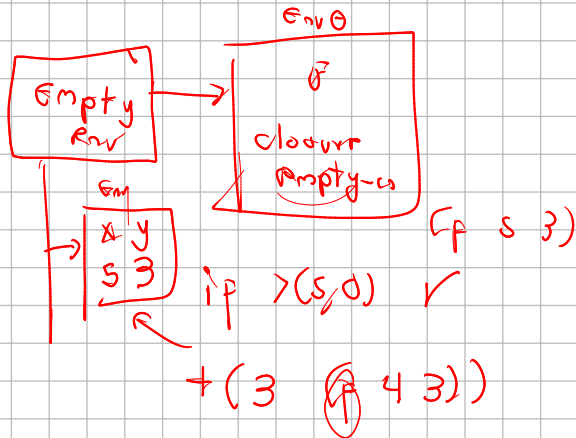


Ambiente vacio

let

```
f = proc(x,y) if >(x,0) (+ y (f sub1(x) y)) else y
in
(f 5 3)
```



let <-- Ligadura local

let

<declaracion>

in

<expresion>

<expresion> se evalua en un extend env que incluye las declaraciones

letrec <-- Definiciones recursivas, exclusivamente para declarar procedimientos

Ambiente inicial vacio

let

```
f = proc(x,y) +(x,y)
g = proc(a,b) +(a,b,2)
in
```

let

```
x = (f (g 1 2) (g 3 2))
y = (g (f 1 3) (f 1 5))
in
```

letrec

```
k(x,y) = if >(x,0) then
          +(y, (k sub1(x) y))
        else y
```

in

(k x y)

156

let

```
f = proc(x,y) +(x,y)
g = proc(a,b) +(a,b,2)
in
```

let

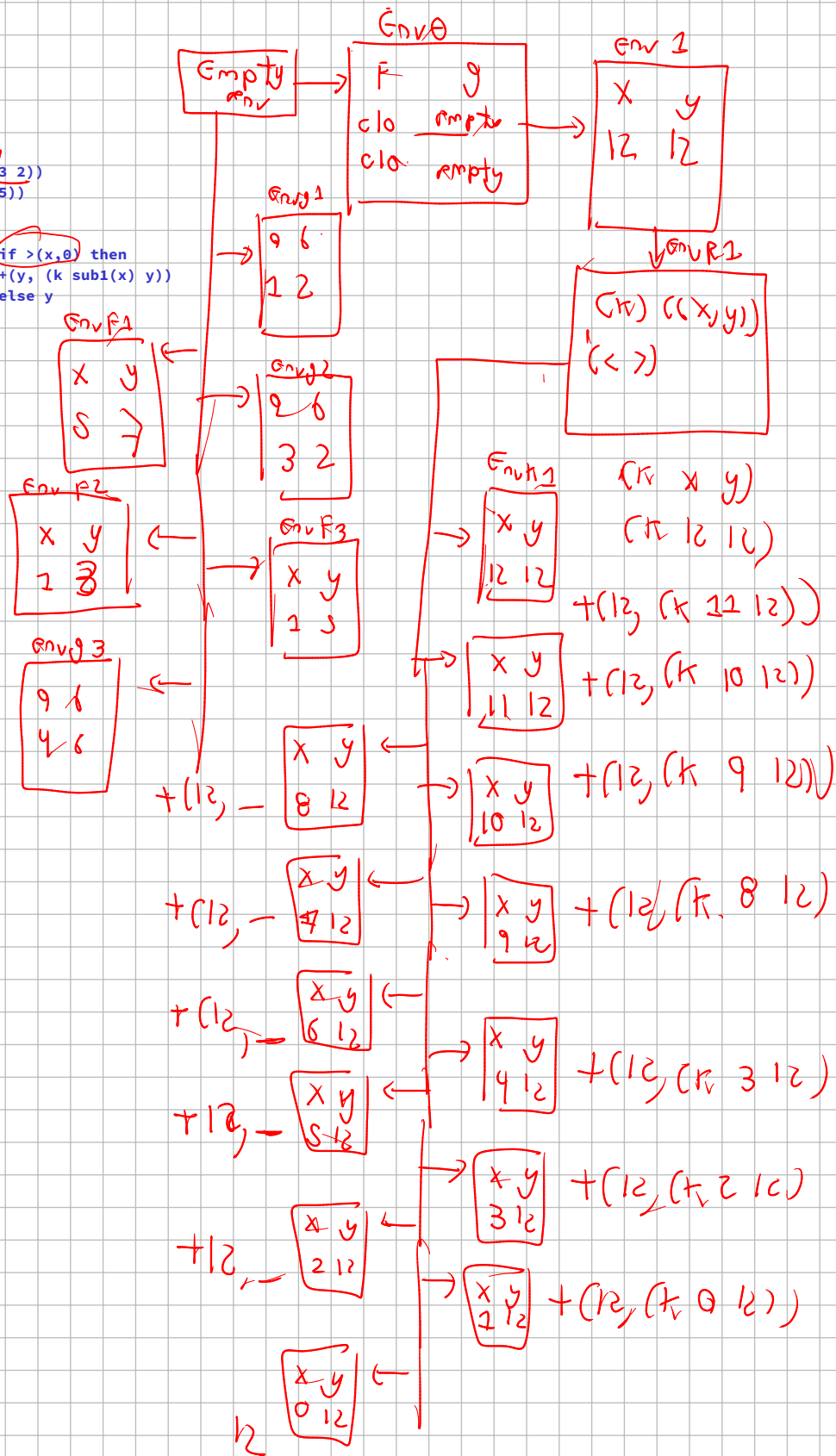
```
x = (f (g 1 2) (g 3 2))
```

```
y = (g (f 1 3) (f 1 5))
```

in

```
letrec
  k(x,y) = if >(x,0) then
    +(y, (k sub1(x) y))
  else y
```

in
(k x y)



12x13

120 + 12x3

120 + 36 = 156