Fork Language Test Program Documentation
program code with AST diagrams

Document Format:

Test programs are shown below inside a box with the program name on the top of the box. On the next page is an AST written by the fork parser after reading the program. A key is below:

🟥	Keyword
🟦	Operators
🟪	Identifiers
🟨	Literals
🟩	Blocks
⬜	Program

empty_main.fk

```
//This is a comment

void main() {
    //comment
    return;
    //comment
}
main();
```

```
                          ┌─────────────┐
                          │   Program   │
                          └──────┬──────┘
                                 │
                          ┌──────┴──────┐
                          │    Block    │
                          └──────┬──────┘
              ┌──────────────────┼──────────────┬──────────────┐
     ┌────────┴────────┐              ┌──────────┴──────┐  ┌────┴─────────┐
     │    Function     │              │  Function Call  │  │ Nullary Op:  │
     │   Definition    │              │                 │  │   Commit     │
     └────────┬────────┘              └──────────┬──────┘  └──────────────┘
   ┌──────┬───┴────┬──────────┐           ┌──────┴──────┐
┌──┴───┐┌─┴────┐┌──┴─────┐┌───┴───┐  ┌────┴────────┐┌───┴──────┐
│Keyword││Ident.││Function││ Block │  │ Identifier: ││ Function │
│ : void││:main ││Arguments││      │  │    main     ││Arguments │
└───────┘└──────┘└───┬────┘└───┬───┘  └─────────────┘└────┬─────┘
                  ┌──┴──┐  ┌───┴──────┐               ┌───┴──┐
                  │ Null│  │ Keyword: │               │ Null │
                  └─────┘  │  Return  │               └──────┘
                           └────┬─────┘
                           ┌────┴─────┐
                           │Nullary Op:│
                           │  Commit   │
                           └──────────┘
```
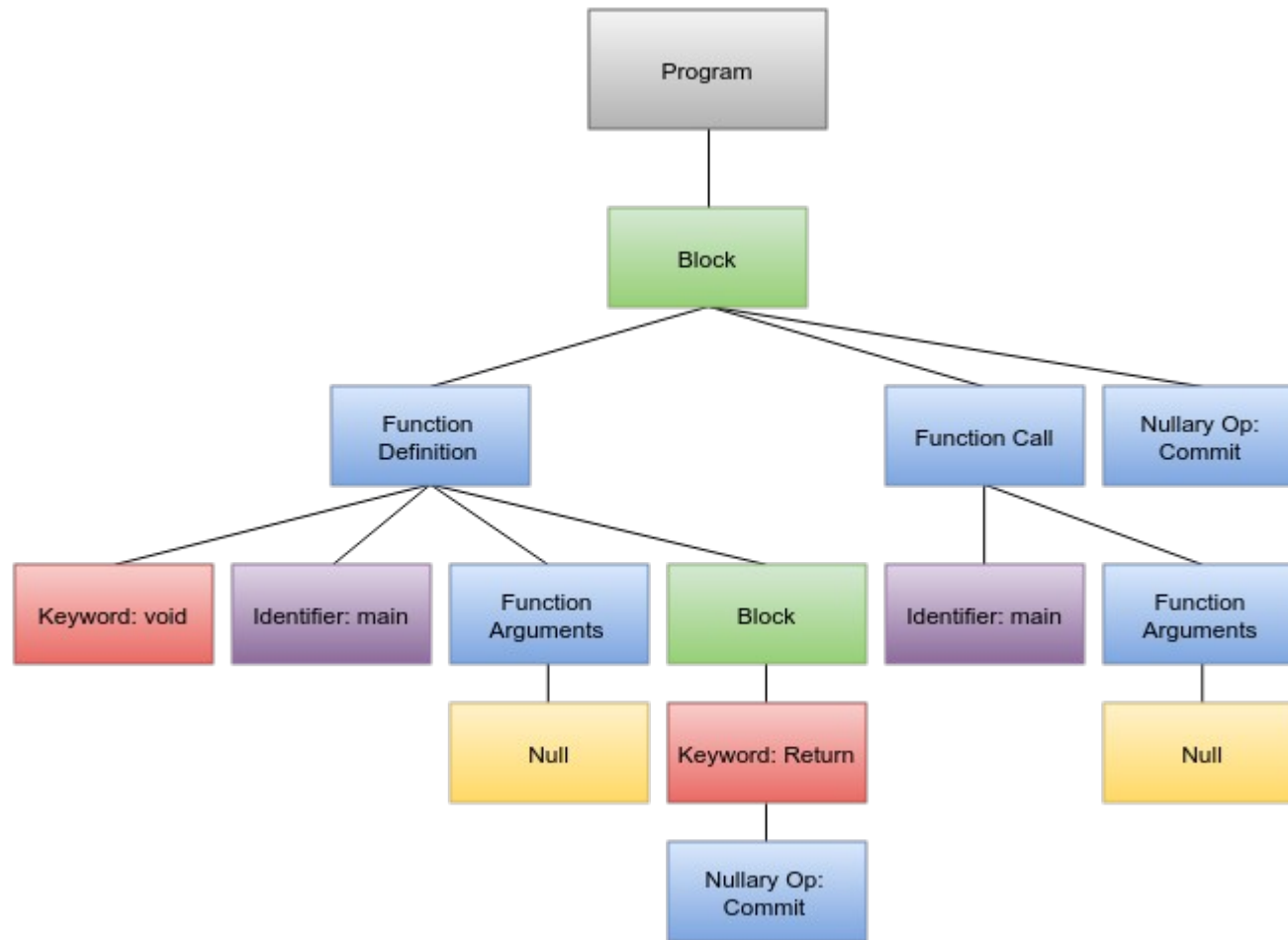
assignment.fk

```
//This is an assignment

void main() {
  int a;
  a = 5;
  return;
}

main();
```

Program

Block

Function Definition

Function Call

Nullary Op: Commit

Keyword: void

Identifier: main

Function Arguments

Block

Identifier: main

Function Arguments

Null

Null

Variable Definition

Binary Op: =

Nullary Op: Commit

Keyword: Return

Keyword: int

Identifier: a

Nullary Op: Commit

Identifier: a

Integer: 5

Nullary Op: Commit
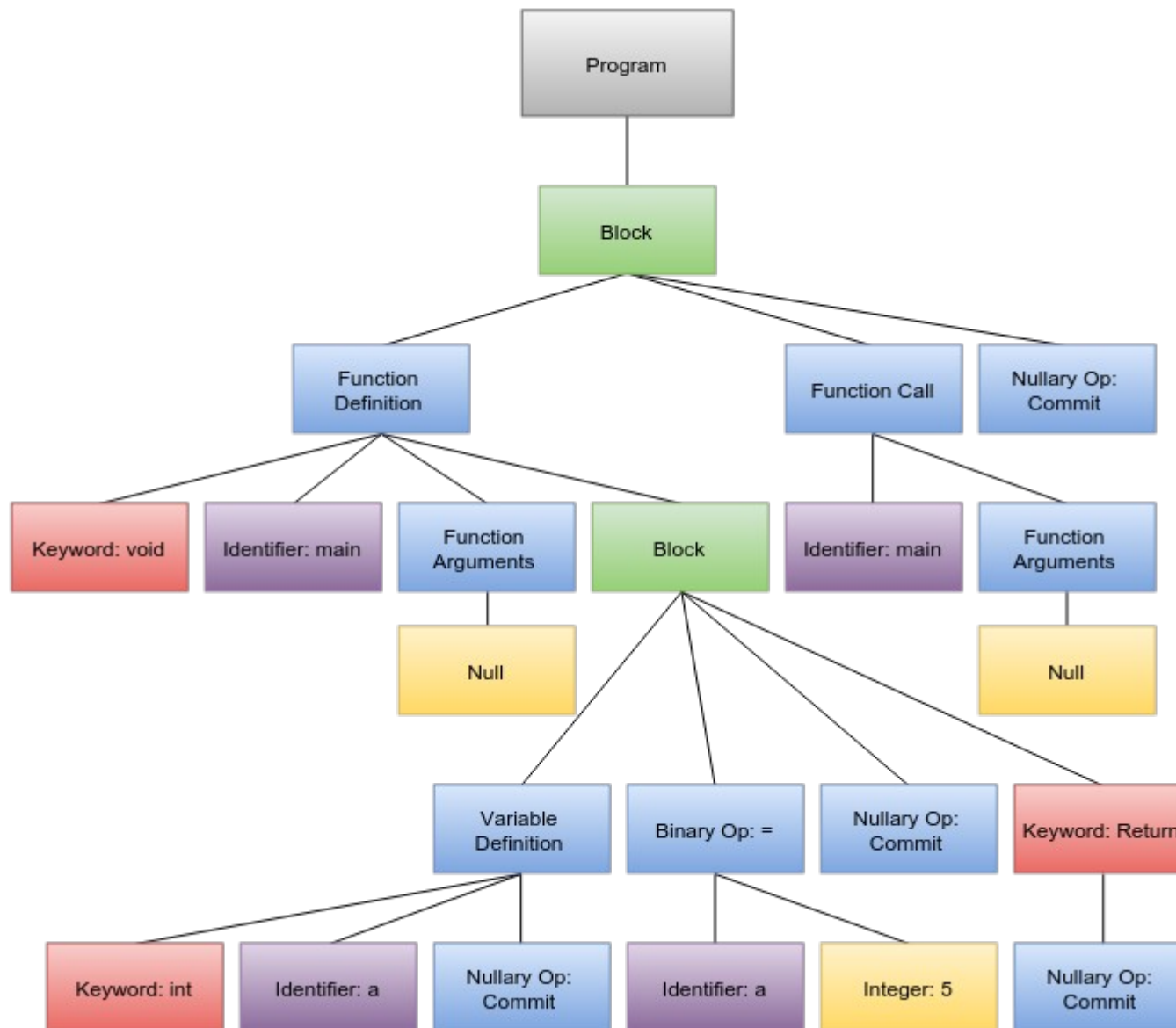
commit.fk

```
void main() {
    6 + 4
    7 * 5;
    return;
}

main();
```

```
Program
  |
Block
  |-- Function Definition
  |     |-- Keyword: void
  |     |-- Identifier: main
  |     |-- Function Arguments
  |     |     |-- Null
  |     |-- Block
  |           |-- Binary Op: +
  |           |     |-- Integer: 6
  |           |     |-- Integer: 4
  |           |-- Binary Op: *
  |           |     |-- Integer: 7
  |           |     |-- Integer: 5
  |           |-- Nullary Op: Commit
  |           |-- Keyword: Return
  |                 |-- Nullary Op: Commit
  |-- Function Call
  |     |-- Identifier: main
  |     |-- Function Arguments
  |           |-- Null
  |-- Nullary Op: Commit
```
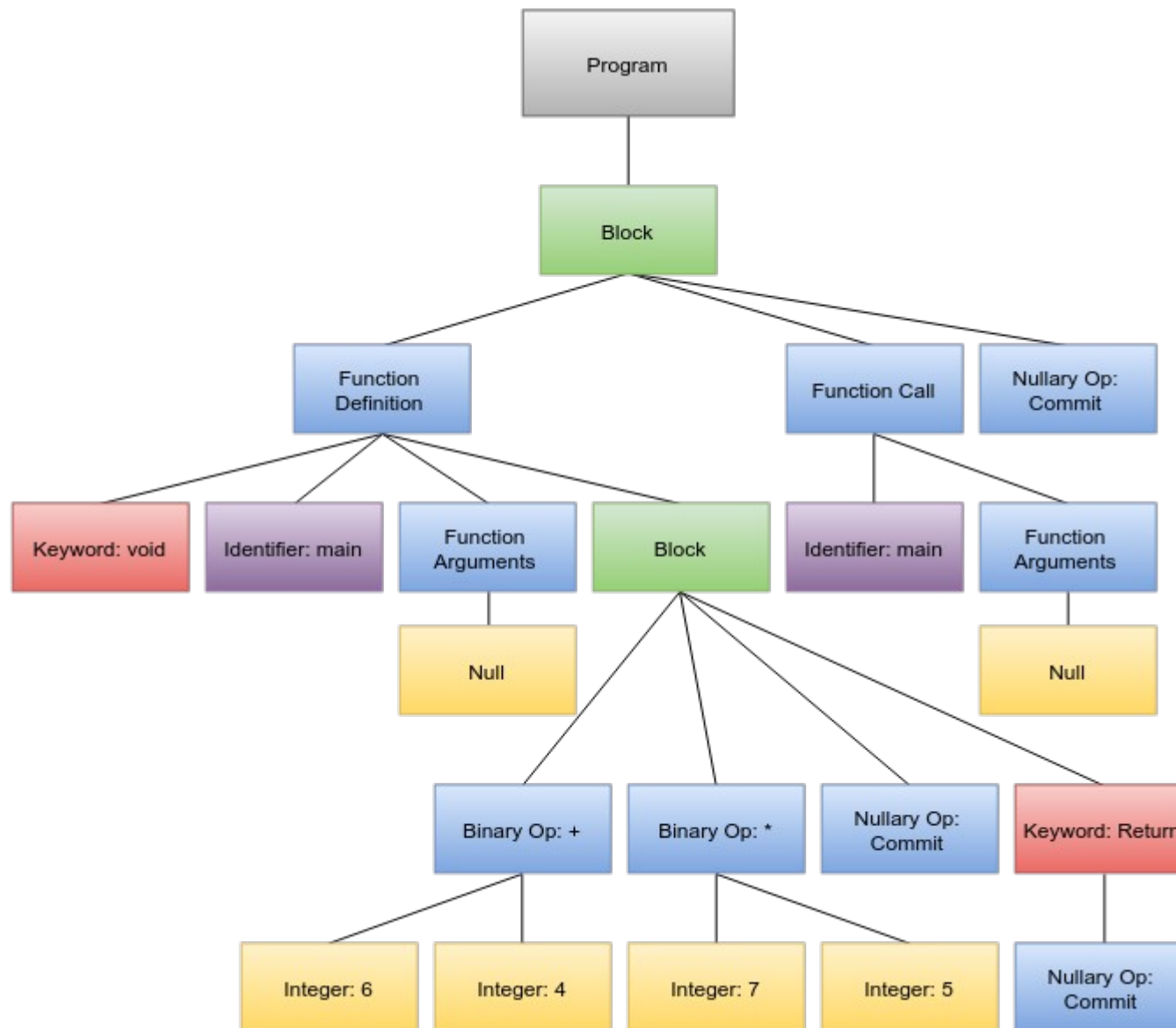
computation.fk

```
void main() {
      5 + 3;
      4 - 2 + 10;
      6 * (4 / 2);
      return;
}
main();
```
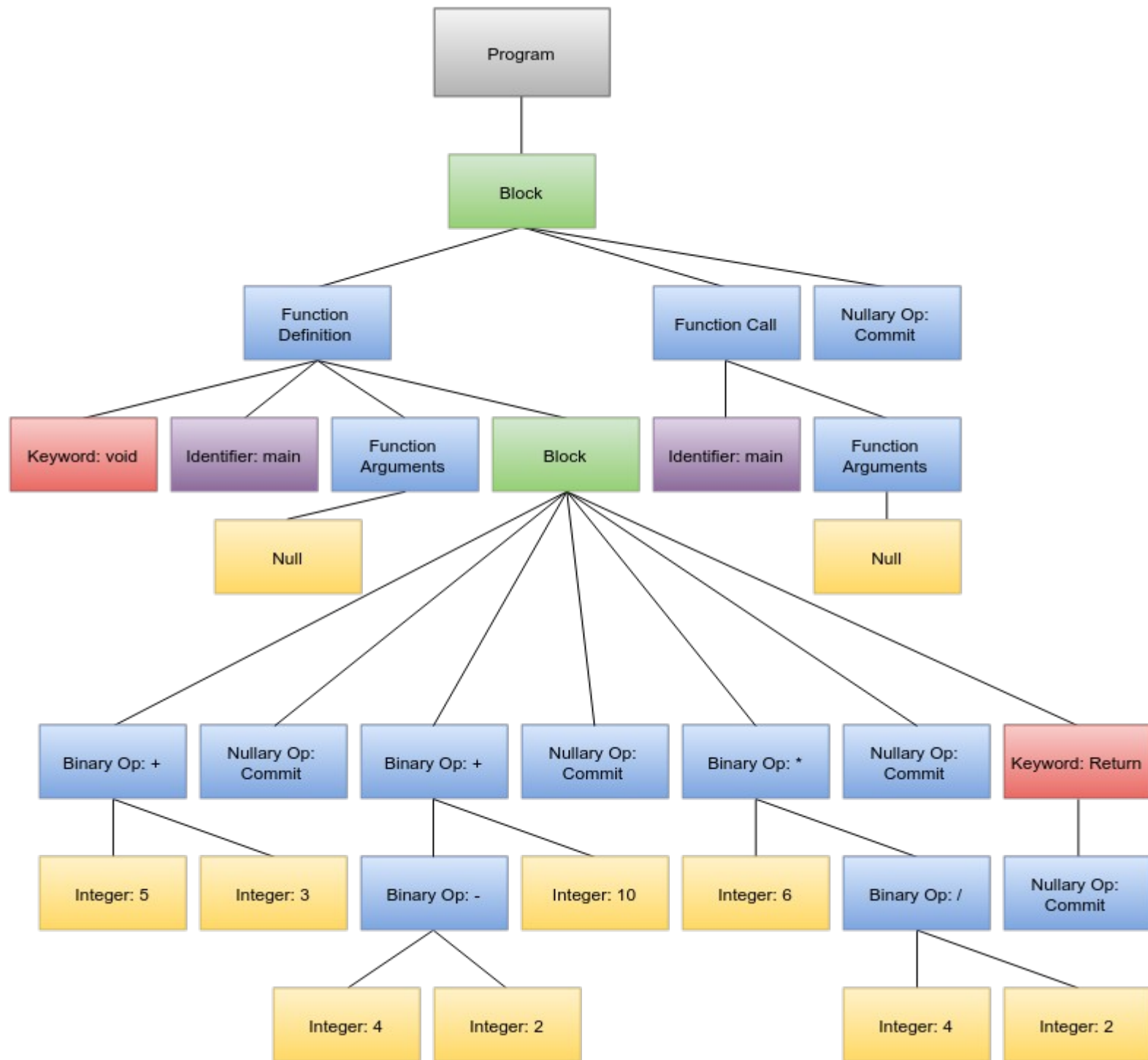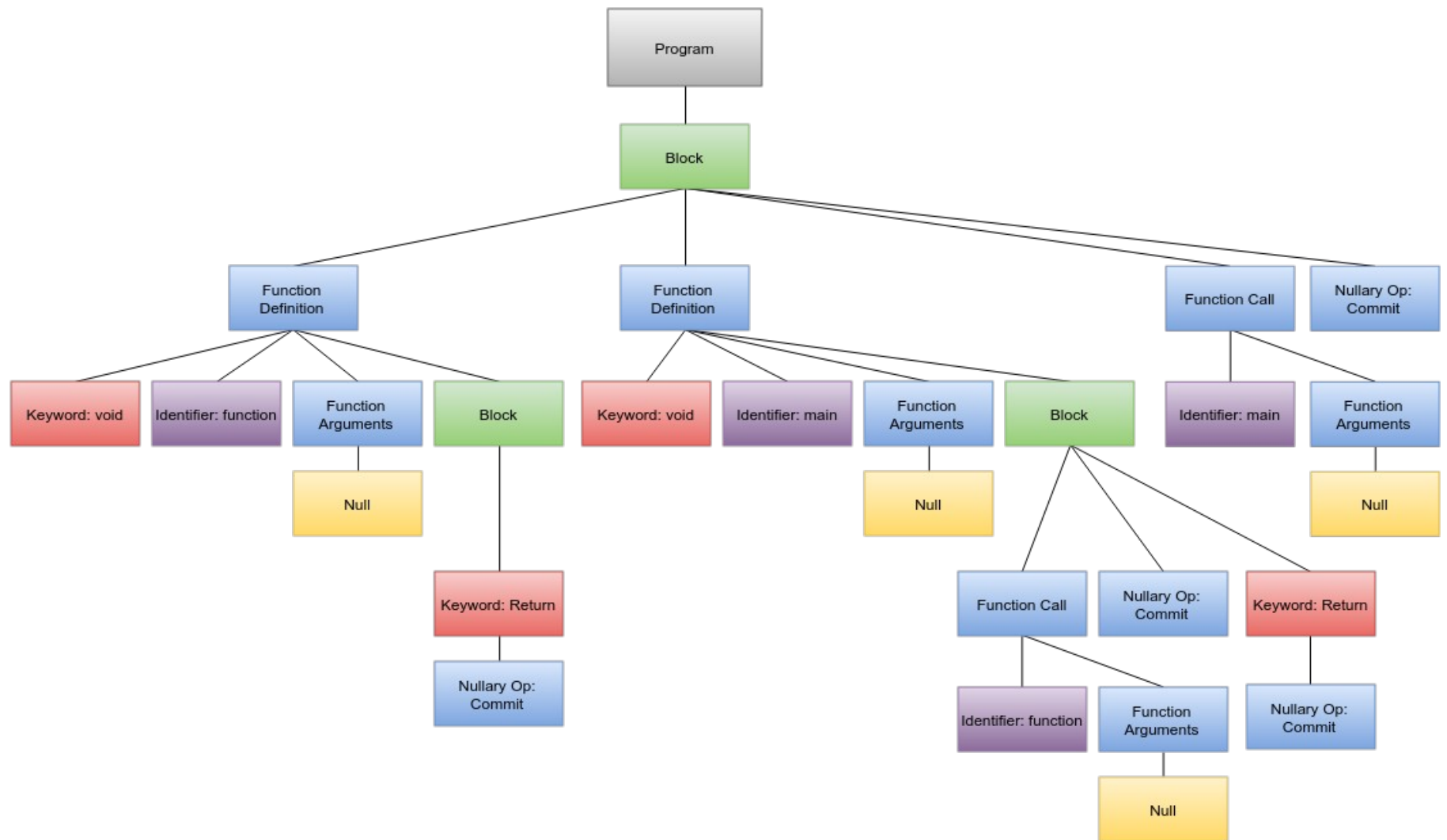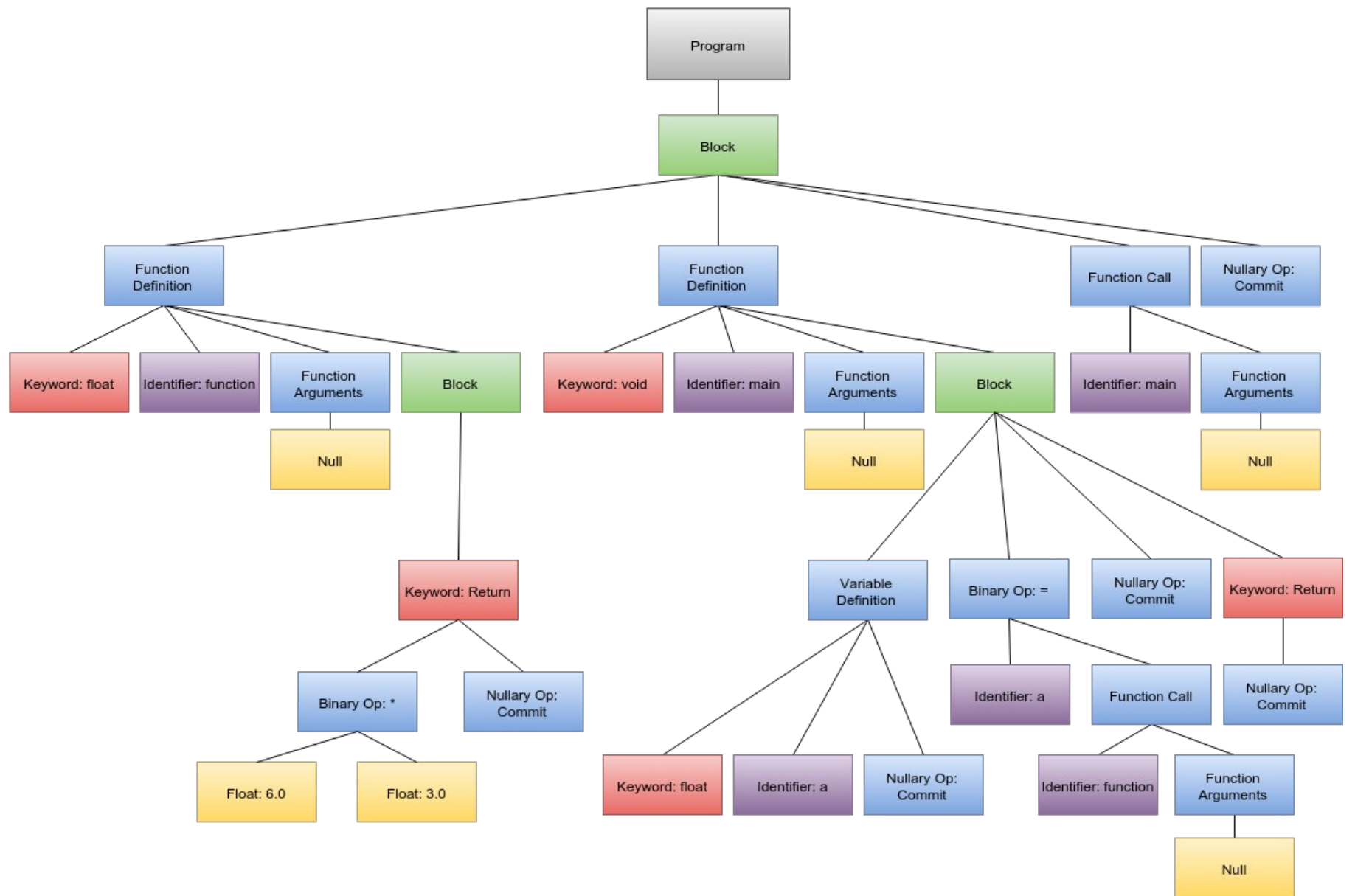
function1.fk

```
void function() {
    return;
}
void main() {
    function();
    return;
}

main();
```

function2.fk

```
float function() {
	return 6.0 * 3.0;
}
void main() {
	float a;
	a = function();
	return;
}
main();
```

function3.fk

```
int function(int a) {
    a = 17;
    return a;
}
void main() {
    int b;
    b = 5;
    b = 4 + function(b);
    return;
}
main();
```