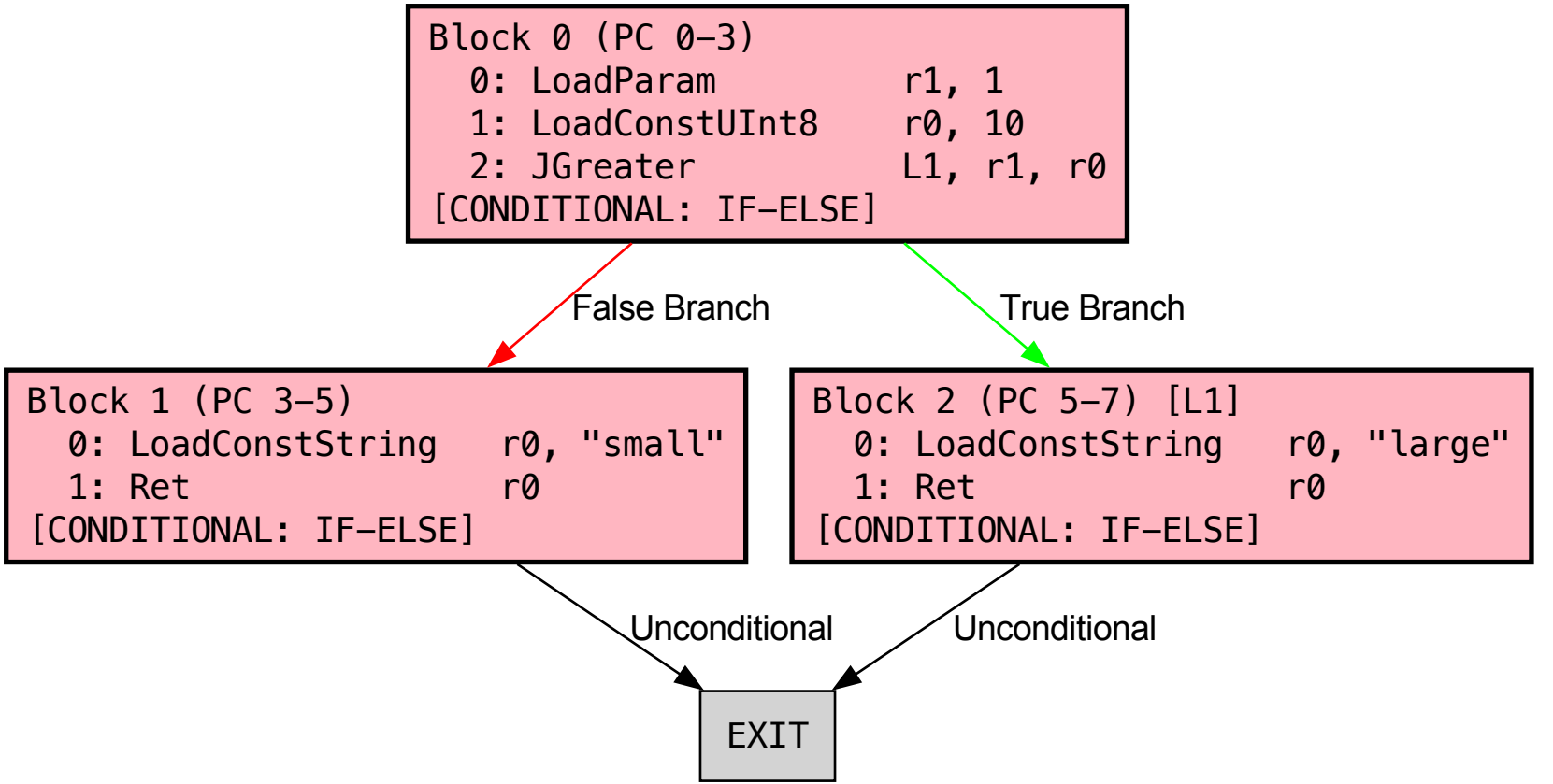
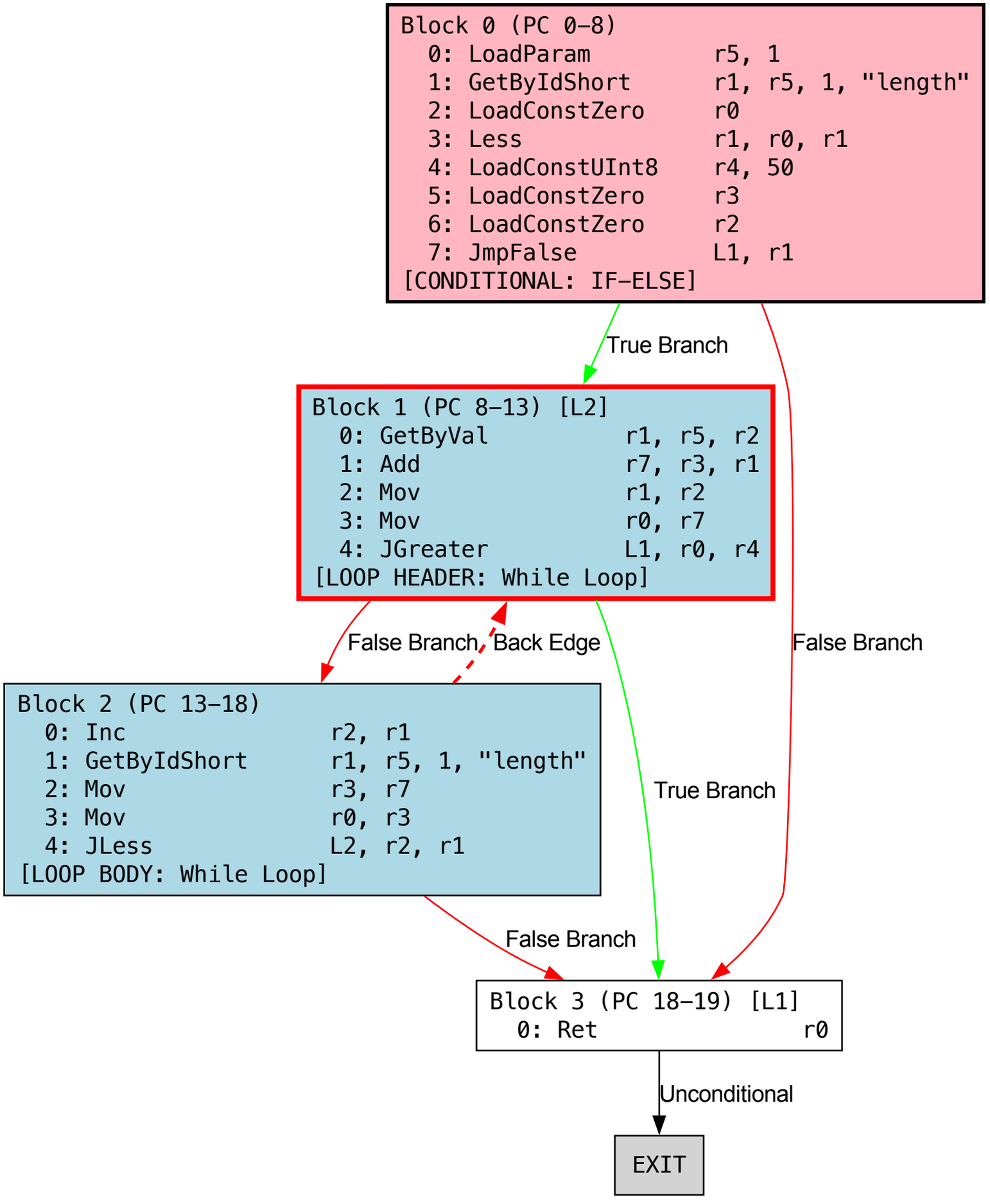
```
Block 0 (PC 0-33)
  0: DeclareGlobalVar  "ifElseTest"
  1: DeclareGlobalVar  "forLoopTest"
  2: DeclareGlobalVar  "switchTest"
  3: DeclareGlobalVar  "whileTest"
  4: DeclareGlobalVar  "tryCatchTest"
  5: DeclareGlobalVar  "asyncAwaitTest"
  6: DeclareGlobalVar  "promiseChainTest"
  7: DeclareGlobalVar  "generatorTest"
  8: DeclareGlobalVar  "callbackTest"
  9: CreateEnvironment r1
  10: CreateClosure     r0, r1, Function<ifElseTest>1
  11: GetGlobalObject   r2
  12: PutById           r2, r0, 1, "ifElseTest"
  13: CreateClosure     r0, r1, Function<forLoopTest>2
  14: PutById           r2, r0, 2, "forLoopTest"
  15: CreateClosure     r0, r1, Function<switchTest>3
  16: PutById           r2, r0, 3, "switchTest"
  17: CreateClosure     r0, r1, Function<whileTest>4
  18: PutById           r2, r0, 4, "whileTest"
  19: CreateClosure     r0, r1, Function<tryCatchTest>5
  20: PutById           r2, r0, 5, "tryCatchTest"
  21: CreateAsyncClosure r0, r1, Function<asyncAwaitTest>6
  22: PutById           r2, r0, 6, "asyncAwaitTest"
  23: CreateClosure     r0, r1, Function<promiseChainTest>9
  24: PutById           r2, r0, 7, "promiseChainTest"
  25: CreateClosure     r0, r1, Function<generatorTest>12
  26: PutById           r2, r0, 8, "generatorTest"
  27: CreateClosure     r0, r1, Function<callbackTest>15
  28: PutById           r2, r0, 9, "callbackTest"
  29: CreateClosure     r0, r1, Function<sleep>17
  30: StoreToEnvironment r1, 0, r0
  31: LoadConstUndefined r0
  32: Ret               r0
```

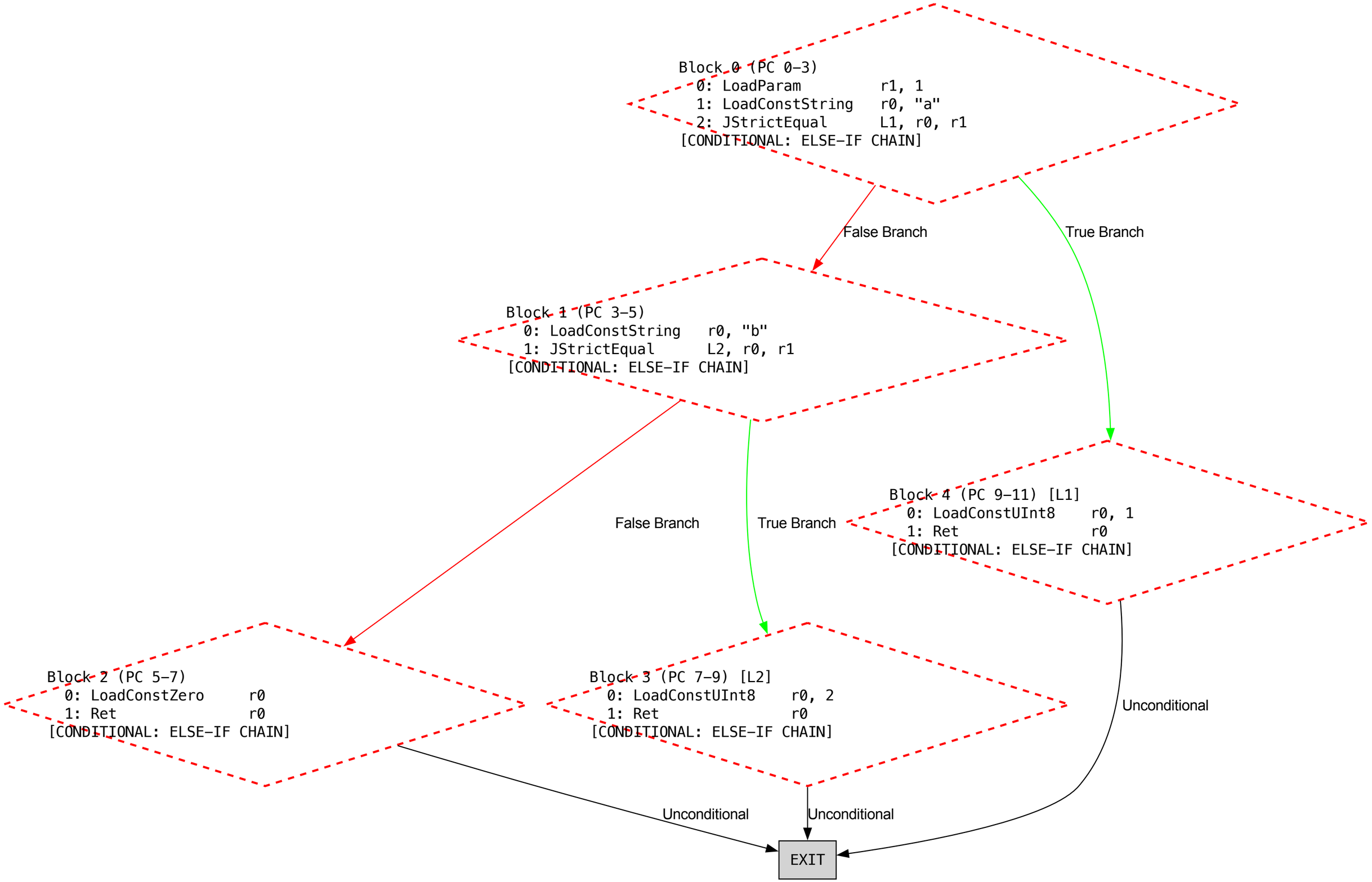
Function 0


Block 0 (PC 0-3)
0: LoadParam r1, 1
1: LoadConstUInt8 r0, 10
2: JGreater L1, r1, r0
[CONDITIONAL: IF-ELSE]
False Branch
True Branch
Block 1 (PC 3-5)
0: LoadConstString r0, "small"
1: Ret r0
[CONDITIONAL: IF-ELSE]
Block 2 (PC 5-7) [L1]
0: LoadConstString r0, "large"
1: Ret r0
[CONDITIONAL: IF-ELSE]
Unconditional
Unconditional
EXIT

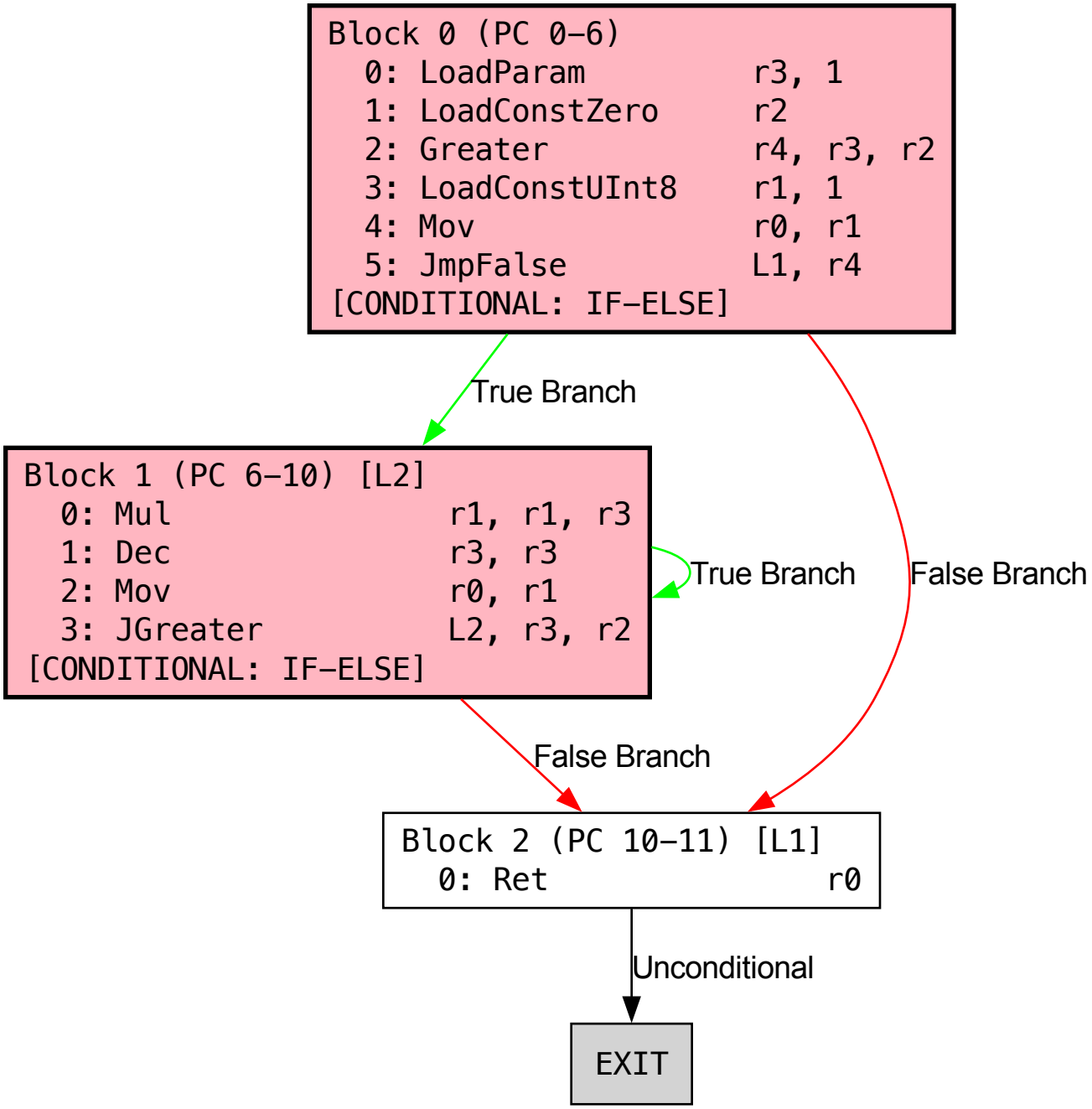
Function 1


Block 0 (PC 0-8)
0: LoadParam r5, 1
1: GetByIdShort r1, r5, 1, "length"
2: LoadConstZero r0
3: Less r1, r0, r1
4: LoadConstUInt8 r4, 50
5: LoadConstZero r3
6: LoadConstZero r2
7: JmpFalse L1, r1
[CONDITIONAL: IF-ELSE]
True Branch
Block 1 (PC 8-13) [L2]
0: GetByVal r1, r5, r2
1: Add r7, r3, r1
2: Mov r1, r2
3: Mov r0, r7
4: JGreater L1, r0, r4
[LOOP HEADER: While Loop]
False Branch
Back Edge
False Branch
Block 2 (PC 13-18)
0: Inc r2, r1
1: GetByIdShort r1, r5, 1, "length"
2: Mov r3, r7
3: Mov r0, r3
4: JLess L2, r2, r1
[LOOP BODY: While Loop]
True Branch
False Branch
Block 3 (PC 18-19) [L1]
0: Ret r0
Unconditional
EXIT

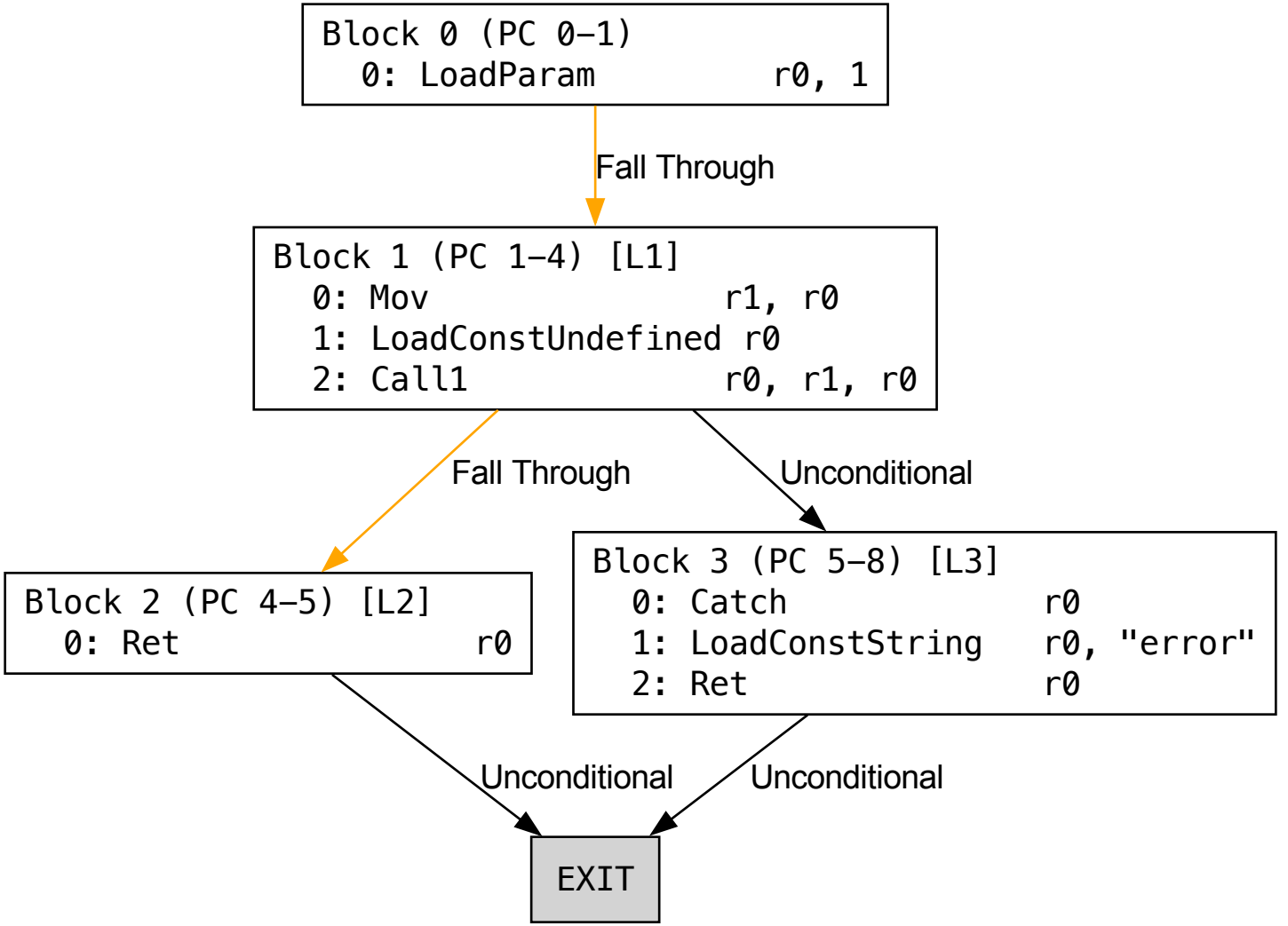
Function 2


Block 0 (PC 0-3)
0: LoadParam r1, 1
1: LoadConstString r0, "a"
2: JStrictEqual L1, r0, r1
[CONDITIONAL: ELSE-IF CHAIN]
False Branch
True Branch
Block 1 (PC 3-5)
0: LoadConstString r0, "b"
1: JStrictEqual L2, r0, r1
[CONDITIONAL: ELSE-IF CHAIN]
False Branch
True Branch
Block 4 (PC 9-11) [L1]
0: LoadConstUInt8 r0, 1
1: Ret r0
[CONDITIONAL: ELSE-IF CHAIN]
Block 2 (PC 5-7)
0: LoadConstZero r0
1: Ret r0
[CONDITIONAL: ELSE-IF CHAIN]
Block 3 (PC 7-9) [L2]
0: LoadConstUInt8 r0, 2
1: Ret r0
[CONDITIONAL: ELSE-IF CHAIN]
Unconditional
Unconditional
Unconditional
EXIT

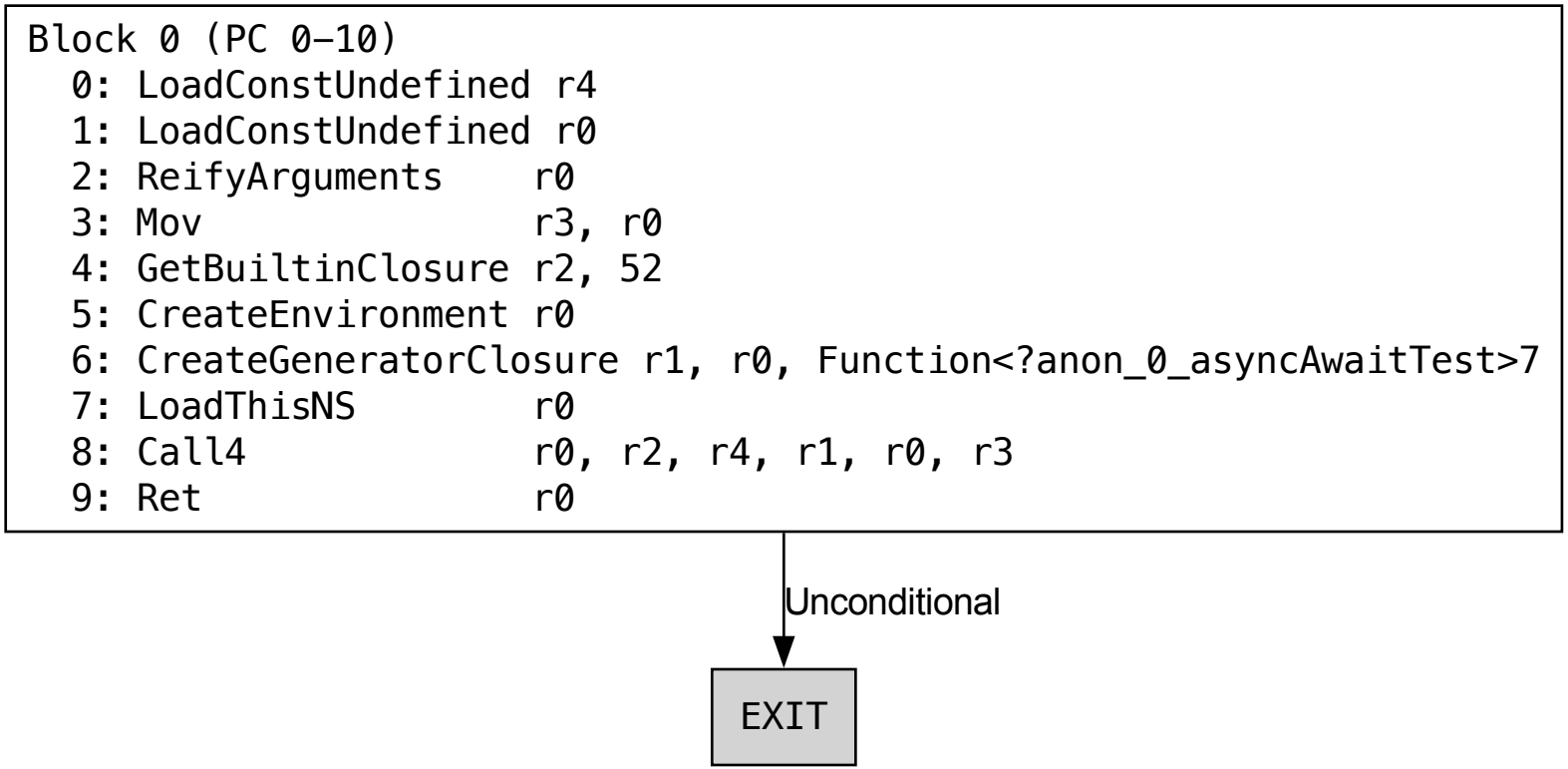
Function 3


Block 0 (PC 0-6)
0: LoadParam r3, 1
1: LoadConstZero r2
2: Greater r4, r3, r2
3: LoadConstUInt8 r1, 1
4: Mov r0, r1
5: JmpFalse L1, r4
[CONDITIONAL: IF-ELSE]
True Branch
Block 1 (PC 6-10) [L2]
0: Mul r1, r1, r3
1: Dec r3, r3
2: Mov r0, r1
3: JGreater L2, r3, r2
[CONDITIONAL: IF-ELSE]
True Branch
False Branch
False Branch
Block 2 (PC 10-11) [L1]
0: Ret r0
Unconditional
EXIT

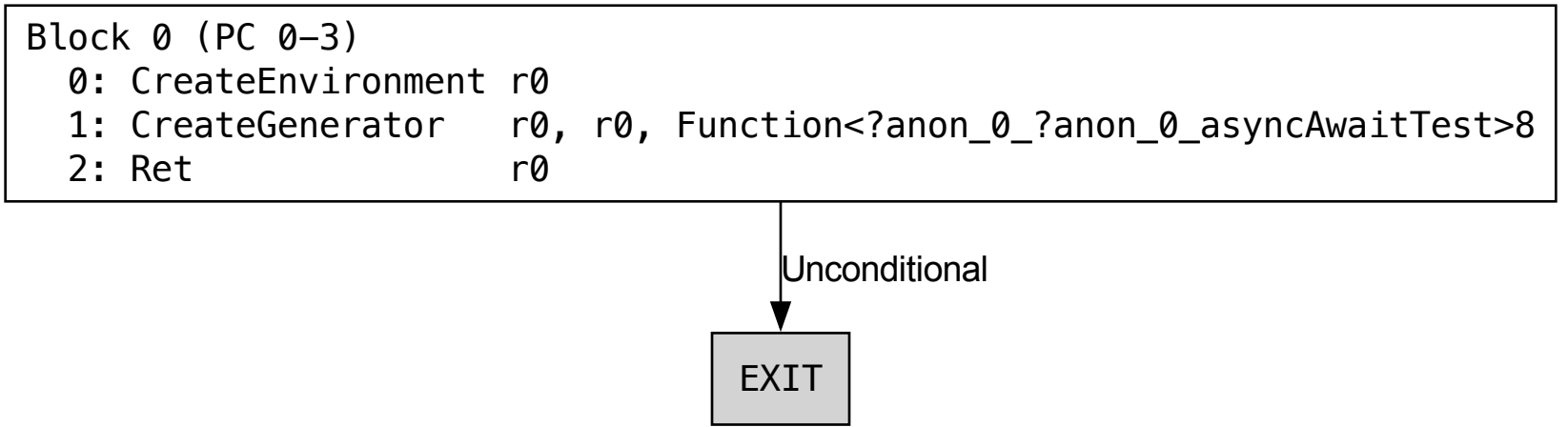
Function 4


Block 0 (PC 0–1)
0: LoadParam r0, 1
Fall Through
Block 1 (PC 1–4) [L1]
0: Mov r1, r0
1: LoadConstUndefined r0
2: Call1 r0, r1, r0
Fall Through
Unconditional
Block 2 (PC 4–5) [L2]
0: Ret r0
Block 3 (PC 5–8) [L3]
0: Catch r0
1: LoadConstString r0, "error"
2: Ret r0
Unconditional
Unconditional
EXIT

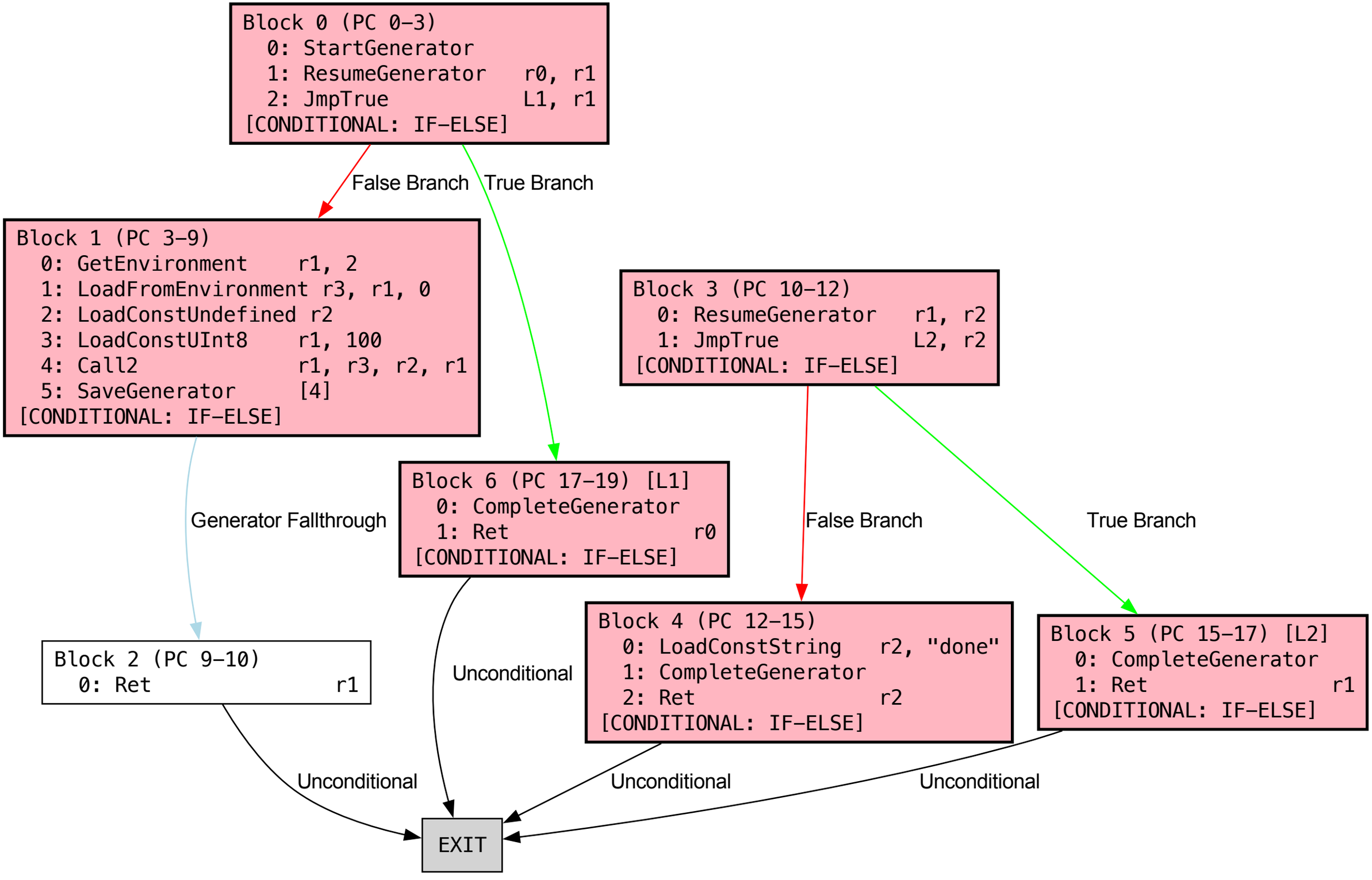
Function 5


Block 0 (PC 0–10)
0: LoadConstUndefined r4
1: LoadConstUndefined r0
2: ReifyArguments r0
3: Mov r3, r0
4: GetBuiltinClosure r2, 52
5: CreateEnvironment r0
6: CreateGeneratorClosure r1, r0, Function<?anon_0_asyncAwaitTest>7
7: LoadThisNS r0
8: Call4 r0, r2, r4, r1, r0, r3
9: Ret r0
Unconditional
EXIT

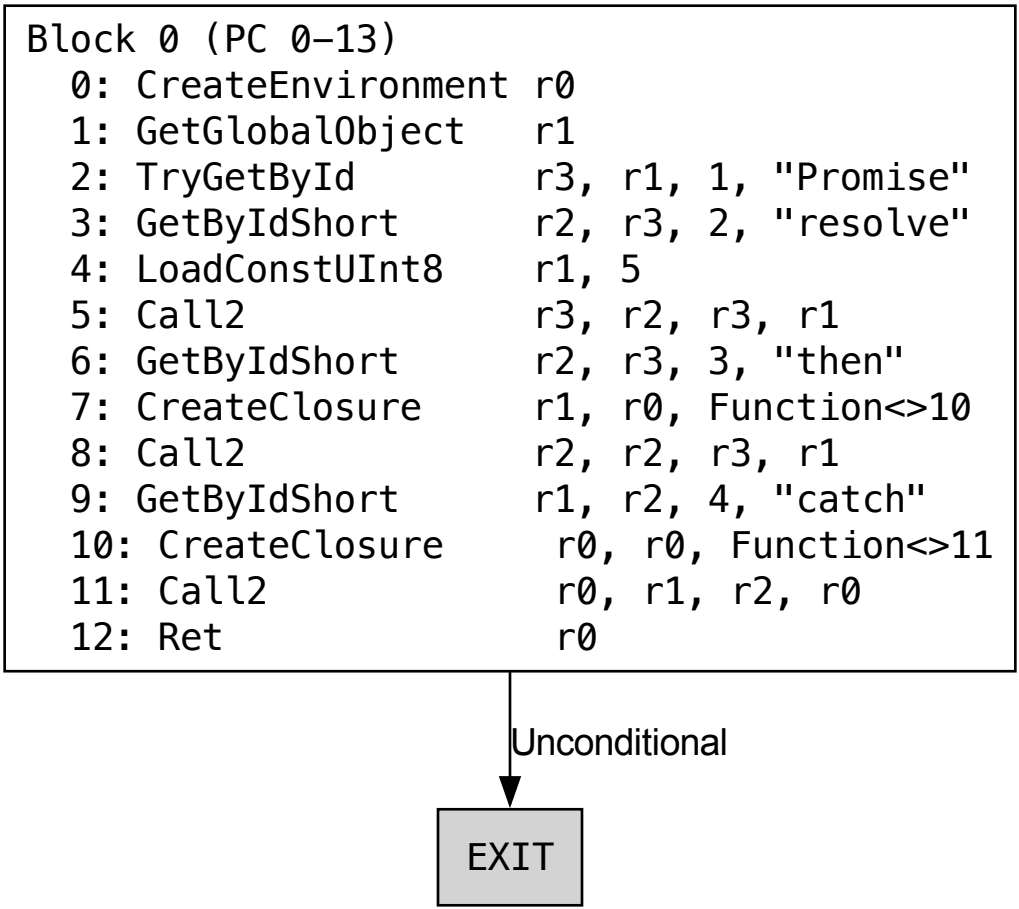
Function 6


Block 0 (PC 0–3)
0: CreateEnvironment r0
1: CreateGenerator r0, r0, Function<?anon_0_?anon_0_asyncAwaitTest>8
2: Ret r0
Unconditional
EXIT

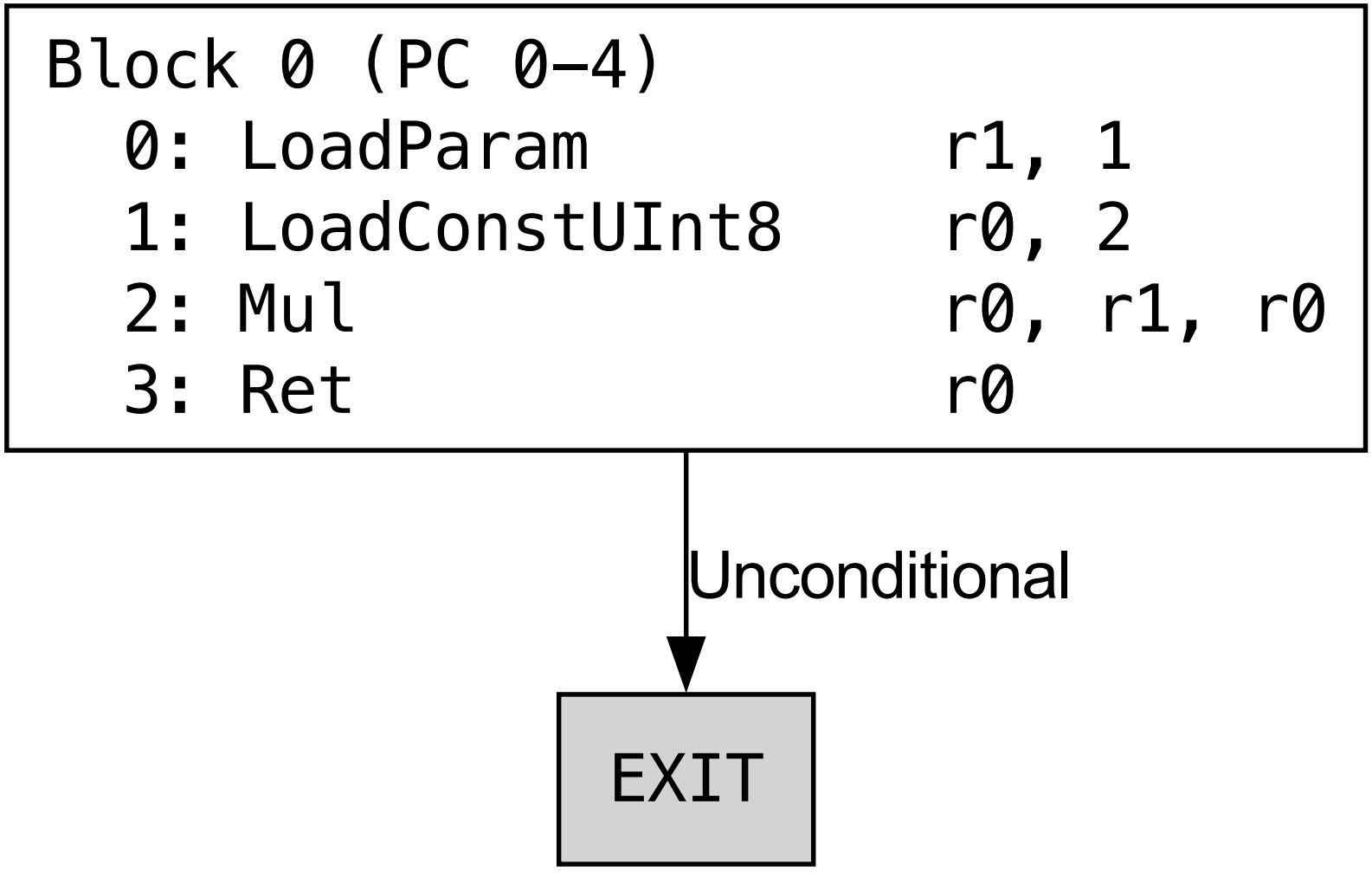
Function 7


Block 0 (PC 0–3)
0: StartGenerator
1: ResumeGenerator r0, r1
2: JmpTrue L1, r1
[CONDITIONAL: IF–ELSE]
False Branch
True Branch
Block 1 (PC 3–9)
0: GetEnvironment r1, 2
1: LoadFromEnvironment r3, r1, 0
2: LoadConstUndefined r2
3: LoadConstUInt8 r1, 100
4: Call2 r1, r3, r2, r1
5: SaveGenerator [4]
[CONDITIONAL: IF–ELSE]
Block 3 (PC 10–12)
0: ResumeGenerator r1, r2
1: JmpTrue L2, r2
[CONDITIONAL: IF–ELSE]
Generator Fallthrough
Block 6 (PC 17–19) [L1]
0: CompleteGenerator
1: Ret r0
[CONDITIONAL: IF–ELSE]
False Branch
True Branch
Block 2 (PC 9–10)
0: Ret r1
Unconditional
Block 4 (PC 12–15)
0: LoadConstString r2, "done"
1: CompleteGenerator
2: Ret r2
[CONDITIONAL: IF–ELSE]
Block 5 (PC 15–17) [L2]
0: CompleteGenerator
1: Ret r1
[CONDITIONAL: IF–ELSE]
Unconditional
Unconditional
Unconditional
EXIT

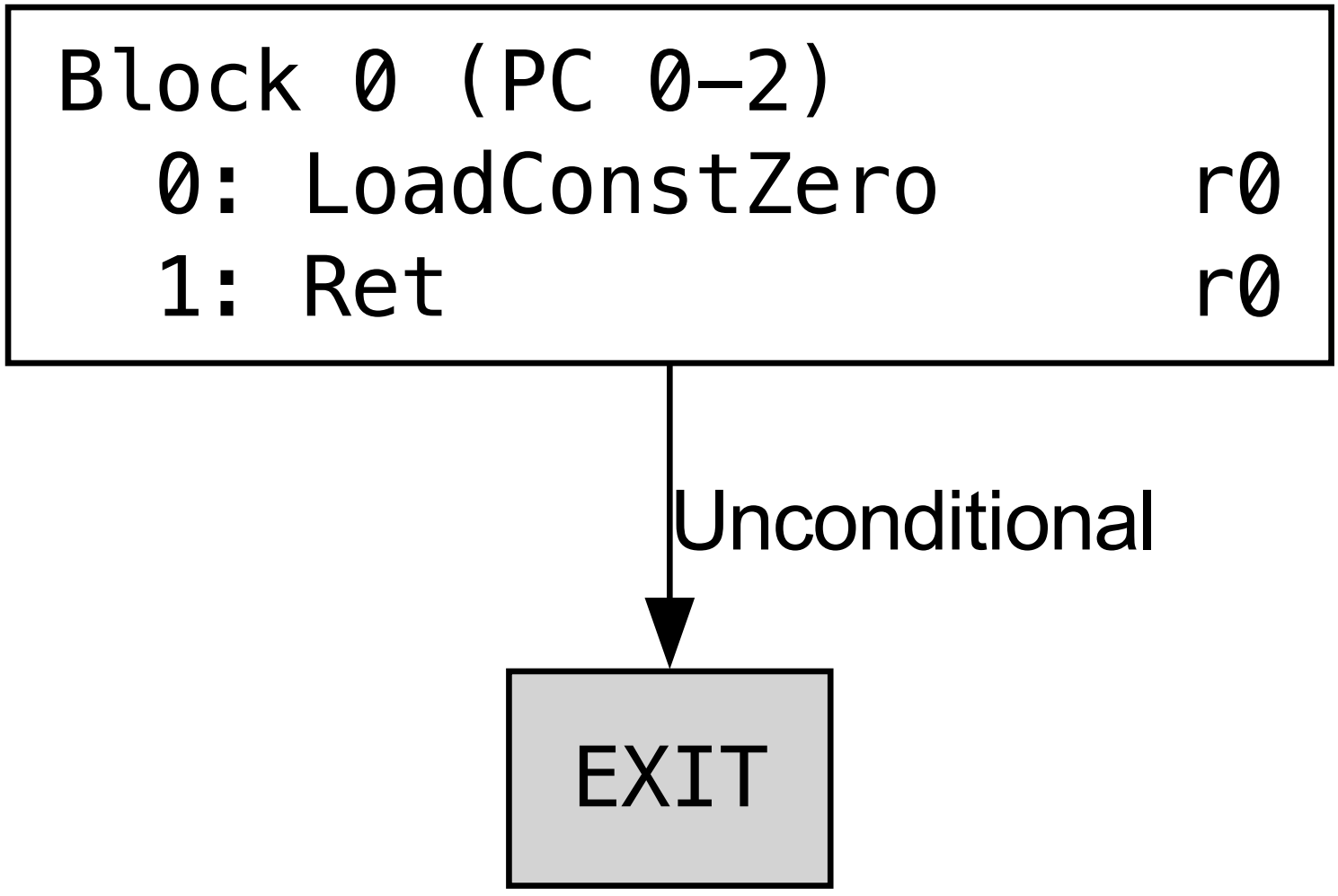
Function 8


Block 0 (PC 0-13)
0: CreateEnvironment r0
1: GetGlobalObject r1
2: TryGetById r3, r1, 1, "Promise"
3: GetByIdShort r2, r3, 2, "resolve"
4: LoadConstUInt8 r1, 5
5: Call2 r3, r2, r3, r1
6: GetByIdShort r2, r3, 3, "then"
7: CreateClosure r1, r0, Function<>10
8: Call2 r2, r2, r3, r1
9: GetByIdShort r1, r2, 4, "catch"
10: CreateClosure r0, r0, Function<>11
11: Call2 r0, r1, r2, r0
12: Ret r0
Unconditional
EXIT

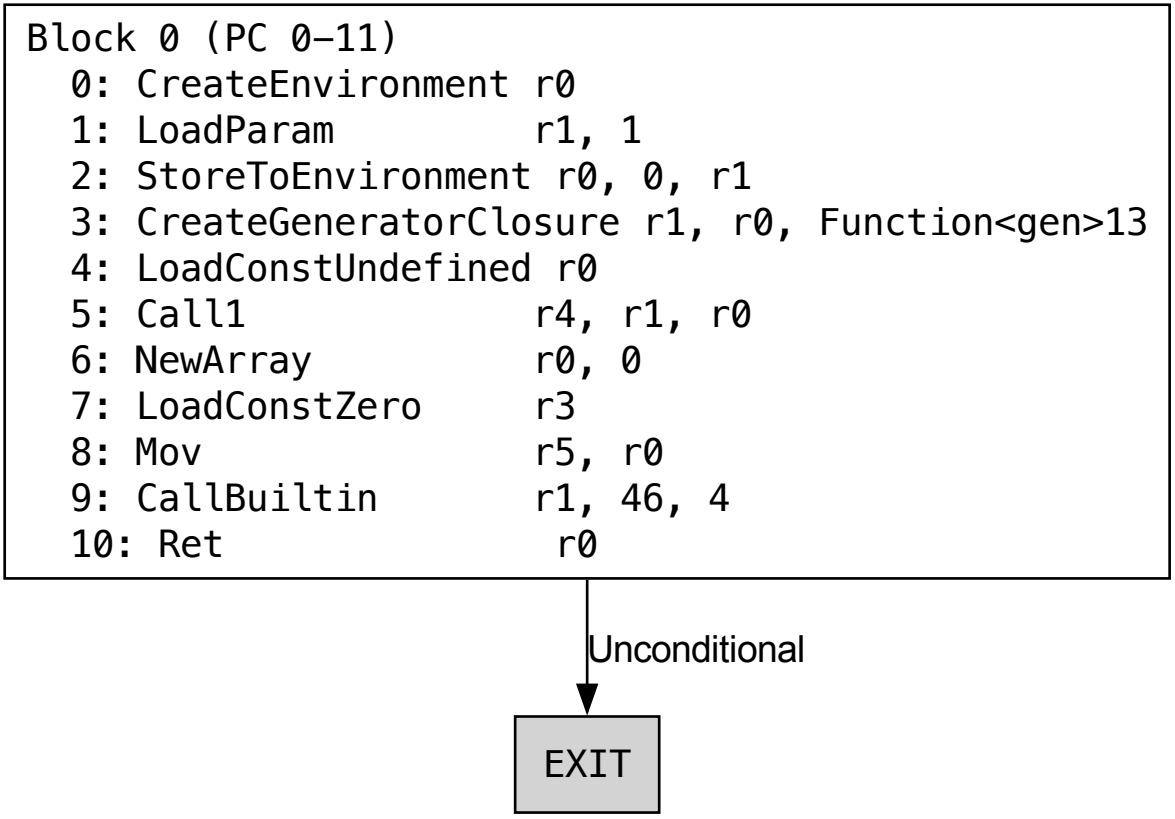
Function 9


Block 0 (PC 0–4)
0: LoadParam r1, 1
1: LoadConstUInt8 r0, 2
2: Mul r0, r1, r0
3: Ret r0
Unconditional
EXIT

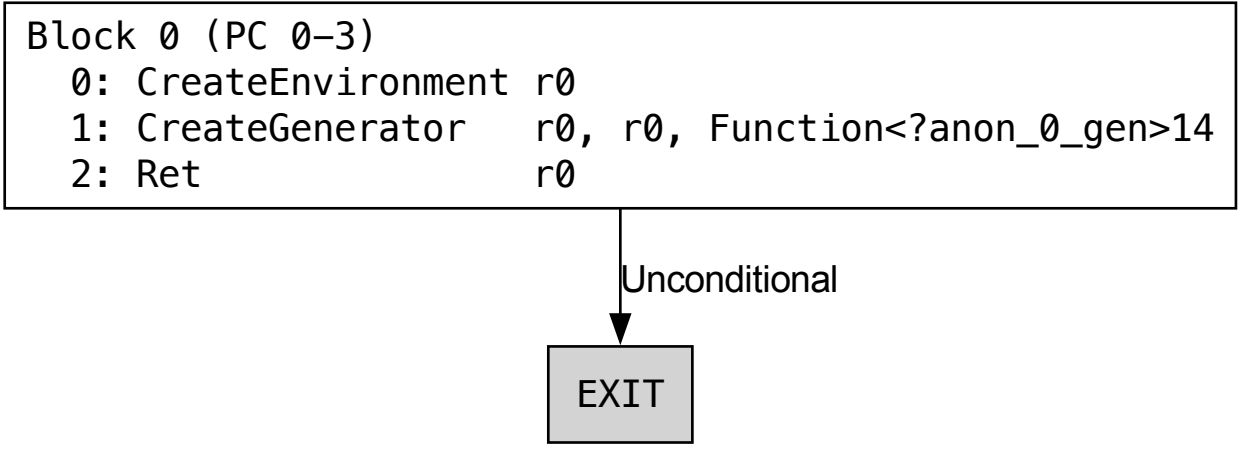
Function 10



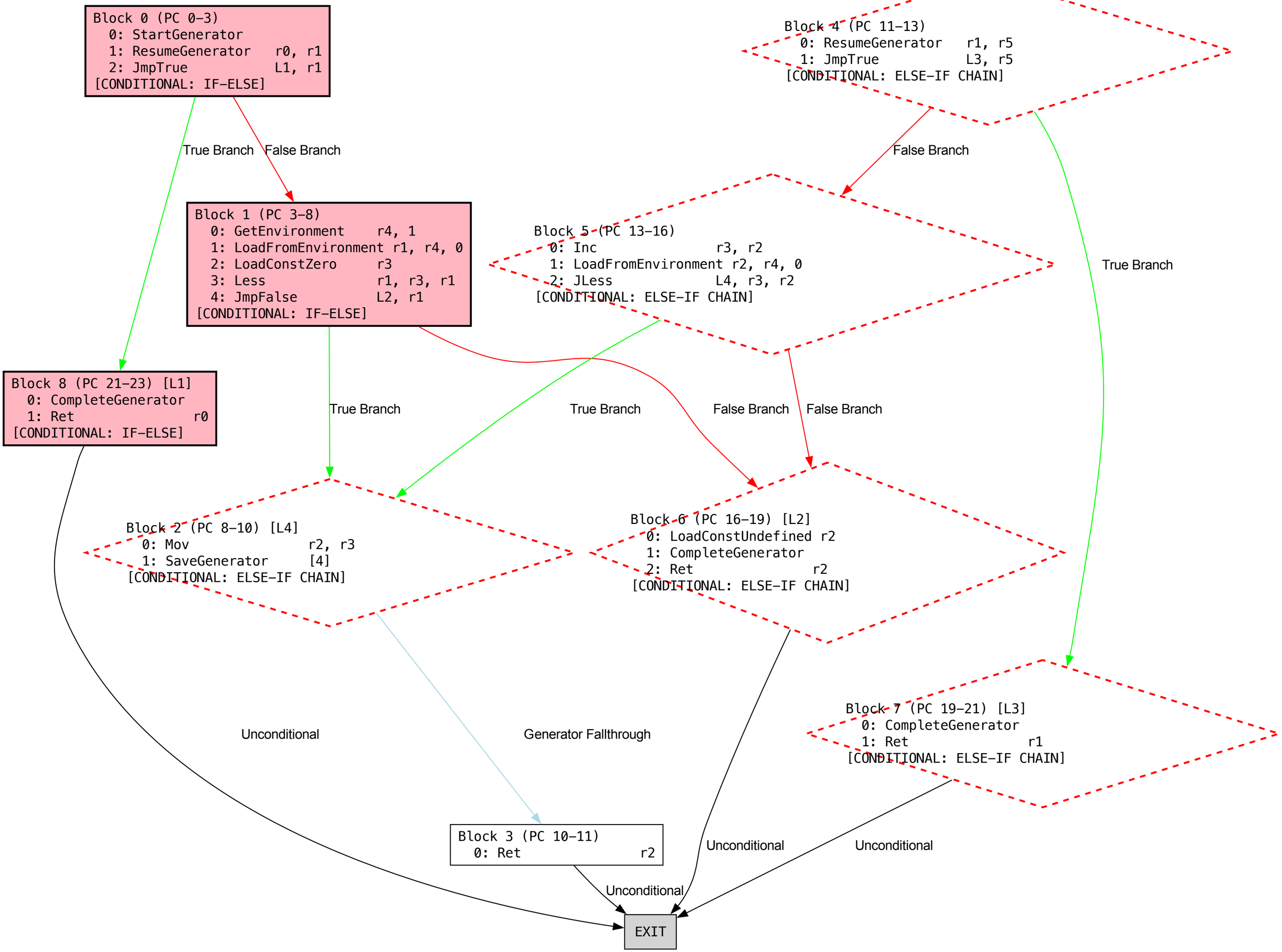
Function 11


Block 0 (PC 0–11)
0: CreateEnvironment r0
1: LoadParam r1, 1
2: StoreToEnvironment r0, 0, r1
3: CreateGeneratorClosure r1, r0, Function<gen>13
4: LoadConstUndefined r0
5: Call1 r4, r1, r0
6: NewArray r0, 0
7: LoadConstZero r3
8: Mov r5, r0
9: CallBuiltin r1, 46, 4
10: Ret r0
Unconditional
EXIT

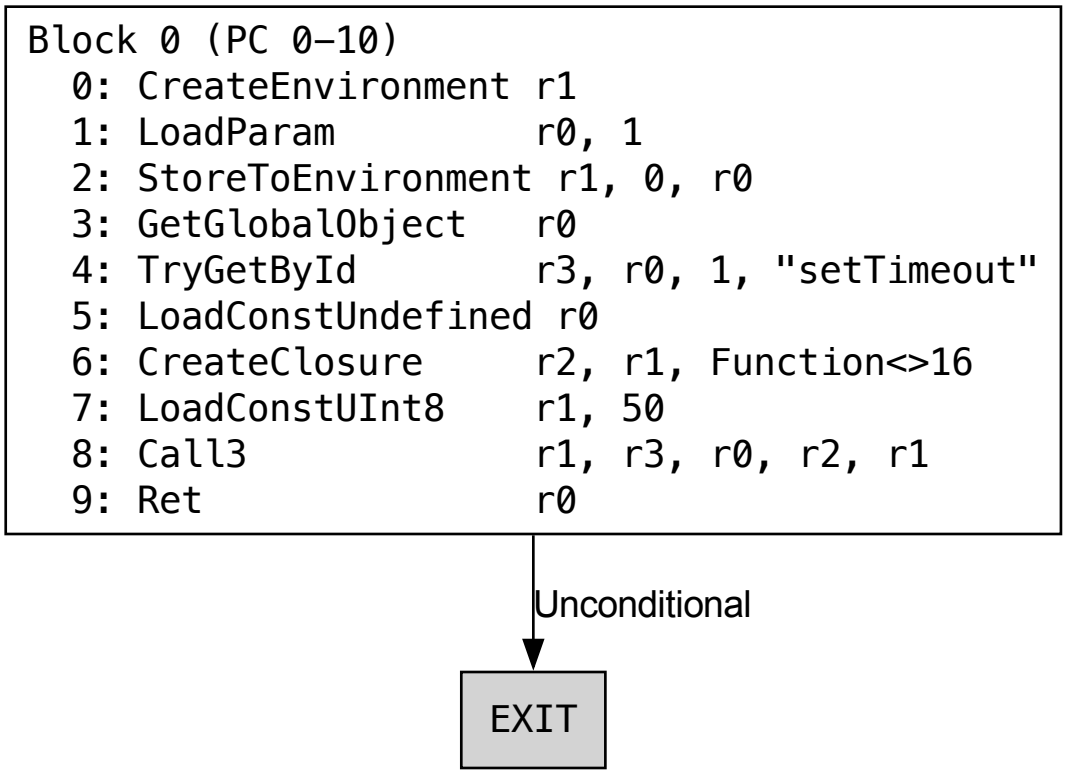
Function 12


Block 0 (PC 0–3)
0: CreateEnvironment r0
1: CreateGenerator r0, r0, Function<?anon_0_gen>14
2: Ret r0
Unconditional
EXIT

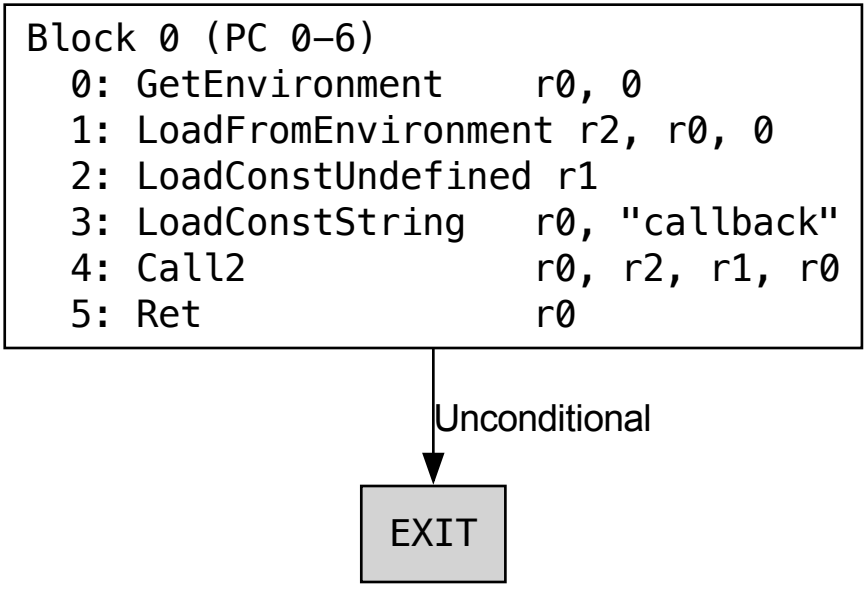
Function 13


Block 0 (PC 0–3)
0: StartGenerator
1: ResumeGenerator r0, r1
2: JmpTrue L1, r1
[CONDITIONAL: IF–ELSE]
Block 4 (PC 11–13)
0: ResumeGenerator r1, r5
1: JmpTrue L3, r5
[CONDITIONAL: ELSE–IF CHAIN]
Block 1 (PC 3–8)
0: GetEnvironment r4, 1
1: LoadFromEnvironment r1, r4, 0
2: LoadConstZero r3
3: Less r1, r3, r1
4: JmpFalse L2, r1
[CONDITIONAL: IF–ELSE]
Block 5 (PC 13–16)
0: Inc r3, r2
1: LoadFromEnvironment r2, r4, 0
2: JLess L4, r3, r2
[CONDITIONAL: ELSE–IF CHAIN]
Block 8 (PC 21–23) [L1]
0: CompleteGenerator
1: Ret r0
[CONDITIONAL: IF–ELSE]
Block 2 (PC 8–10) [L4]
0: Mov r2, r3
1: SaveGenerator [4]
[CONDITIONAL: ELSE–IF CHAIN]
Block 6 (PC 16–19) [L2]
0: LoadConstUndefined r2
1: CompleteGenerator
2: Ret r2
[CONDITIONAL: ELSE–IF CHAIN]
Block 7 (PC 19–21) [L3]
0: CompleteGenerator
1: Ret r1
[CONDITIONAL: ELSE–IF CHAIN]
Block 3 (PC 10–11)
0: Ret r2
EXIT
True Branch
False Branch
True Branch
False Branch
True Branch
False Branch
True Branch
Unconditional
Generator Fallthrough
Unconditional
Unconditional
Unconditional

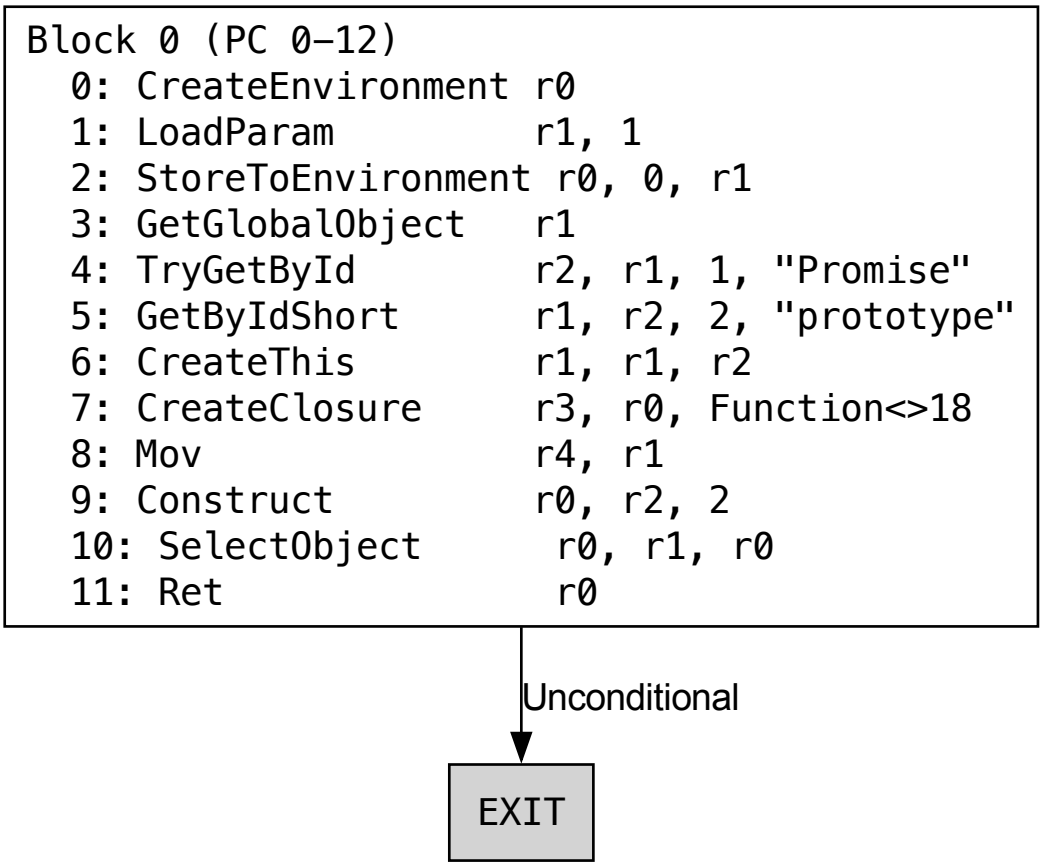
Function 14


Block 0 (PC 0–10)
0: CreateEnvironment r1
1: LoadParam r0, 1
2: StoreToEnvironment r1, 0, r0
3: GetGlobalObject r0
4: TryGetById r3, r0, 1, "setTimeout"
5: LoadConstUndefined r0
6: CreateClosure r2, r1, Function<>16
7: LoadConstUInt8 r1, 50
8: Call3 r1, r3, r0, r2, r1
9: Ret r0
Unconditional
EXIT

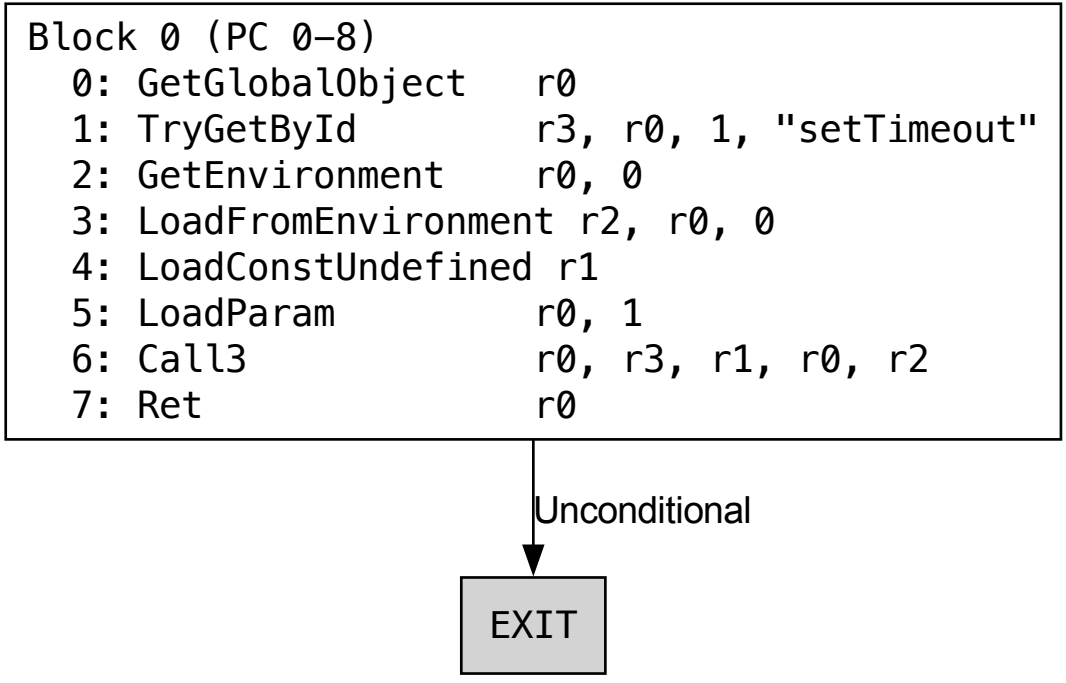
Function 15


Block 0 (PC 0-6)
0: GetEnvironment r0, 0
1: LoadFromEnvironment r2, r0, 0
2: LoadConstUndefined r1
3: LoadConstString r0, "callback"
4: Call2 r0, r2, r1, r0
5: Ret r0
Unconditional
EXIT

Function 16


Block 0 (PC 0–12)
0: CreateEnvironment r0
1: LoadParam r1, 1
2: StoreToEnvironment r0, 0, r1
3: GetGlobalObject r1
4: TryGetById r2, r1, 1, "Promise"
5: GetByIdShort r1, r2, 2, "prototype"
6: CreateThis r1, r1, r2
7: CreateClosure r3, r0, Function<>18
8: Mov r4, r1
9: Construct r0, r2, 2
10: SelectObject r0, r1, r0
11: Ret r0
Unconditional
EXIT

Function 17


Block 0 (PC 0-8)
0: GetGlobalObject r0
1: TryGetById r3, r0, 1, "setTimeout"
2: GetEnvironment r0, 0
3: LoadFromEnvironment r2, r0, 0
4: LoadConstUndefined r1
5: LoadParam r0, 1
6: Call3 r0, r3, r1, r0, r2
7: Ret r0
Unconditional
EXIT

Function 18