```
Block 0 (PC 0-46)
  0: DeclareGlobalVar    "testx"
  1: DeclareGlobalVar    "gen"
  2: DeclareGlobalVar    "ze"
  3: CreateEnvironment r1
  4: CreateAsyncClosure r2, r1, Function<testx>1
  5: GetGlobalObject     r0
  6: PutById             r0, r2, 1, "testx"
  7: CreateGeneratorClosure r2, r1, Function<gen>4
  8: PutById             r0, r2, 2, "gen"
  9: CreateClosure       r1, r1, Function<ze>6
  10: PutById            r0, r1, 3, "ze"
  11: TryGetById         r2, r0, 1, "window"
  12: GetByIdShort       r1, r0, 2, "ze"
  13: PutById            r2, r1, 4, "onload"
  14: GetByIdShort       r1, r0, 2, "ze"
  15: LoadConstUndefined r4
  16: Call1              r1, r1, r4
     17: CreateRegExp       r1, "dkooDD JPOD D09D
                                       \ @ .

t@ x00 D+D  ", "gmi", 0
  18: PutById            r0, r1, 5, "test2"
  19: TryGetById         r3, r0, 3, "console"
  20: GetByIdShort       r2, r3, 4, "log"
  21: TryGetById         r6, r0, 5, "test2"
  22: GetByIdShort       r5, r6, 6, "exec"
  23: LoadConstString    r1, "bonjour"
  24: Call2              r1, r5, r6, r1
  25: Call2              r1, r2, r3, r1
  26: TryGetById         r5, r0, 3, "console"
  27: GetByIdShort       r3, r5, 4, "log"
  28: GetByIdShort       r1, r0, 7, "gen"
  29: Call1              r2, r1, r4
  30: GetByIdShort       r1, r2, 8, "next"
  31: Call1              r2, r1, r2
  32: LoadConstString    r1, "a"
  33: Call3              r1, r3, r5, r1, r2
  34: TryGetById         r3, r0, 3, "console"
  35: GetByIdShort       r2, r3, 4, "log"
  36: GetByIdShort       r1, r0, 9, "testx"
  37: LoadConstDouble    r5, 7.3
  38: LoadConstString    r0, "le"
  39: Add                r0, r5, r0
  40: Call2              r1, r1, r4, r0
  41: GetByIdShort       r0, r1, 8, "next"
  42: Call1              r1, r0, r1
  43: LoadConstString    r0, "b"
  44: Call3              r0, r2, r3, r0, r1
  45: Ret                r0
```

Unconditional

EXIT

Function 0

```
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_testx>2
  7: LoadThisNS        r0
  8: Call4             r0, r2, r4, r1, r0, r3
  9: Ret               r0
```

Unconditional

EXIT

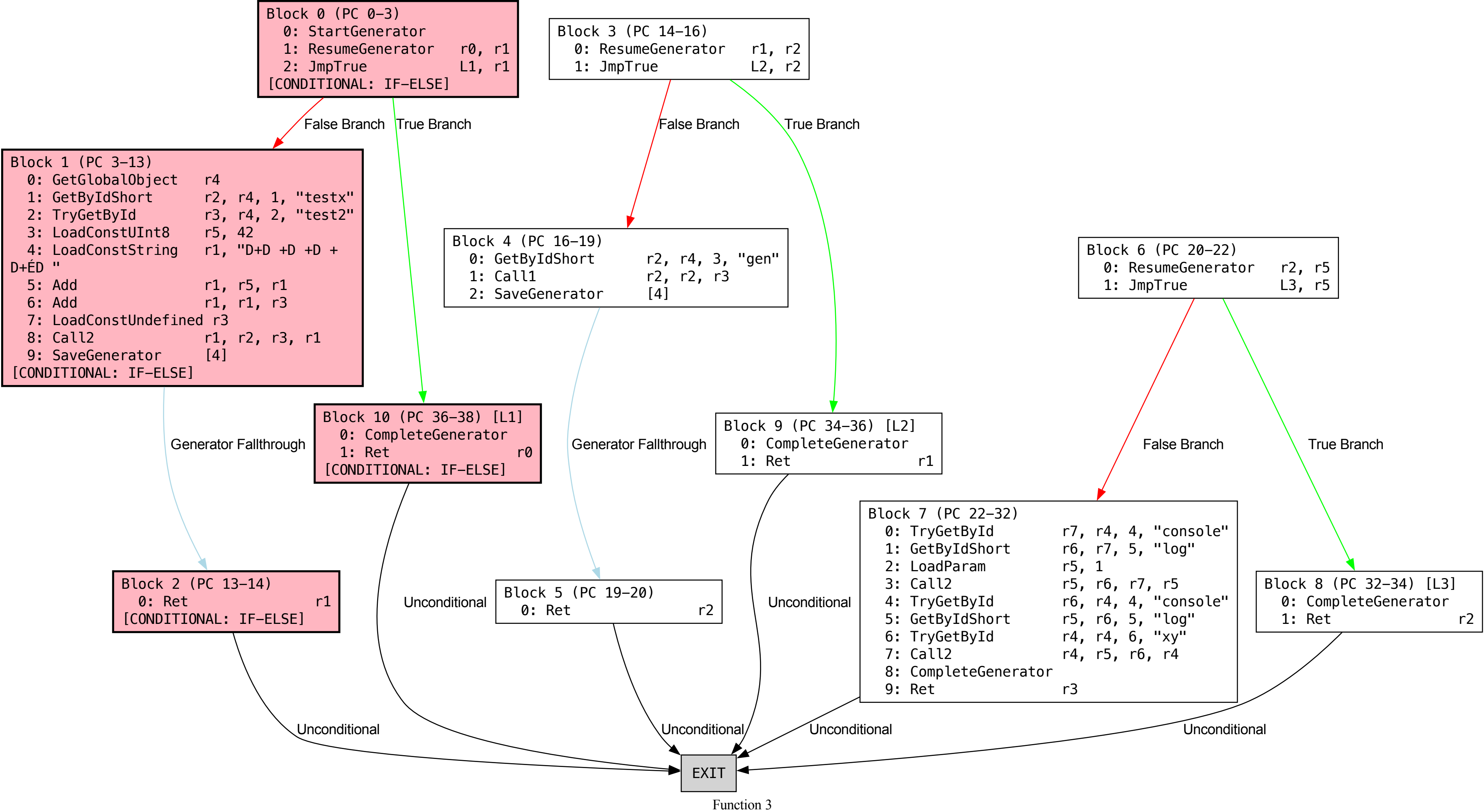Function 1

```
Block 0 (PC 0-3)
  0: CreateEnvironment r0
  1: CreateGenerator    r0, r0, Function<?anon_0_?anon_0_testx>3
  2: Ret                r0
```

Unconditional

EXIT

Function 2

Block 0 (PC 0–3)
```
0: StartGenerator
1: ResumeGenerator    r0, r1
2: JmpTrue            L1, r1
[CONDITIONAL: IF–ELSE]
```

Block 3 (PC 14–16)
```
0: ResumeGenerator    r1, r2
1: JmpTrue            L2, r2
```

False Branch   True Branch

False Branch   True Branch

Block 1 (PC 3–13)
```
0: GetGlobalObject    r4
1: GetByIdShort       r2, r4, 1, "testx"
2: TryGetById         r3, r4, 2, "test2"
3: LoadConstUInt8     r5, 42
4: LoadConstString    r1, "D+D +D +D +
D+ÉD "
5: Add                r1, r5, r1
6: Add                r1, r1, r3
7: LoadConstUndefined r3
8: Call2              r1, r2, r3, r1
9: SaveGenerator      [4]
[CONDITIONAL: IF–ELSE]
```

Block 4 (PC 16–19)
```
0: GetByIdShort       r2, r4, 3, "gen"
1: Call1              r2, r2, r3
2: SaveGenerator      [4]
```

Block 6 (PC 20–22)
```
0: ResumeGenerator    r2, r5
1: JmpTrue            L3, r5
```

Generator Fallthrough

Block 10 (PC 36–38) [L1]
```
0: CompleteGenerator
1: Ret                r0
[CONDITIONAL: IF–ELSE]
```

Generator Fallthrough

Block 9 (PC 34–36) [L2]
```
0: CompleteGenerator
1: Ret                r1
```

False Branch   True Branch

Block 7 (PC 22–32)
```
0: TryGetById         r7, r4, 4, "console"
1: GetByIdShort       r6, r7, 5, "log"
2: LoadParam          r5, 1
3: Call2              r5, r6, r7, r5
4: TryGetById         r6, r4, 4, "console"
5: GetByIdShort       r5, r6, 5, "log"
6: TryGetById         r4, r4, 6, "xy"
7: Call2              r4, r5, r6, r4
8: CompleteGenerator
9: Ret                r3
```

Block 2 (PC 13–14)
```
0: Ret                r1
[CONDITIONAL: IF–ELSE]
```

Unconditional

Block 5 (PC 19–20)
```
0: Ret                r2
```

Unconditional

Unconditional

Block 8 (PC 32–34) [L3]
```
0: CompleteGenerator
1: Ret                r2
```

Unconditional   Unconditional   Unconditional   Unconditional

EXIT

Function 3

```
Block 0 (PC 0–3)
  0: CreateEnvironment  r0
  1: CreateGenerator    r0, r0, Function<?anon_0_gen>5
  2: Ret                r0
```

Unconditional

EXIT

Function 4

Function 5
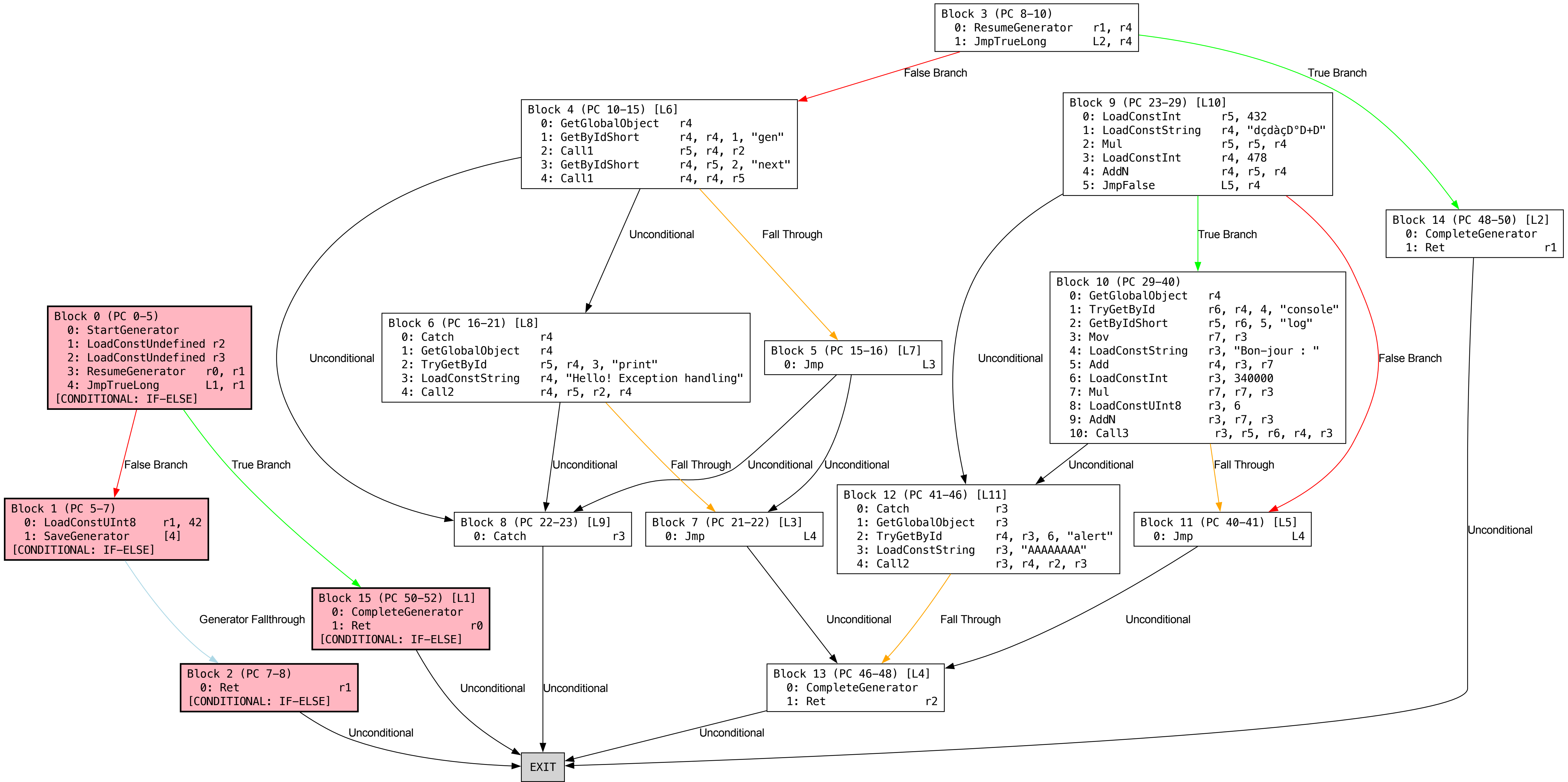
```
Block 0 (PC 0-7)
  0: LoadConstUndefined r0
  1: CreateEnvironment r3
  2: GetGlobalObject   r1
  3: TryGetById        r4, r1, 1, "Math"
  4: GetByIdShort      r2, r4, 2, "random"
  5: Call1             r2, r2, r4
  6: JmpTrue           L1, r2
[CONDITIONAL: IF-ELSE]
```
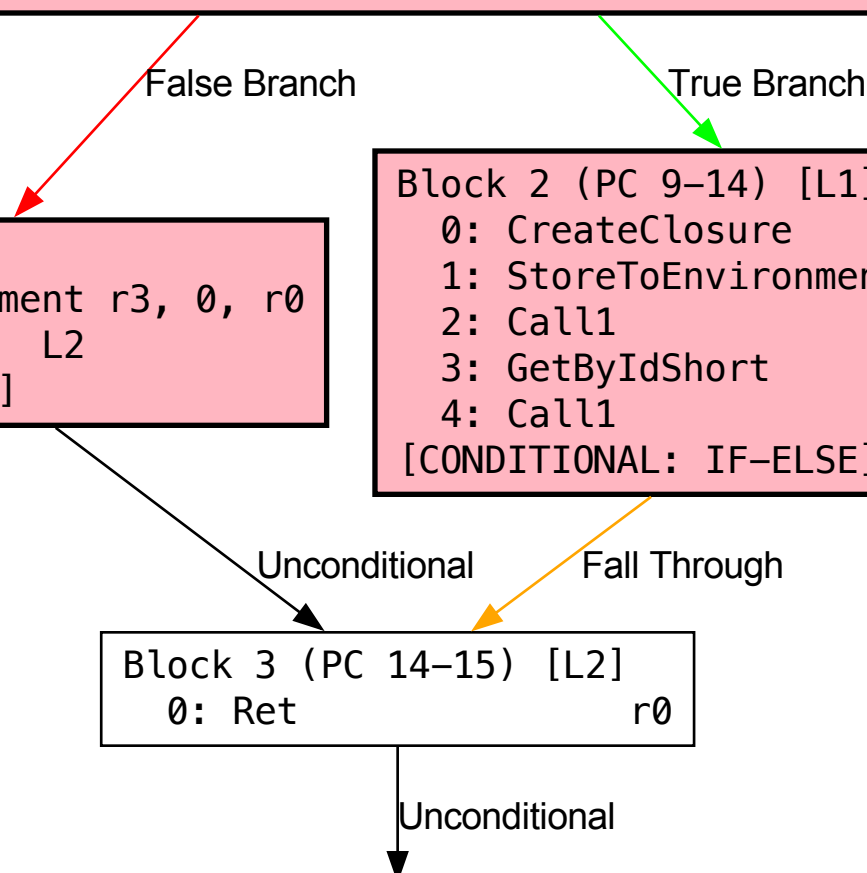
False Branch

True Branch

```
Block 1 (PC 7-9)
  0: StoreNPToEnvironment r3, 0, r0
  1: Jmp                   L2
[CONDITIONAL: IF-ELSE]
```

```
Block 2 (PC 9-14) [L1]
  0: CreateClosure     r2, r3, Function<zb>7
  1: StoreToEnvironment r3, 0, r2
  2: Call1             r2, r2, r0
  3: GetByIdShort      r1, r1, 3, "ze"
  4: Call1             r1, r1, r0
[CONDITIONAL: IF-ELSE]
```

Unconditional

Fall Through

```
Block 3 (PC 14-15) [L2]
  0: Ret               r0
```

Unconditional

EXIT

Function 6

```
Block 0 (PC 0–5)
  0: GetGlobalObject    r0
  1: TryGetById         r1, r0, 1, "Date"
  2: GetByIdShort       r0, r1, 2, "now"
  3: Call1              r0, r0, r1
  4: JmpFalse           L1, r0
[CONDITIONAL: IF–ELSE]
```

True Branch

```
Block 1 (PC 5–9)
  0: GetEnvironment      r0, 0
  1: LoadFromEnvironment r1, r0, 0
  2: LoadConstUndefined r0
  3: Call1               r0, r1, r0
[CONDITIONAL: IF–ELSE]
```

False Branch

Fall Through

```
Block 2 (PC 9–11) [L1]
  0: LoadConstUndefined r0
  1: Ret                r0
```

Unconditional

EXIT

Function 7