```
Block 0 (PC 0-3)
  0: LoadConstUndefined r0
  1: AsyncBreakCheck
  2: Ret                  r0
```
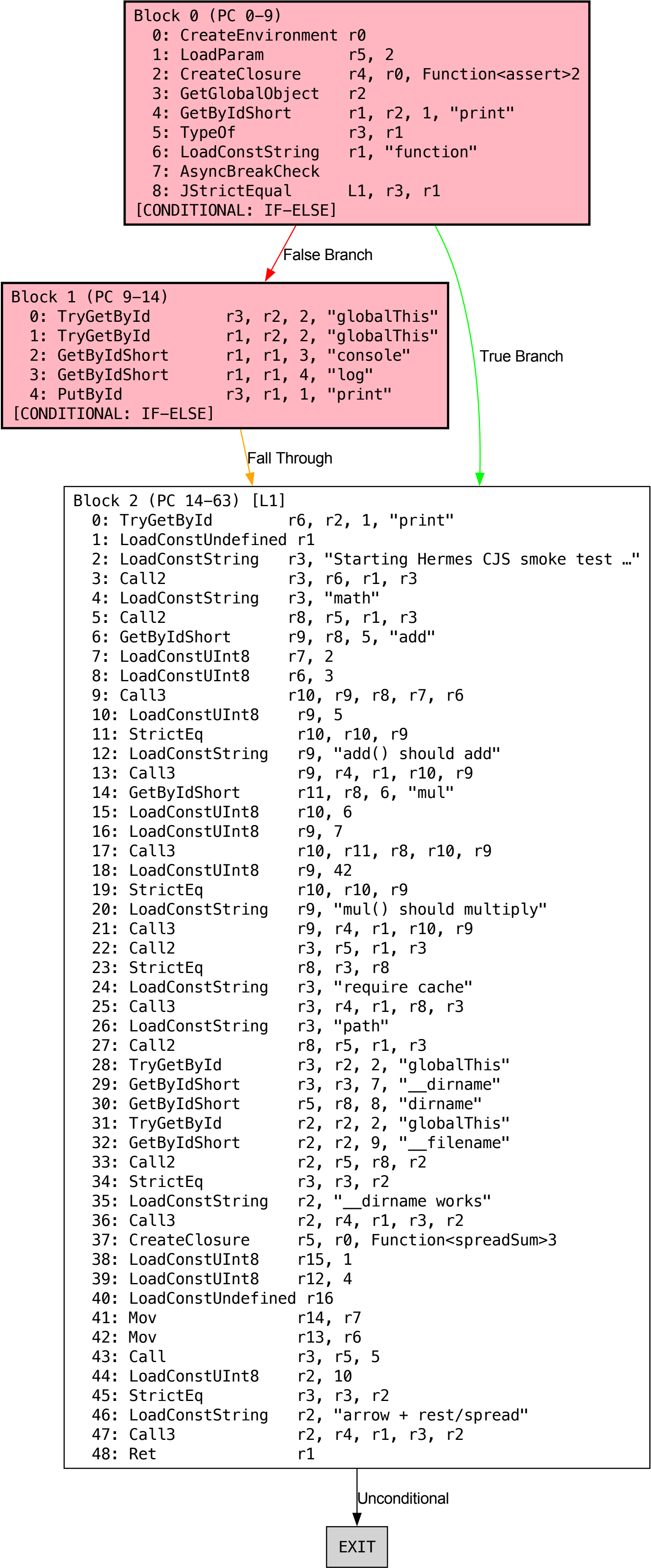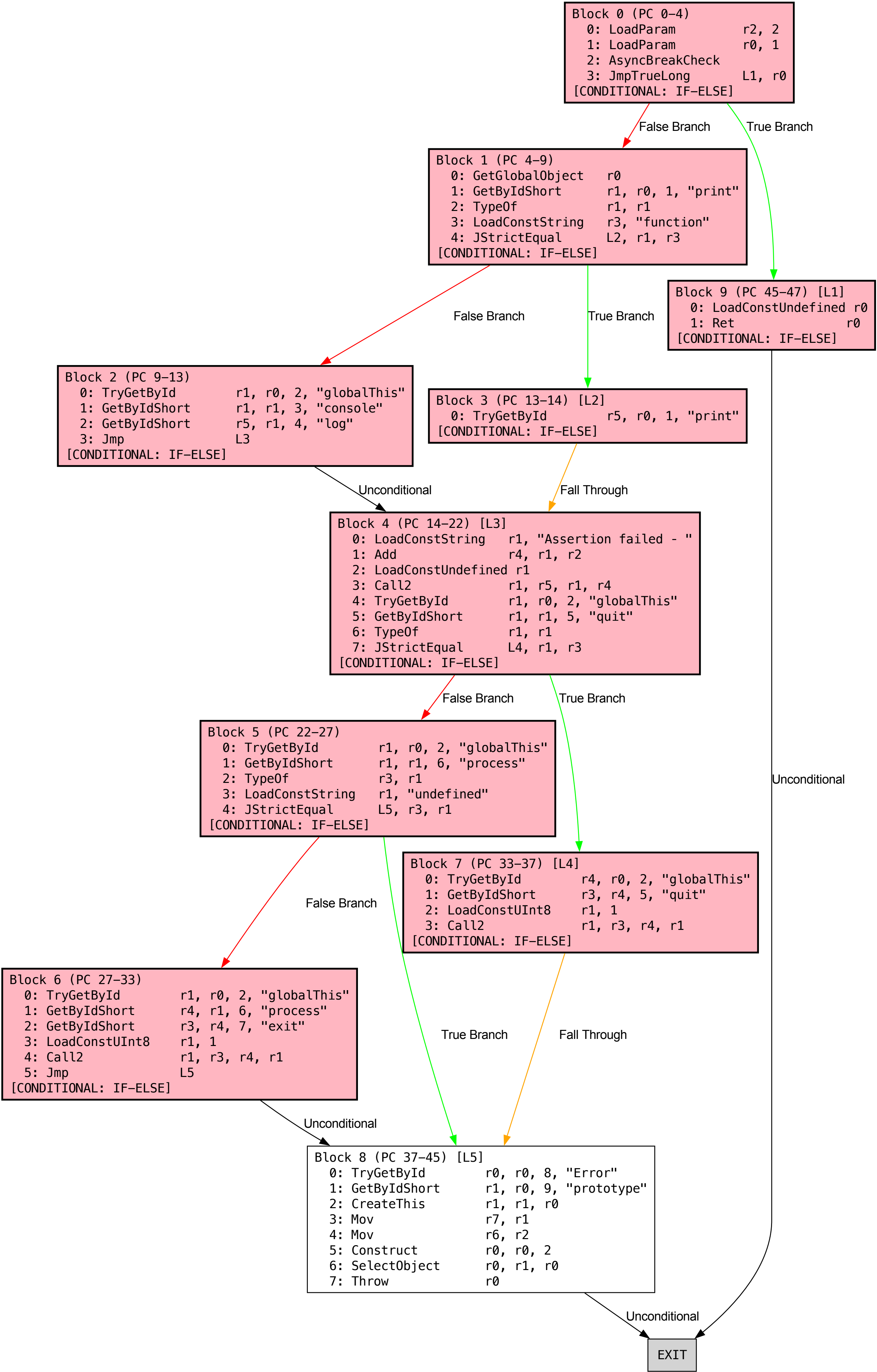
Unconditional

EXIT

Function 0

```
Block 0 (PC 0-9)
    0: CreateEnvironment r0
    1: LoadParam        r5, 2
    2: CreateClosure     r4, r0, Function<assert>2
    3: GetGlobalObject   r2
    4: GetByIdShort      r1, r2, 1, "print"
    5: TypeOf            r3, r1
    6: LoadConstString   r1, "function"
    7: AsyncBreakCheck
    8: JStrictEqual      L1, r3, r1
[CONDITIONAL: IF-ELSE]
```

False Branch

True Branch

```
Block 1 (PC 9-14)
  0: TryGetById         r3, r2, 2, "globalThis"
  1: TryGetById         r1, r2, 2, "globalThis"
  2: GetByIdShort       r1, r1, 3, "console"
  3: GetByIdShort       r1, r1, 4, "log"
  4: PutById            r3, r1, 1, "print"
[CONDITIONAL: IF-ELSE]
```

Fall Through

```
Block 2 (PC 14-63) [L1]
   0: TryGetById         r6, r2, 1, "print"
   1: LoadConstUndefined r1
   2: LoadConstString    r3, "Starting Hermes CJS smoke test …"
   3: Call2              r3, r6, r1, r3
   4: LoadConstString    r3, "math"
   5: Call2              r8, r5, r1, r3
   6: GetByIdShort       r9, r8, 5, "add"
   7: LoadConstUInt8     r7, 2
   8: LoadConstUInt8     r6, 3
   9: Call3              r10, r9, r8, r7, r6
  10: LoadConstUInt8     r9, 5
  11: StrictEq           r10, r10, r9
  12: LoadConstString    r9, "add() should add"
  13: Call3              r9, r4, r1, r10, r9
  14: GetByIdShort       r11, r8, 6, "mul"
  15: LoadConstUInt8     r10, 6
  16: LoadConstUInt8     r9, 7
  17: Call3              r10, r11, r8, r10, r9
  18: LoadConstUInt8     r9, 42
  19: StrictEq           r10, r10, r9
  20: LoadConstString    r9, "mul() should multiply"
  21: Call3              r9, r4, r1, r10, r9
  22: Call2              r3, r5, r1, r3
  23: StrictEq           r8, r3, r8
  24: LoadConstString    r3, "require cache"
  25: Call3              r3, r4, r1, r8, r3
  26: LoadConstString    r3, "path"
  27: Call2              r8, r5, r1, r3
  28: TryGetById         r3, r2, 2, "globalThis"
  29: GetByIdShort       r3, r3, 7, "__dirname"
  30: GetByIdShort       r5, r8, 8, "dirname"
  31: TryGetById         r2, r2, 2, "globalThis"
  32: GetByIdShort       r2, r2, 9, "__filename"
  33: Call2              r2, r5, r8, r2
  34: StrictEq           r3, r3, r2
  35: LoadConstString    r2, "__dirname works"
  36: Call3              r2, r4, r1, r3, r2
  37: CreateClosure      r5, r0, Function<spreadSum>3
  38: LoadConstUInt8     r15, 1
  39: LoadConstUInt8     r12, 4
  40: LoadConstUndefined r16
  41: Mov                r14, r7
  42: Mov                r13, r6
  43: Call               r3, r5, 5
  44: LoadConstUInt8     r2, 10
  45: StrictEq           r3, r3, r2
  46: LoadConstString    r2, "arrow + rest/spread"
  47: Call3              r2, r4, r1, r3, r2
  48: Ret                r1
```

Unconditional

EXIT

Function 1

```
Block 0 (PC 0-4)
  0: LoadParam        r2, 2
  1: LoadParam        r0, 1
  2: AsyncBreakCheck
  3: JmpTrueLong      L1, r0
[CONDITIONAL: IF-ELSE]
```

False Branch → / True Branch →

```
Block 1 (PC 4-9)
  0: GetGlobalObject  r0
  1: GetByIdShort     r1, r0, 1, "print"
  2: TypeOf           r1, r1
  3: LoadConstString  r3, "function"
  4: JStrictEqual     L2, r1, r3
[CONDITIONAL: IF-ELSE]
```

```
Block 9 (PC 45-47) [L1]
  0: LoadConstUndefined r0
  1: Ret              r0
[CONDITIONAL: IF-ELSE]
```

False Branch / True Branch

```
Block 2 (PC 9-13)
  0: TryGetById       r1, r0, 2, "globalThis"
  1: GetByIdShort     r1, r1, 3, "console"
  2: GetByIdShort     r5, r1, 4, "log"
  3: Jmp              L3
[CONDITIONAL: IF-ELSE]
```

```
Block 3 (PC 13-14) [L2]
  0: TryGetById       r5, r0, 1, "print"
[CONDITIONAL: IF-ELSE]
```

Unconditional / Fall Through

```
Block 4 (PC 14-22) [L3]
  0: LoadConstString  r1, "Assertion failed - "
  1: Add              r4, r1, r2
  2: LoadConstUndefined r1
  3: Call2            r1, r5, r1, r4
  4: TryGetById       r1, r0, 2, "globalThis"
  5: GetByIdShort     r1, r1, 5, "quit"
  6: TypeOf           r1, r1
  7: JStrictEqual     L4, r1, r3
[CONDITIONAL: IF-ELSE]
```

False Branch / True Branch

```
Block 5 (PC 22-27)
  0: TryGetById       r1, r0, 2, "globalThis"
  1: GetByIdShort     r1, r1, 6, "process"
  2: TypeOf           r3, r1
  3: LoadConstString  r1, "undefined"
  4: JStrictEqual     L5, r3, r1
[CONDITIONAL: IF-ELSE]
```

```
Block 7 (PC 33-37) [L4]
  0: TryGetById       r4, r0, 2, "globalThis"
  1: GetByIdShort     r3, r4, 5, "quit"
  2: LoadConstUInt8   r1, 1
  3: Call2            r1, r3, r4, r1
[CONDITIONAL: IF-ELSE]
```

False Branch

```
Block 6 (PC 27-33)
  0: TryGetById       r1, r0, 2, "globalThis"
  1: GetByIdShort     r4, r1, 6, "process"
  2: GetByIdShort     r3, r4, 7, "exit"
  3: LoadConstUInt8   r1, 1
  4: Call2            r1, r3, r4, r1
  5: Jmp              L5
[CONDITIONAL: IF-ELSE]
```

True Branch / Fall Through

Unconditional

```
Block 8 (PC 37-45) [L5]
  0: TryGetById       r0, r0, 8, "Error"
  1: GetByIdShort     r1, r0, 9, "prototype"
  2: CreateThis       r1, r1, r0
  3: Mov              r7, r1
  4: Mov              r6, r2
  5: Construct        r0, r0, 2
  6: SelectObject     r0, r1, r0
  7: Throw            r0
```

Unconditional

EXIT

Unconditional

Function 2

```
Block 0 (PC 0–9)
  0: LoadConstZero    r4
  1: LoadConstZero    r6
  2: CallBuiltin      r3, 45, 2
  3: GetByIdShort     r2, r3, 1, "reduce"
  4: CreateEnvironment r0
  5: CreateClosure    r1, r0, Function<>4
  6: Call3            r1, r2, r3, r1, r4
  7: AsyncBreakCheck
  8: Ret              r1
```

Unconditional

EXIT

Function 3

```
Block 0 (PC 0–5)
  0: LoadParam          r1, 1
  1: LoadParam          r0, 2
  2: Add                r0, r1, r0
  3: AsyncBreakCheck
  4: Ret                r0
```

Unconditional

EXIT

Function 4

```
Block 0 (PC 0–11)
  0: CreateEnvironment r0
  1: CreateClosure      r2, r0, Function<add>7
  2: LoadParam          r1, 1
  3: PutById            r1, r2, 1, "add"
  4: LoadParam          r1, 3
  5: GetByIdShort       r2, r1, 1, "exports"
  6: CreateClosure      r1, r0, Function<mul>6
  7: PutById            r2, r1, 2, "mul"
  8: LoadConstUndefined r1
  9: AsyncBreakCheck
  10: Ret               r1
```

Unconditional

EXIT

Function 5

```
Block 0 (PC 0-5)
  0: LoadParam          r1, 1
  1: LoadParam          r0, 2
  2: Mul                r0, r1, r0
  3: AsyncBreakCheck
  4: Ret                r0
```

Unconditional

EXIT

Function 6

```
Block 0 (PC 0–5)
  0: LoadParam          r1, 1
  1: LoadParam          r0, 2
  2: Add                r0, r1, r0
  3: AsyncBreakCheck
  4: Ret                r0
```

Unconditional

EXIT

Function 7