Пример 1.

Вход:

```
function sum(b, c){
     return b +c;
 }

 function ff(){
     var b = 0;
     for(var i = 0, j = 8; (i < 10) || (j > 1); i++, j--)
         b = i ** j;
     return b;
 }

 function swap(a, b){
     var c = b;
     b = a;
     a = c;
 }

 var k = 5;
 var r = sum(k, -1);
 var d = ff();
 swap(r, d);
```

Выход:

```
block
├ function sum
│ ├ block
│ │ └ return
│ │    └ +
│ │      ├ b
│ │      └ c
│ └ args: b, c
├ function ff
│ ├ args:
│ └ block
│   ├ var
│   │ └ b
│   │    └ 0
│   ├ for
│   │ ├ var
│   │ │ ├ i
│   │ │ │ └ 0
│   │ │ └ j
│   │ │    └ 8
│   │ ├ ||
│   │ │ ├ <
│   │ │ │ ├ i
│   │ │ │ └ 10
│   │ │ └ >
│   │ │    ├ j
```

```
|   |   |       └ 1
|   |   ├ block
|   |   | ├ ++
|   |   | | └ i
|   |   | └ --
|   |   |     └ j
|   |   └ =
|   |     ├ b
|   |     └ **
|   |         ├ i
|   |         └ j
|   └ return
|       └ b
├ function swap
| ├ block
| | ├ var
| | | └ c
| | |     └ b
| | ├ =
| | | ├ b
| | | └ a
| | └ =
| |     ├ a
| |     └ c
| └ args: a, b
├ var
| └ k
|     └ 5
├ var
| └ r
|     └ call
|       ├ sum
|       ├ k
|       └ -1
├ var
| └ d
|     └ call
|       └ ff
└ call
  ├ swap
  ├ r
  └ d
```

Пример 2.

Вход:
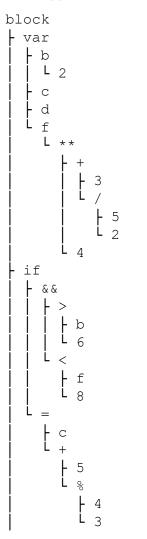
```
var b = 2, c, d, f = (3 + 5/2)**4;
if((b > 6) && (f < 8))
    c = 5 + 4 % 3;
d = 0;
while(d < 3)
{
    d++;
}
do
    c--;
while(c > 0)

for(;;);
for (var i = 0, j = 8; i <= 5 && j > 1; i++, j--)
    for(; b < c;)
        c = b + f;

/* comment*/
```

Выход:

```
block
├ var
│ ├ b
│ │ └ 2
│ ├ c
│ ├ d
│ └ f
│   └ **
│      ├ +
│      │ ├ 3
│      │ └ /
│      │   ├ 5
│      │   └ 2
│      └ 4
├ if
│ ├ &&
│ │ ├ >
│ │ │ ├ b
│ │ │ └ 6
│ │ └ <
│ │   ├ f
│ │   └ 8
│ └ =
│   ├ c
│   └ +
│     ├ 5
│     └ %
│       ├ 4
│       └ 3
│
```

```
├ =
│ ├ d
│ └ 0
├ while
│ ├ <
│ │ ├ d
│ │ └ 3
│ └ block
│     └ ++
│         └ d
├ do while
│ ├ --
│ │ └ c
│ └ >
│     ├ c
│     └ 0
├ for
│ ├ block
│ ├ block
└ for
  ├ var
  │ ├ i
  │ │ └ 0
  │ └ j
  │     └ 8
  ├ &&
  │ ├ <=
  │ │ ├ i
  │ │ └ 5
  │ └ >
  │     ├ j
  │     └ 1
  ├ block
  │ ├ ++
  │ │ └ i
  │ └ --
  │     └ j
  └ for
    ├ block
    ├ <
    │ ├ b
    │ └ c
    ├ block
    └ =
      ├ c
      └ +
          ├ b
          └ f
```