
Variable Table:
Method Body:
Block([])
METHOD: 2, scan_float, None, public, static, float
Method Parameters:
Variable Table:
Method Body:
Block([])
--------------------------------------------------------------------------
Class name: Out
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 3, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, i, formal, int
Method Body:
Block([])
METHOD: 4, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, f, formal, float
Method Body:
Block([])
METHOD: 5, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, b, formal, boolean
Method Body:
Block([])
METHOD: 6, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, s, formal, string
Method Body:
Block([])
--------------------------------------------------------------------------
Class name: test016
Superclass Name:
```

```
Fields:
Constructors:
Methods:
METHOD: 7, main, test016, public, static, void
Method Parameters:
Variable Table:
VARIABLE 1, a, local, int
Method Body:
Block([
Expr(Assign(Variable(1), Constant(Integer-constant(18))))
])
---------------------------------------------------------------------------


     STUDENT ERROR:

     SHOULD BE: Check print out of AST


Press Enter to continue...
```

TEST CASE test017: FLOAT CONST Declaration and Assignment

```
     STUDENT:
---------------------------------------------------------------------------
Class name: In
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 1, scan_int, None, public, static, int
Method Parameters:
Variable Table:
Method Body:
Block([])
METHOD: 2, scan_float, None, public, static, float
Method Parameters:
Variable Table:
Method Body:
Block([])
---------------------------------------------------------------------------
Class name: Out
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 3, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, i, formal, int
Method Body:
Block([])
METHOD: 4, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, f, formal, float
Method Body:
Block([])
METHOD: 5, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, b, formal, boolean
Method Body:
Block([])
METHOD: 6, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, s, formal, string
Method Body:
Block([])
---------------------------------------------------------------------------
Class name: test017
Superclass Name:
```

```
Fields:
Constructors:
Methods:
METHOD: 7, main, test017, public, static, void
Method Parameters:
Variable Table:
VARIABLE 1, a, local, float
Method Body:
Block([
Expr(Assign(Variable(1), Constant(Float-constant(17.4))))
])
--------------------------------------------------------------------------


     STUDENT ERROR:

     SHOULD BE: Check print out of AST


Press Enter to continue...
```

```
TEST CASE test018: BOOLEAN TRUE CONST Declaration and Assignment

     STUDENT:
--------------------------------------------------------------------------
Class name: In
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 1, scan_int, None, public, static, int
Method Parameters:
Variable Table:
Method Body:
Block([])
METHOD: 2, scan_float, None, public, static, float
Method Parameters:
Variable Table:
Method Body:
Block([])
--------------------------------------------------------------------------
Class name: Out
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 3, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, i, formal, int
Method Body:
Block([])
METHOD: 4, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, f, formal, float
Method Body:
Block([])
METHOD: 5, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, b, formal, boolean
Method Body:
Block([])
METHOD: 6, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, s, formal, string
Method Body:
Block([])
--------------------------------------------------------------------------
Class name: test018
Superclass Name:
```

```
Fields:
Constructors:
Methods:
METHOD: 7, main, test018, public, static, void
Method Parameters:
Variable Table:
VARIABLE 1, a, local, boolean
Method Body:
Block([
Expr(Assign(Variable(1), Constant(True)))
])
---------------------------------------------------------------------------


     STUDENT ERROR:

     SHOULD BE: Check print out of AST


Press Enter to continue...
```

```
TEST CASE test019: BOOLEAN FALSE CONST Declaration and Assignment

     STUDENT:
---------------------------------------------------------------------------
Class name: In
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 1, scan_int, None, public, static, int
Method Parameters:
Variable Table:
Method Body:
Block([])
METHOD: 2, scan_float, None, public, static, float
Method Parameters:
Variable Table:
Method Body:
Block([])
---------------------------------------------------------------------------
Class name: Out
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 3, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, i, formal, int
Method Body:
Block([])
METHOD: 4, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, f, formal, float
Method Body:
Block([])
METHOD: 5, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, b, formal, boolean
Method Body:
Block([])
METHOD: 6, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, s, formal, string
Method Body:
Block([])
---------------------------------------------------------------------------
Class name: test019
Superclass Name:
```

```
Fields:
Constructors:
Methods:
METHOD: 7, main, test019, public, static, void
Method Parameters:
Variable Table:
VARIABLE 1, a, local, boolean
Method Body:
Block([
Expr(Assign(Variable(1), Constant(False)))
])
--------------------------------------------------------------------------


     STUDENT ERROR:

     SHOULD BE: Check print out of AST


Press Enter to continue...
```

```
TEST CASE test020: STRING CONST Declaration and Assignment

     STUDENT:
--------------------------------------------------------------------------
Class name: In
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 1, scan_int, None, public, static, int
Method Parameters:
Variable Table:
Method Body:
Block([])
METHOD: 2, scan_float, None, public, static, float
Method Parameters:
Variable Table:
Method Body:
Block([])
--------------------------------------------------------------------------
Class name: Out
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 3, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, i, formal, int
Method Body:
Block([])
METHOD: 4, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, f, formal, float
Method Body:
Block([])
METHOD: 5, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, b, formal, boolean
Method Body:
Block([])
METHOD: 6, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, s, formal, string
Method Body:
Block([])
--------------------------------------------------------------------------
Class name: test020
Superclass Name:
```

```
Fields:
Constructors:
Methods:
METHOD: 7, main, test020, public, static, void
Method Parameters:
Variable Table:
VARIABLE 1, a, local, user(string)
Method Body:
Block([
Expr(Assign(Variable(1), Constant(String-constant("Hello"))))
])
---------------------------------------------------------------------------


     STUDENT ERROR:

     SHOULD BE: Check print out of AST


Press Enter to continue...
```

```
TEST CASE test021: AUTO-INCREMENT PRE

     STUDENT:
--------------------------------------------------------------------------
Class name: In
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 1, scan_int, None, public, static, int
Method Parameters:
Variable Table:
Method Body:
Block([])
METHOD: 2, scan_float, None, public, static, float
Method Parameters:
Variable Table:
Method Body:
Block([])
--------------------------------------------------------------------------
Class name: Out
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 3, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, i, formal, int
Method Body:
Block([])
METHOD: 4, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, f, formal, float
Method Body:
Block([])
METHOD: 5, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, b, formal, boolean
Method Body:
Block([])
METHOD: 6, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, s, formal, string
Method Body:
Block([])
--------------------------------------------------------------------------
Class name: test021
Superclass Name:
```

```
Fields:
Constructors:
Methods:
METHOD: 7, main, test021, public, static, void
Method Parameters:
Variable Table:
VARIABLE 1, a, local, int
Method Body:
Block([
Expr(Assign(Variable(1), Constant(Integer-constant(10))))
, Expr(Auto(Variable(1), inc, pre))
])
---------------------------------------------------------------------------


     STUDENT ERROR:

     SHOULD BE: Check print out of AST


Press Enter to continue...
```

```
TEST CASE test022: AUTO-DECREMENT PRE

     STUDENT:
--------------------------------------------------------------------------
Class name: In
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 1, scan_int, None, public, static, int
Method Parameters:
Variable Table:
Method Body:
Block([])
METHOD: 2, scan_float, None, public, static, float
Method Parameters:
Variable Table:
Method Body:
Block([])
--------------------------------------------------------------------------
Class name: Out
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 3, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, i, formal, int
Method Body:
Block([])
METHOD: 4, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, f, formal, float
Method Body:
Block([])
METHOD: 5, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, b, formal, boolean
Method Body:
Block([])
METHOD: 6, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, s, formal, string
Method Body:
Block([])
--------------------------------------------------------------------------
Class name: test022
Superclass Name:
```

```
Fields:
Constructors:
Methods:
METHOD: 7, main, test022, public, static, void
Method Parameters:
Variable Table:
VARIABLE 1, a, local, int
Method Body:
Block([
Expr(Assign(Variable(1), Constant(Integer-constant(10))))
, Expr(Auto(Variable(1), dec, pre))
])
--------------------------------------------------------------------------


     STUDENT ERROR:

     SHOULD BE: Check print out of AST


Press Enter to continue...
```

```
TEST CASE test023: AUTO-INCREMENT POST

     STUDENT:
--------------------------------------------------------------------------
Class name: In
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 1, scan_int, None, public, static, int
Method Parameters:
Variable Table:
Method Body:
Block([])
METHOD: 2, scan_float, None, public, static, float
Method Parameters:
Variable Table:
Method Body:
Block([])
--------------------------------------------------------------------------
Class name: Out
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 3, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, i, formal, int
Method Body:
Block([])
METHOD: 4, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, f, formal, float
Method Body:
Block([])
METHOD: 5, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, b, formal, boolean
Method Body:
Block([])
METHOD: 6, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, s, formal, string
Method Body:
Block([])
--------------------------------------------------------------------------
Class name: test023
Superclass Name:
```

```
Fields:
Constructors:
Methods:
METHOD: 7, main, test023, public, static, void
Method Parameters:
Variable Table:
VARIABLE 1, a, local, int
Method Body:
Block([
Expr(Assign(Variable(1), Constant(Integer-constant(10))))
, Expr(Auto(Variable(1), inc, post))
])
---------------------------------------------------------------------------


     STUDENT ERROR:

     SHOULD BE: Check print out of AST


Press Enter to continue...
```

```
TEST CASE test024: AUTO-DECREMENT POST

     STUDENT:
--------------------------------------------------------------------------
Class name: In
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 1, scan_int, None, public, static, int
Method Parameters:
Variable Table:
Method Body:
Block([])
METHOD: 2, scan_float, None, public, static, float
Method Parameters:
Variable Table:
Method Body:
Block([])
--------------------------------------------------------------------------
Class name: Out
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 3, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, i, formal, int
Method Body:
Block([])
METHOD: 4, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, f, formal, float
Method Body:
Block([])
METHOD: 5, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, b, formal, boolean
Method Body:
Block([])
METHOD: 6, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, s, formal, string
Method Body:
Block([])
--------------------------------------------------------------------------
Class name: test024
Superclass Name:
```

```
Fields:
Constructors:
Methods:
METHOD: 7, main, test024, public, static, void
Method Parameters:
Variable Table:
VARIABLE 1, a, local, int
Method Body:
Block([
Expr(Assign(Variable(1), Constant(Integer-constant(10))))
, Expr(Auto(Variable(1), dec, post))
])
---------------------------------------------------------------------------


     STUDENT ERROR:

     SHOULD BE: Check print out of AST


Press Enter to continue...
```

```
TEST CASE test025: IF Statement

     STUDENT:
--------------------------------------------------------------------------
Class name: In
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 1, scan_int, None, public, static, int
Method Parameters:
Variable Table:
Method Body:
Block([])
METHOD: 2, scan_float, None, public, static, float
Method Parameters:
Variable Table:
Method Body:
Block([])
--------------------------------------------------------------------------
Class name: Out
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 3, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, i, formal, int
Method Body:
Block([])
METHOD: 4, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, f, formal, float
Method Body:
Block([])
METHOD: 5, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, b, formal, boolean
Method Body:
Block([])
METHOD: 6, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, s, formal, string
Method Body:
Block([])
--------------------------------------------------------------------------
Class name: test025
Superclass Name:
```

```
Fields:
Constructors:
Methods:
METHOD: 7, main, test025, public, static, void
Method Parameters:
Variable Table:
VARIABLE 1, a, local, int
Method Body:
Block([
Expr(Assign(Variable(1), Constant(Integer-constant(0))))
, If(Binary(gt, Constant(Integer-constant(10)), Constant(Integer-
constant(5))), Block([
Expr(Assign(Variable(1), Constant(Integer-constant(10))))
])
, Skip
)
])
----------------------------------------------------------------------------


     STUDENT ERROR:

     SHOULD BE: Check print out of AST


Press Enter to continue...
```

```
TEST CASE test026: IF ELSE Statement

     STUDENT:
--------------------------------------------------------------------------
Class name: In
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 1, scan_int, None, public, static, int
Method Parameters:
Variable Table:
Method Body:
Block([])
METHOD: 2, scan_float, None, public, static, float
Method Parameters:
Variable Table:
Method Body:
Block([])
--------------------------------------------------------------------------
Class name: Out
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 3, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, i, formal, int
Method Body:
Block([])
METHOD: 4, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, f, formal, float
Method Body:
Block([])
METHOD: 5, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, b, formal, boolean
Method Body:
Block([])
METHOD: 6, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, s, formal, string
Method Body:
Block([])
--------------------------------------------------------------------------
Class name: test026
Superclass Name:
```

```
Fields:
Constructors:
Methods:
METHOD: 7, main, test026, public, static, void
Method Parameters:
Variable Table:
VARIABLE 1, a, local, int
Method Body:
Block([
Expr(Assign(Variable(1), Constant(Integer-constant(0))))
, If(Binary(lt, Constant(Integer-constant(10)), Constant(Integer-
constant(5))), Block([
Expr(Assign(Variable(1), Constant(Integer-constant(10))))
])
, Block([
Expr(Assign(Variable(1), Constant(Integer-constant(20))))
])
)
])
--------------------------------------------------------------------------


     STUDENT ERROR:

     SHOULD BE: Check print out of AST


Press Enter to continue...
```

```
TEST CASE test027: DANGLING ELSE Statement

     STUDENT:
'IFStatementNode' object has no attribute 'valid_block' column 1

     STUDENT ERROR:

     SHOULD BE: Check print out of AST


Press Enter to continue...
```

```
TEST CASE test028: FOR Statement

     STUDENT:
--------------------------------------------------------------------------
Class name: In
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 1, scan_int, None, public, static, int
Method Parameters:
Variable Table:
Method Body:
Block([])
METHOD: 2, scan_float, None, public, static, float
Method Parameters:
Variable Table:
Method Body:
Block([])
--------------------------------------------------------------------------
Class name: Out
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 3, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, i, formal, int
Method Body:
Block([])
METHOD: 4, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, f, formal, float
Method Body:
Block([])
METHOD: 5, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, b, formal, boolean
Method Body:
Block([])
METHOD: 6, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, s, formal, string
Method Body:
Block([])
--------------------------------------------------------------------------
Class name: test028
Superclass Name:
```

```
Fields:
Constructors:
Methods:
METHOD: 7, main, test028, public, static, void
Method Parameters:
Variable Table:
VARIABLE 1, a, local, int
VARIABLE 2, i, local, int
Method Body:
Block([
Expr(Assign(Variable(1), Constant(Integer-constant(0))))
, For(Assign(Variable(2), Constant(Integer-constant(0))), Binary(lt,
Variable(2), Constant(Integer-constant(10))), Auto(Variable(2), inc,
post), Block([
Expr(Assign(Variable(1), Binary(add, Variable(1), Constant(Integer-
constant(1)))))
])
)
])
--------------------------------------------------------------------------


     STUDENT ERROR:

     SHOULD BE: Check print out of AST


Press Enter to continue...
```

```
TEST CASE test029: WHILE Statement

     STUDENT:
--------------------------------------------------------------------------
Class name: In
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 1, scan_int, None, public, static, int
Method Parameters:
Variable Table:
Method Body:
Block([])
METHOD: 2, scan_float, None, public, static, float
Method Parameters:
Variable Table:
Method Body:
Block([])
--------------------------------------------------------------------------
Class name: Out
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 3, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, i, formal, int
Method Body:
Block([])
METHOD: 4, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, f, formal, float
Method Body:
Block([])
METHOD: 5, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, b, formal, boolean
Method Body:
Block([])
METHOD: 6, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, s, formal, string
Method Body:
Block([])
--------------------------------------------------------------------------
Class name: test029
Superclass Name:
```

```
Fields:
Constructors:
Methods:
METHOD: 7, main, test029, public, static, void
Method Parameters:
Variable Table:
VARIABLE 1, a, local, int
Method Body:
Block([
Expr(Assign(Variable(1), Constant(Integer-constant(0))))
, While(Binary(lt, Variable(1), Constant(Integer-constant(10))), Block([
Expr(Assign(Variable(1), Binary(add, Variable(1), Constant(Integer-constant(1)))))
])
)
])
---------------------------------------------------------------------------


     STUDENT ERROR:

     SHOULD BE: Check print out of AST


Press Enter to continue...
```

TEST CASE test030: STATIC METHOD Definition, RETURN Statment

```
     STUDENT:
---------------------------------------------------------------------------
Class name: In
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 1, scan_int, None, public, static, int
Method Parameters:
Variable Table:
Method Body:
Block([])
METHOD: 2, scan_float, None, public, static, float
Method Parameters:
Variable Table:
Method Body:
Block([])
---------------------------------------------------------------------------
Class name: Out
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 3, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, i, formal, int
Method Body:
Block([])
METHOD: 4, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, f, formal, float
Method Body:
Block([])
METHOD: 5, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, b, formal, boolean
Method Body:
Block([])
METHOD: 6, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, s, formal, string
Method Body:
Block([])
---------------------------------------------------------------------------
Class name: test030
Superclass Name:
```

```
Fields:
Constructors:
Methods:
METHOD: 7, add5, test030, public, static, int
Method Parameters: 1
Variable Table:
VARIABLE 1, n, formal, int
Method Body:
Block([
Return(Binary(add, Variable(1), Constant(Integer-constant(5))))
])
METHOD: 8, main, test030, public, static, void
Method Parameters:
Variable Table:
VARIABLE 1, a, local, int
Method Body:
Block([
Expr(Assign(Variable(1), Constant(Integer-constant(0))))
])
--------------------------------------------------------------------------


     STUDENT ERROR:

     SHOULD BE: Check print out of AST


Press Enter to continue...
```

```
TEST CASE test031: STATIC METHOD Invocation Statement

     STUDENT:
--------------------------------------------------------------------------
Class name: In
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 1, scan_int, None, public, static, int
Method Parameters:
Variable Table:
Method Body:
Block([])
METHOD: 2, scan_float, None, public, static, float
Method Parameters:
Variable Table:
Method Body:
Block([])
--------------------------------------------------------------------------
Class name: Out
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 3, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, i, formal, int
Method Body:
Block([])
METHOD: 4, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, f, formal, float
Method Body:
Block([])
METHOD: 5, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, b, formal, boolean
Method Body:
Block([])
METHOD: 6, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, s, formal, string
Method Body:
Block([])
--------------------------------------------------------------------------
Class name: test031
Superclass Name:
```

```
Fields:
Constructors:
Methods:
METHOD: 7, add5, test031, public, static, int
Method Parameters: 1
Variable Table:
VARIABLE 1, n, formal, int
Method Body:
Block([
Return(Binary(add, Variable(1), Constant(Integer-constant(5))))
])
METHOD: 8, main, test031, public, static, void
Method Parameters:
Variable Table:
VARIABLE 1, a, local, int
Method Body:
Block([
Expr(Assign(Variable(1), Method-call(Class-reference(test031), add5,
[Variable(1)])))
])
-------------------------------------------------------------------------


     STUDENT ERROR:

     SHOULD BE: Check print out of AST


Press Enter to continue...
```

```
TEST CASE test032: CLASS Definition, STATIC FIELD

     STUDENT:
'NoneType' object is not subscriptable column 1

     STUDENT ERROR:

     SHOULD BE: Check print out of AST


Press Enter to continue...
```

```
TEST CASE test033: CLASS Definition, CONSTRUCTOR Definition

     STUDENT:
'NoneType' object is not subscriptable column 1

     STUDENT ERROR:

     SHOULD BE: Check print out of AST


Press Enter to continue...
```

```
TEST CASE test034: CLASS Definition, CONSTRUCTOR Call

     STUDENT:
'NoneType' object is not subscriptable column 1

     STUDENT ERROR:

     SHOULD BE: Check print out of AST


Press Enter to continue...
```

```
TEST CASE test035: CLASS Definition, NON-STATIC METHOD Definition

     STUDENT:
'NoneType' object is not subscriptable column 1

     STUDENT ERROR:

     SHOULD BE: Check print out of AST


Press Enter to continue...
```

```
TEST CASE test036: CLASS Definition, NON-STATIC METHOD Invocation

     STUDENT:
'NoneType' object is not subscriptable column 1

     STUDENT ERROR:

     SHOULD BE: Check print out of AST


Press Enter to continue...
```

```
TEST CASE test037: Example in Assignment, tests EXTENDS

     STUDENT:
'NoneType' object is not subscriptable column 1

     STUDENT ERROR:

     SHOULD BE: Check print out of AST


Press Enter to continue...
```

```
TEST CASE test038: hello_world.decaf

     STUDENT:
--------------------------------------------------------------------------
Class name: In
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 1, scan_int, None, public, static, int
Method Parameters:
Variable Table:
Method Body:
Block([])
METHOD: 2, scan_float, None, public, static, float
Method Parameters:
Variable Table:
Method Body:
Block([])
--------------------------------------------------------------------------
Class name: Out
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 3, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, i, formal, int
Method Body:
Block([])
METHOD: 4, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, f, formal, float
Method Body:
Block([])
METHOD: 5, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, b, formal, boolean
Method Body:
Block([])
METHOD: 6, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, s, formal, string
Method Body:
Block([])
--------------------------------------------------------------------------
Class name: hello_world
Superclass Name:
```

```
Fields:
Constructors:
Methods:
METHOD: 7, main, hello_world, public, static, void
Method Parameters:
Variable Table:
Method Body:
Block([
Expr(Method-call(Class-reference(Out), print, [Constant(String-
constant("Hello World!\n"))]))
])
--------------------------------------------------------------------------


     STUDENT ERROR:

     SHOULD BE: Check print out of AST


Press Enter to continue...
```

```
TEST CASE test039: nrfib.decaf

     STUDENT:
--------------------------------------------------------------------------
Class name: In
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 1, scan_int, None, public, static, int
Method Parameters:
Variable Table:
Method Body:
Block([])
METHOD: 2, scan_float, None, public, static, float
Method Parameters:
Variable Table:
Method Body:
Block([])
--------------------------------------------------------------------------
Class name: Out
Superclass Name:
Fields:
Constructors:
Methods:
METHOD: 3, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, i, formal, int
Method Body:
Block([])
METHOD: 4, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, f, formal, float
Method Body:
Block([])
METHOD: 5, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, b, formal, boolean
Method Body:
Block([])
METHOD: 6, print, None, public, static, None
Method Parameters: 1
Variable Table:
VARIABLE 1, s, formal, string
Method Body:
Block([])
--------------------------------------------------------------------------
Class name: nrfib
Superclass Name:
```

```
Fields:
Constructors:
Methods:
METHOD: 7, main, nrfib, public, static, void
Method Parameters:
Variable Table:
VARIABLE 1, n, local, int
VARIABLE 2, i, local, int
VARIABLE 3, fn, local, int
VARIABLE 4, fn_prev, local, int
Method Body:
Block([
Expr(Assign(Variable(1), Method-call(Class-reference(In), scan_int, [])))
, Expr(Assign(Variable(3), Constant(Integer-constant(1))))
, Expr(Assign(Variable(4), Constant(Integer-constant(0))))
, For(Assign(Variable(2), Constant(Integer-constant(1))), Binary(lt,
Variable(2), Variable(1)), Assign(Variable(2), Binary(add, Variable(2),
Constant(Integer-constant(1)))), Block([
Expr(Assign(Variable(3), Binary(add, Variable(4), Variable(3))))
, Expr(Assign(Variable(4), Binary(sub, Variable(3), Variable(4))))
])
)
, Expr(Method-call(Class-reference(Out), print, [Constant(String-
constant("Fib = "))]))
, Expr(Method-call(Class-reference(Out), print, [Variable(3)]))
, Expr(Method-call(Class-reference(Out), print, [Constant(String-
constant("\n"))]))
])
---------------------------------------------------------------------------


     STUDENT ERROR:

     SHOULD BE: Check print out of AST


Press Enter to continue...
```

```
TEST CASE test040: rfib.decaf

     STUDENT:
Unexpected token "-1" of type "INT_CONST" found on line 6 column 25

     STUDENT ERROR:

     SHOULD BE: Check print out of AST


Press Enter to continue...
```

```
TEST CASE test041: IntList.decaf

     STUDENT:
'NoneType' object is not subscriptable column 1

     STUDENT ERROR:

     SHOULD BE: Check print out of AST


Press Enter to continue...
```

```
TEST CASE test042: Bank Accounts

     STUDENT:
Unexpected token "}" of type "RBRACE" found on line 29 column 9

     STUDENT ERROR:

     SHOULD BE: Check print out of AST


Press Enter to continue...
```

```
TEST CASE error01: Duplicate Class Name

     STUDENT:
Duplicate classname detected: repeat error01 found on line 16 column 1

     STUDENT ERROR:

     SHOULD BE: Check print out of AST


Press Enter to continue...
```

```
TEST CASE error02: Duplicate Field Name

     STUDENT:
Duplicate field name "x" found on lines 3 and 4 column 1

     STUDENT ERROR:

     SHOULD BE: Check print out of AST


Press Enter to continue...
```

```
TEST CASE error03: Duplicate Variable

        STUDENT:
'NoneType' object is not subscriptable column 1

        STUDENT ERROR:

        SHOULD BE: Check print out of AST


Press Enter to continue...
```