$$\frac{\Gamma(x) = \tau}{\Gamma \vdash \mathtt{x} : \tau}\ (\text{var})$$

$$\frac{\Gamma \vdash e : \mathbf{Bool} \quad \Gamma \vdash e_1 : \tau \quad \Gamma \vdash e_2 : \tau}{\Gamma \vdash \mathtt{if}\ e\ \mathtt{then}\ e_1\ \mathtt{else}\ e_2 : \tau}\ (\text{if})$$

$$\frac{\Gamma, x : \tau_1 \vdash e : \tau_2}{\Gamma \vdash \mathtt{fun}\ x : \tau_1 = e : \tau_1 \to \tau_2}\ (\text{fun})$$

$$\frac{\Gamma \vdash e_1 : \tau_1 \to \tau_2 \quad \Gamma \vdash e_2 : \tau_1}{\Gamma \vdash (e_1 e_2) : \tau_2}\ (\text{appl})$$

① `let f x y =`

```
(match (x 10) with
 | true  -> y
 | false -> [])
```

e

`let f = fun x -> (fun y -> ( ... ))`

'a → 'b x : 'a

$\emptyset, x : 'a \vdash$ `(fun y -> match (x 10) with | true -> y | false -> [])` : 'b

'b = 'c → 'd y : 'c

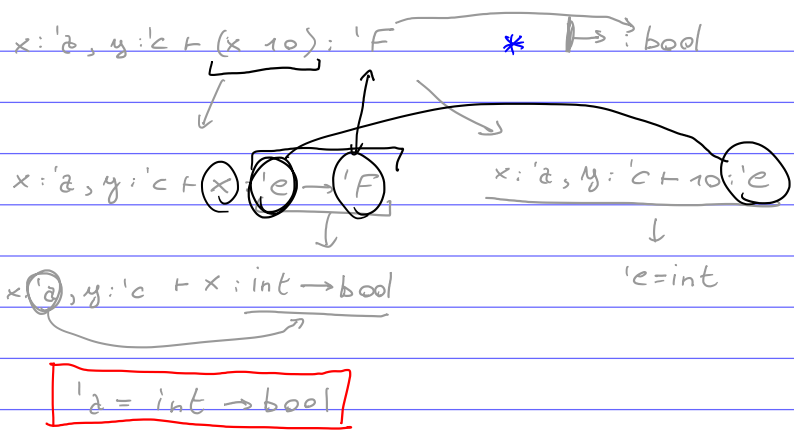
$x : 'a, y : 'c \vdash$ `(match (x 10) with | true -> y | false -> [])` : 'd

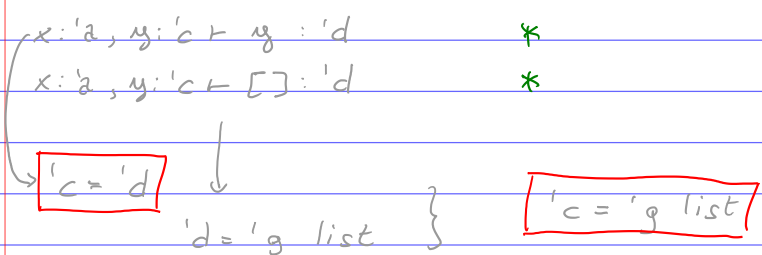
: bool

$x : 'a, y : 'c \vdash (x\ 10) : bool$ *

$x : 'a, y : 'c \vdash y : 'd$ *

$x : 'a, y : 'c \vdash [] : 'd$ *





f : 'a → 'b ??

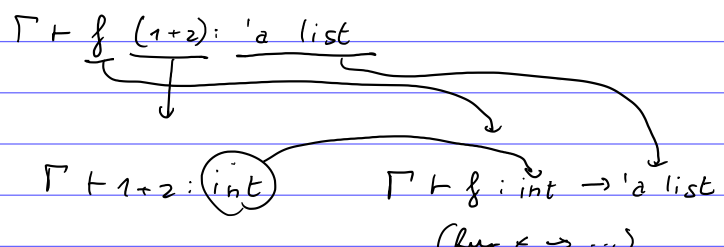
'b = 'c → 'd

'a = int → bool

'c = 'g list

'c = 'd

(int → bool) → 'a list → 'a list



```
let h x y =
  let (w, z) = y true in
    if w then z::x else x;;
```

```
let h = fun x -> (fun y -> (...))
```
$\tau_2 = \tau_3 \to \tau_4$

$h : \tau_1 \to \tau_2$

$x : \tau_1$

$\tau_2 = \tau_3 \to \tau_4$

$y : \tau_3$

```
( let (w, z) = y true in
    if w then z::x else x ) : τ4
```

$(w, z) \equiv$ `y true`

$\tau_5 \quad \tau_5$

$\tau_6 * \tau_7$

$w : \tau_6$

$z : \tau_7$

$y : \text{bool} \to \tau_6 * \tau_7 = \tau_3$

```
if w then z::x else x : τ4
```

`w` : bool — `(z::x)` : τ_4 — `x` : τ_4

$w : \text{bool} = \tau_6$

$x : \tau_4 = \tau_1$

`(z::x)` : τ_4

$z : \tau_2$

$x : \tau_2$ list $= \tau_4$

$\tau_4 = \tau_2$ list

$z : \tau_7$

$x : \tau_7$ list $= \tau_4 = \tau_1$

$h : \text{'a list} \to ((\text{bool} \to \text{bool} * \text{'a}) \to \text{'a list})$

$\tau_1 \qquad \tau_2 \qquad \tau_3$