a && b   if a is <u>false</u>, b is not evaluated expression "returns" <u>false</u>.

if ((E instanceof Type) && ((Type)(E)). isMutable())

a ⏜ (brace) b ⏜ (brace)

will fail
if E is not a type.

a || b   if a is <u>true</u>, b is not evaluated expression "returns" true.

(And(Code 置 , Boolean, Boolean, BOOLEAN))

a

value code for a (1st argument)

JZero falseLabel

b

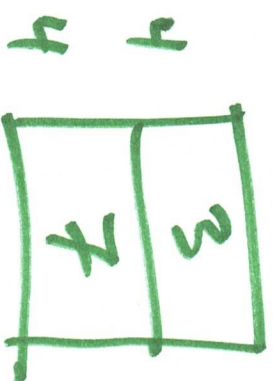value code for b (2nd argument)

Jump endLabel
Label falseLabel
PushI 0
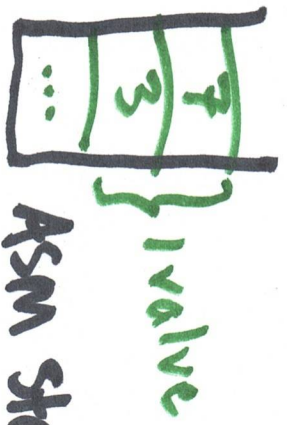Label endLabel.

```
if (running)
    a = 1;
```

~~`if (running)`~~ no! ✗✗

```
if (running) {
    a = 1;
}
```

```
<int>
```

```
mut x := <3..7>;
            int int
```

`int*int → range[int]`
x

`mut y := <x+14 .. 3*x*x*x + 5*x>;`

range[int] is 8 bytes.

4	7
4	3

ASM stack.

Value: 7, 3, ...

Exch.

addr → addr + 4
int range.

Push addr
Exch
Store I

addr / a → a / addr   Store.

Push Addr + 4
Exch
Store I
Push Addr Exch
Store I

Do not generate addr twice — use temp storage.