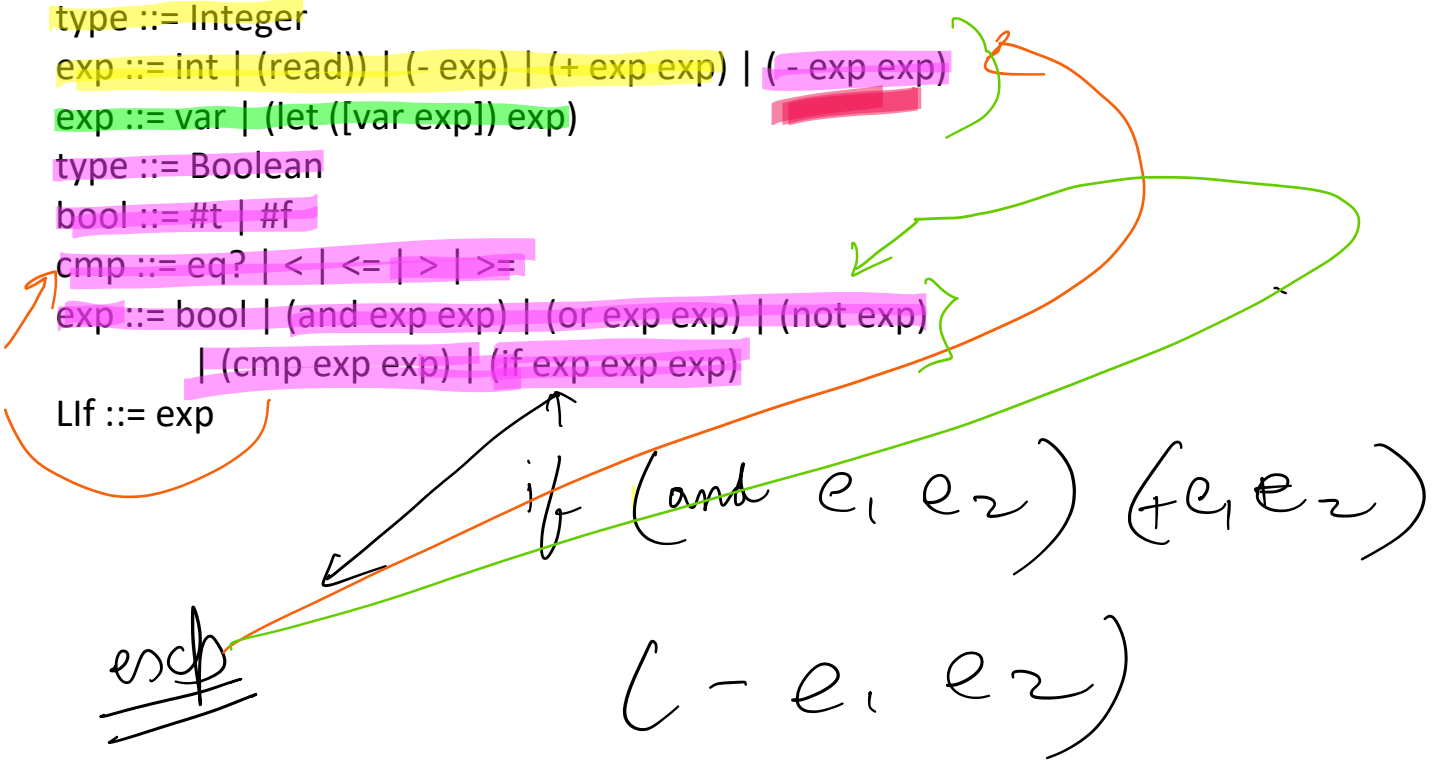
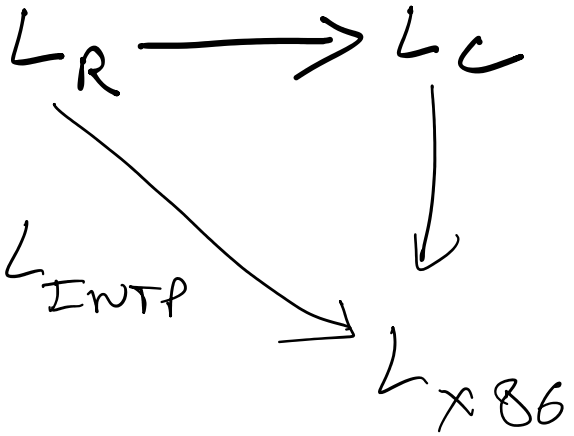
R2

16 February 2022 13:12

```
type ::= Integer
exp ::= int | (read)) | (- exp) | (+ exp exp) | ( - exp exp)
exp ::= var | (let ([var exp]) exp)
type ::= Boolean
bool ::= #t | #f
cmp ::= eq? | < | <= | > | >=
exp ::= bool | (and exp exp) | (or exp exp) | (not exp)
        | (cmp exp exp) | (if exp exp exp)
LIf ::= exp
```





If e1 e2 e3

E1? Int or bool?
E2? Bool or int ?

semantics

16 February 2022 13:16

Type checking

(not 1) (#f)

#f (not #f)

(A-Z)* → #t #t

(#f) → #f

(not 1) → error

(not #t) } ✓
(not #f)

(not ±) ✗

if (not 1) ✓
(car 1) ?

error → pair or list

no check	static/ comp. type	dynamic run time type
C	ada	java

a[11] → java C?

R2

R2

let var exp_1 exp_2
if exp_3 exp_4 exp_5

$e_1 : I | B$ $e_5 :$

$e_2 : I | B$

$e_3 : B$

$e_4 = e_5 : I | B$

Shrinking

Thursday, 17 February 2022 9:29 AM

(- e1 e2)

sub

(+ e1 (- e2))

eq? < (> e1 e2)

→ (< e2 e1) (- e2 e1)

— (read) (read) → bug

(>= e1 e2) → (not (< e2 e1))

(or (< e2 e1) (eq? e1 e2))

(let [temp e1] (< e2 temp))

not

eq? < <= > >=

(let [t1 e1] ((let [t2 e2] t2) (not (t1 t2))))

jmp

cmp

conditional x86

exp_ctrl if

jmp

cmp

xor

not

set

movzbq

not $1, v

cmpq

x86

voodoom