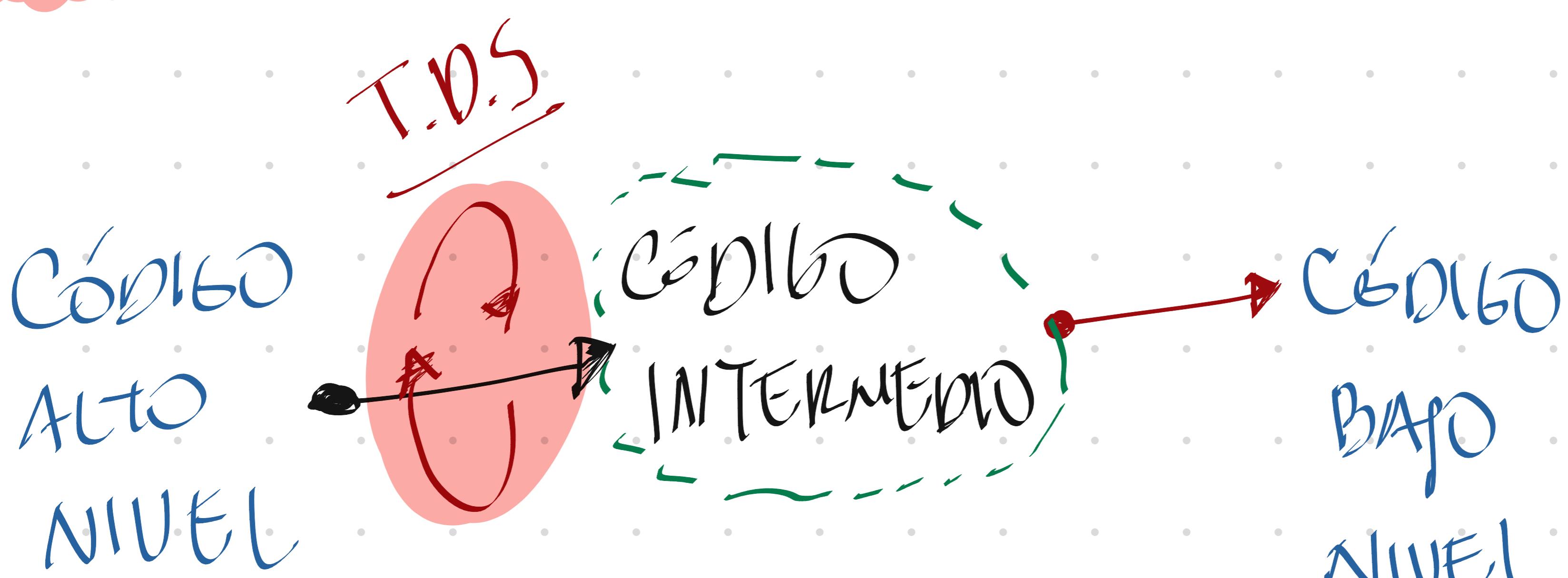


Clase 9
15/01/2018

Unidad 2



class A2


$$t_1 = p11$$

$$t_2 = x * t_1$$

$$t_3 = t_2 + y$$

00001111

11111000

11100011

CÓDIGO
INTERMEDIO

1. GDA
2. Triplets
3. Cuádruplos
4. Cód. p 3 direcciones

Código 3 Direcciones

$x \text{ AND } y \text{ AND } z \text{ OR } w \text{ OR } p$

$t_1 = x \text{ AND } y$

$t_2 = t_1 \text{ AND } z$

$t_3 = t_2 \text{ OR } w$

$t_4 = t_3 \text{ OR } p$

C3D

A

MOV DX, 010B

MOV AH, 09

MOV AH, 00

000111

11011001

Instrucciones (C3D)

1. $x = y$
2. $x = y \text{ op } z$ ($x = y + z$)
3. go to l_x (Salto incondicional)
4. if $x < \text{prel } y$ go to l_x (Salto condicional)
5. call $x;$
6. $x = y[i]$ $x[i] = y$

Expresiones aritméticas

$$a+b$$



$$t_1 = a+b$$

$$a * b$$



$$t_1 = a * b$$



$$a * b + c$$



$$t_1 = a * b$$

$$t_2 = t_1 + c$$

$$a + b * c * d$$



$$t_1 = b * c$$

$$t_2 = t_1 * d$$

$$t_3 = a + t_2$$

$$a + (b * c) + d$$

$$t_1 = b * c$$

$$t_2 = a + t_1$$

$$t_3 = t_2 + d$$

a

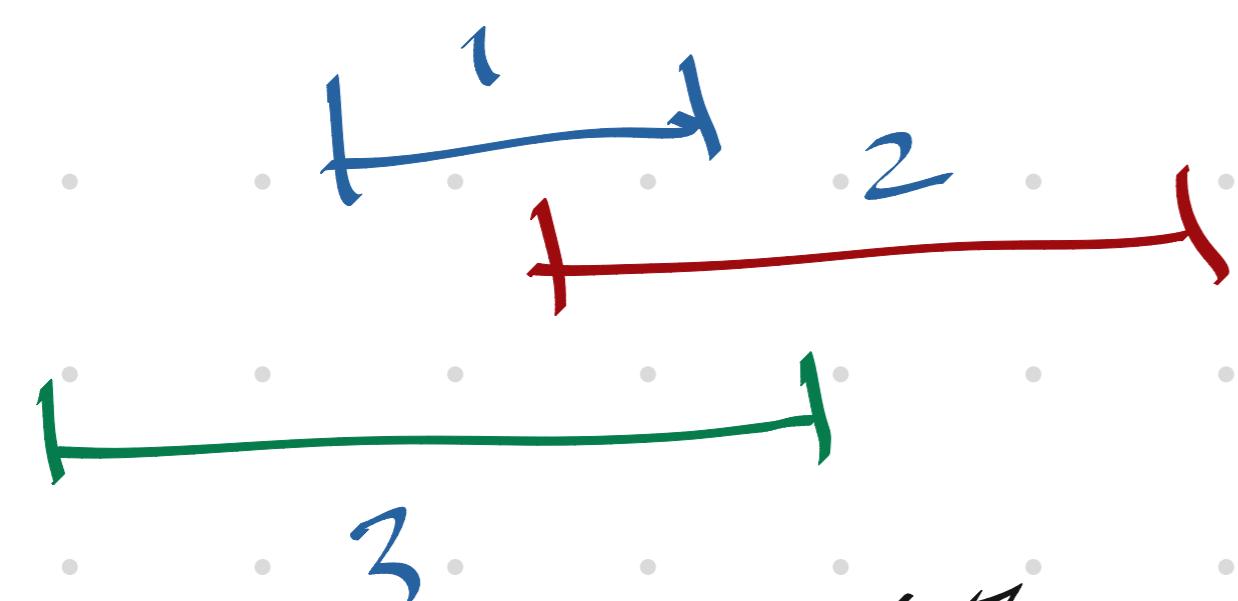


a'

Class 10
Variables



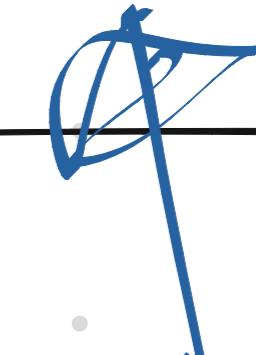
$a + b * c * d$



$$t_1 = b * c$$

$$t_2 = t_1 * d$$

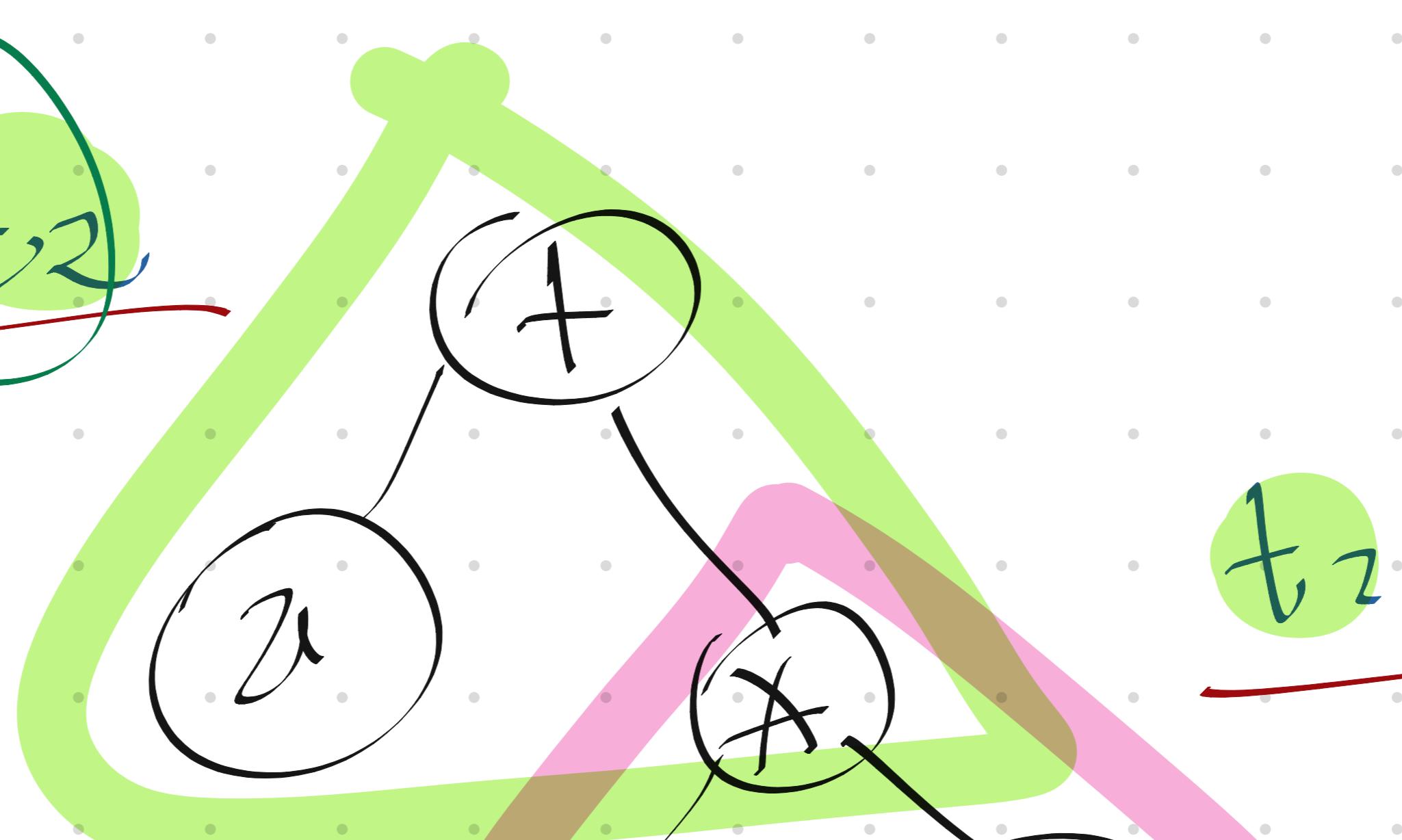
$$t_3 = a + t_2$$



$$\frac{b * c}{c * d} \cancel{t_2}$$

$$t_1 = b * c$$

$$t_3 = a + t_2$$



$$t_1 = b * c$$

getTemp() \rightarrow t_1, t_2, t_3, \dots

t_n

$$t_3 = a + t_2$$

$$t_1 = b * c$$

$$t_2 = t_1 * d$$

$$t_3 = a + t_2$$

$S \rightarrow E \{ \text{print}(E.C3D) \}$

$E \rightarrow E + T \left\{ \begin{array}{l} E.REP = \text{getTemp}() \\ E.C3D = T.C3D + T.REP \\ T.REP += " + " \\ E1.REP += " + " \\ T.REP \end{array} \right\}$

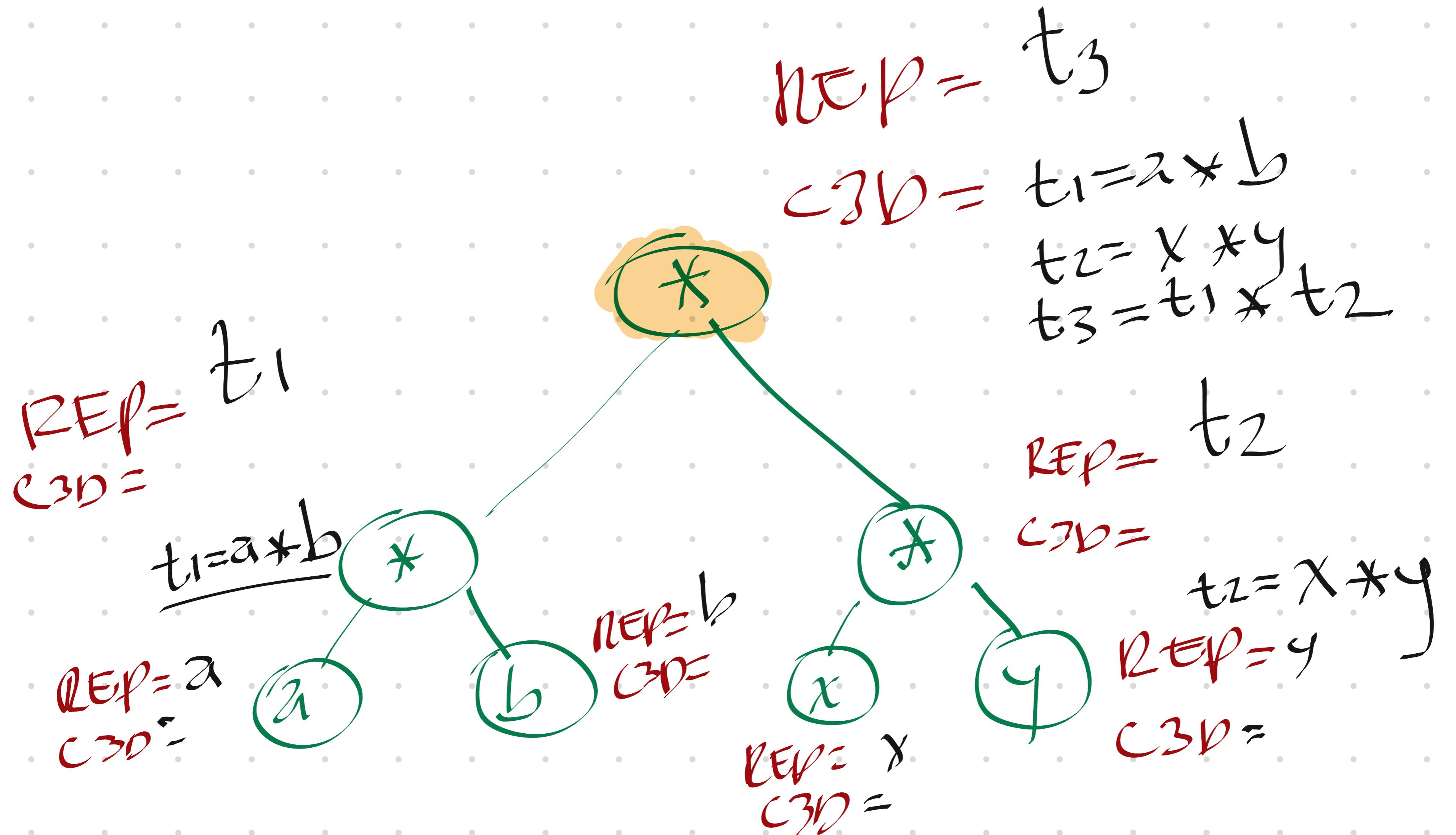
$| T \left\{ \begin{array}{l} E.REP = T.REP \\ E.C3D = T.C3D \end{array} \right\}$

$T \rightarrow T * F \left\{ \begin{array}{l} T.REP = \text{getTemp}() \\ T.C3D = T.C3D + F.C3D \\ T.REP += " * " + T.REP \\ " * " + F.REP \end{array} \right\}$

$| F \left\{ \begin{array}{l} T.REP = F.REP \\ T.C3D = F.C3D \end{array} \right\}$

$F \rightarrow \text{id} \left\{ \begin{array}{l} F.REP = "id" \\ F.C3D = \text{NULL} \end{array} \right\}$

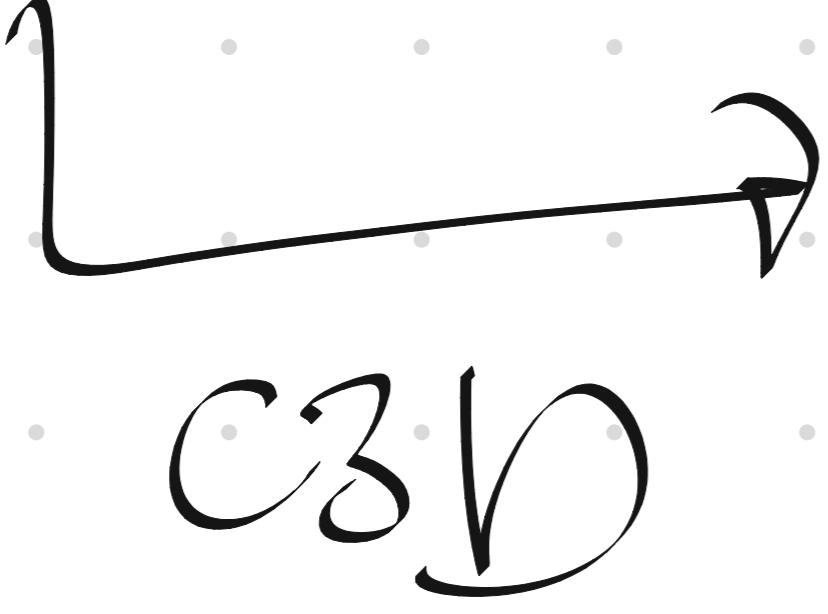
$| (E) \left\{ \begin{array}{l} F.REP = E.REP \\ F.C3D = E.C3D \end{array} \right\}$



Operaciones Relacionales

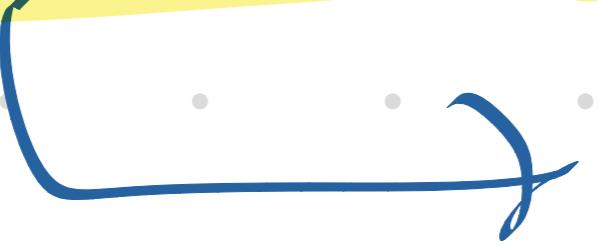
$>, >=, <, <=, !=, ==$ \leftarrow opREL

$a > b ?$



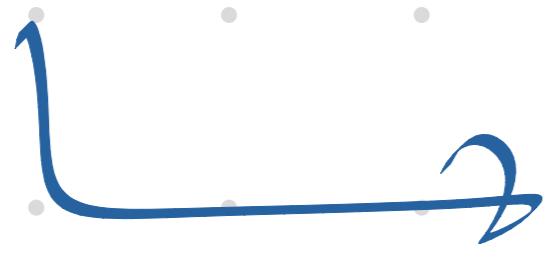
if $a > b$ goto L1
goto L2

$a == b$



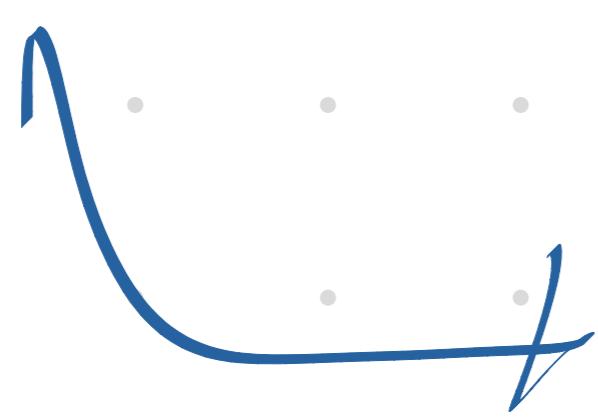
if $a == b$ goto L1
goto L2

$a <= b$



if $a <= b$ goto L1
goto L2

$a != b$



if $a != b$ goto L1
goto L2

$$(a+b)*c > x-y*z$$

$$t_1 = a+b$$

$$t_2 = t_1 * c$$

$$t_3 = y * z$$

$$t_4 = x - t_3$$

if $t_2 > t_4$ go to L_1

go to L_2

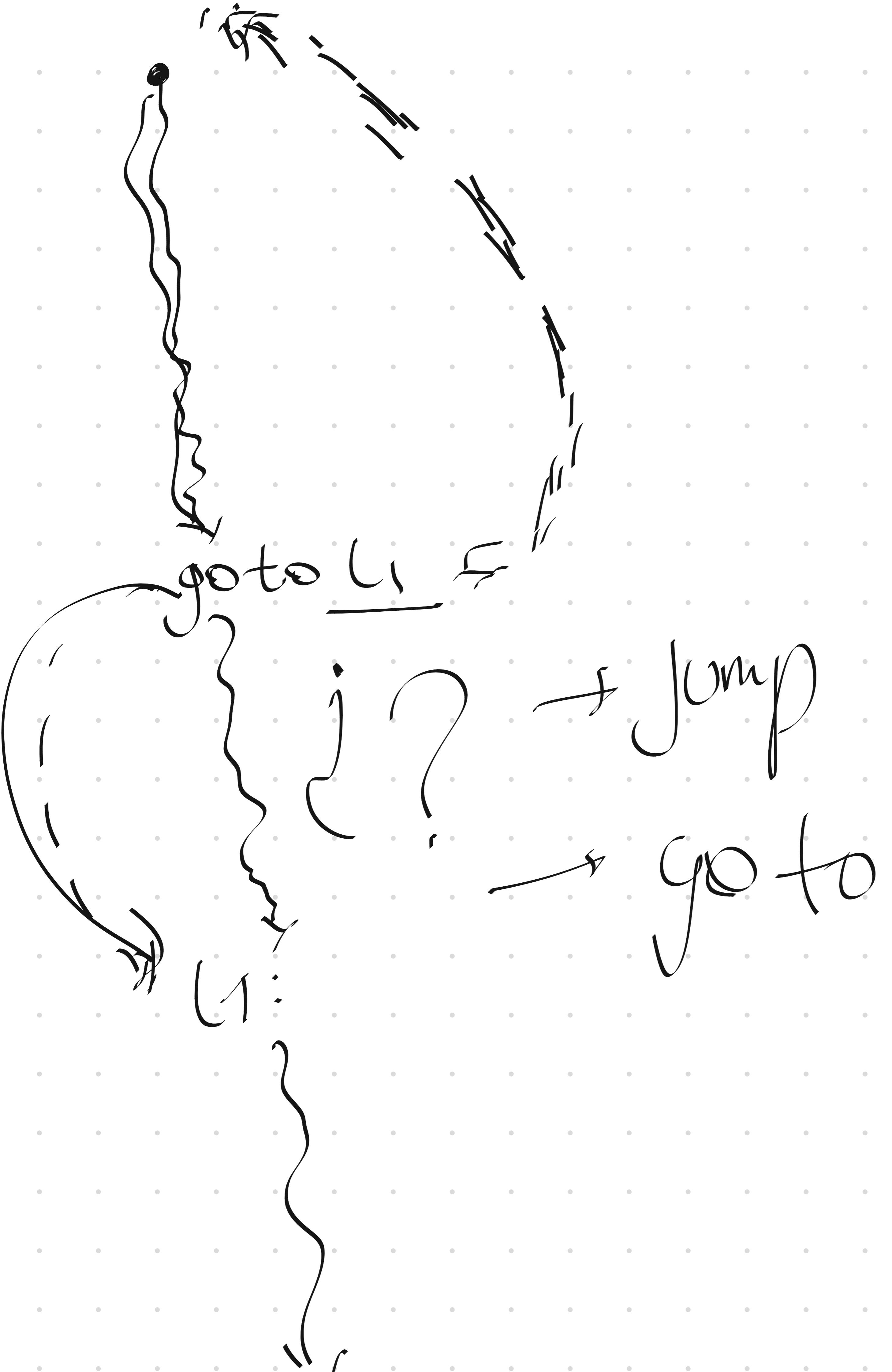
$$a > (x-y)*z$$

$$t_1 = x - y$$

$$t_2 = t_1 * z$$

if $a > t_2$ goto L_1

go to L_2



CLASE 11

04/03/2025



TRUE
FALSE

VALORES
DE VERDAD

Destino

if $a > b$ goto lv

goto LF

$$a+b > y * z$$

ENFOQUE C3D

$$t_1 = a + b$$

C3D y REP

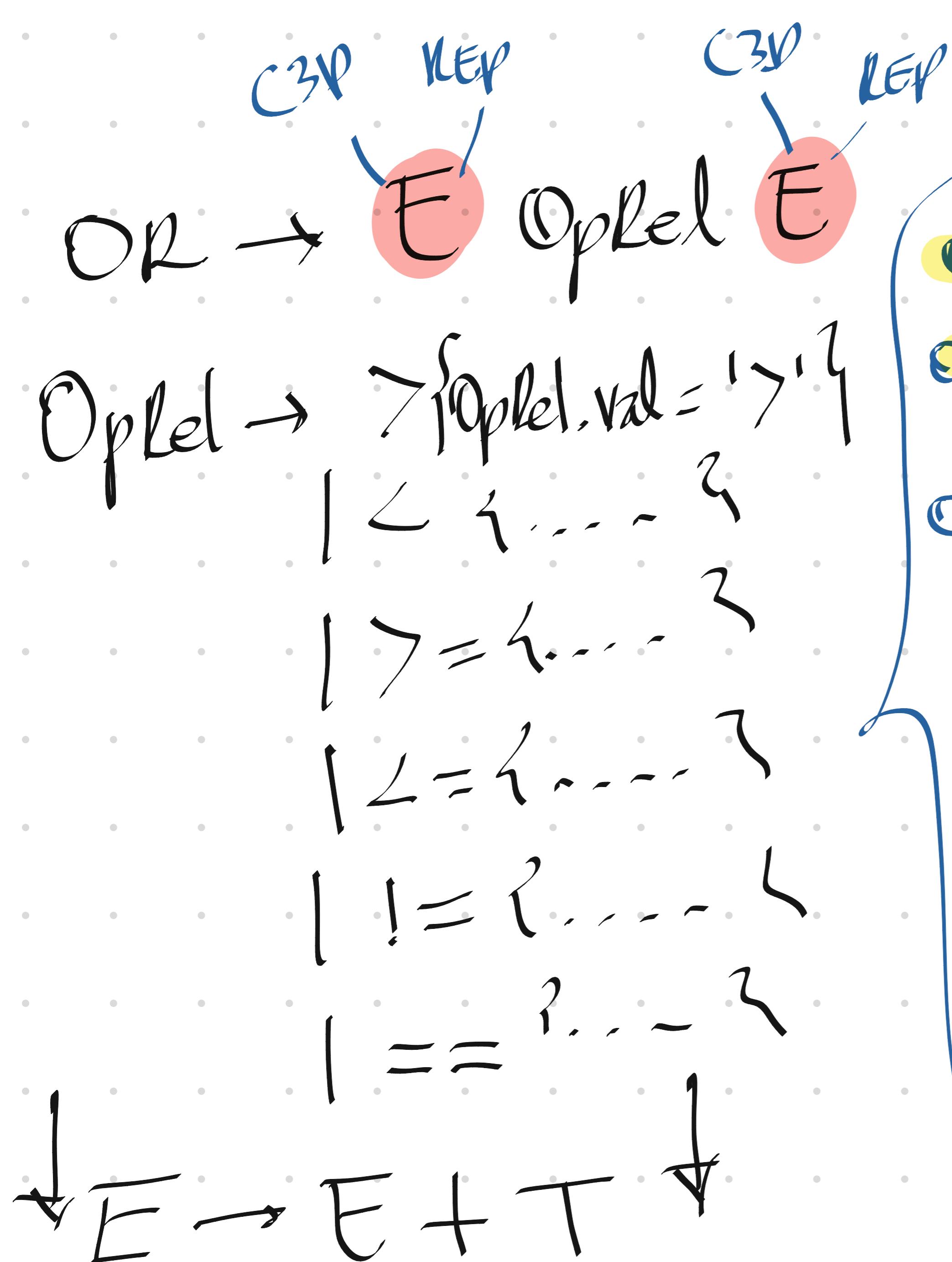
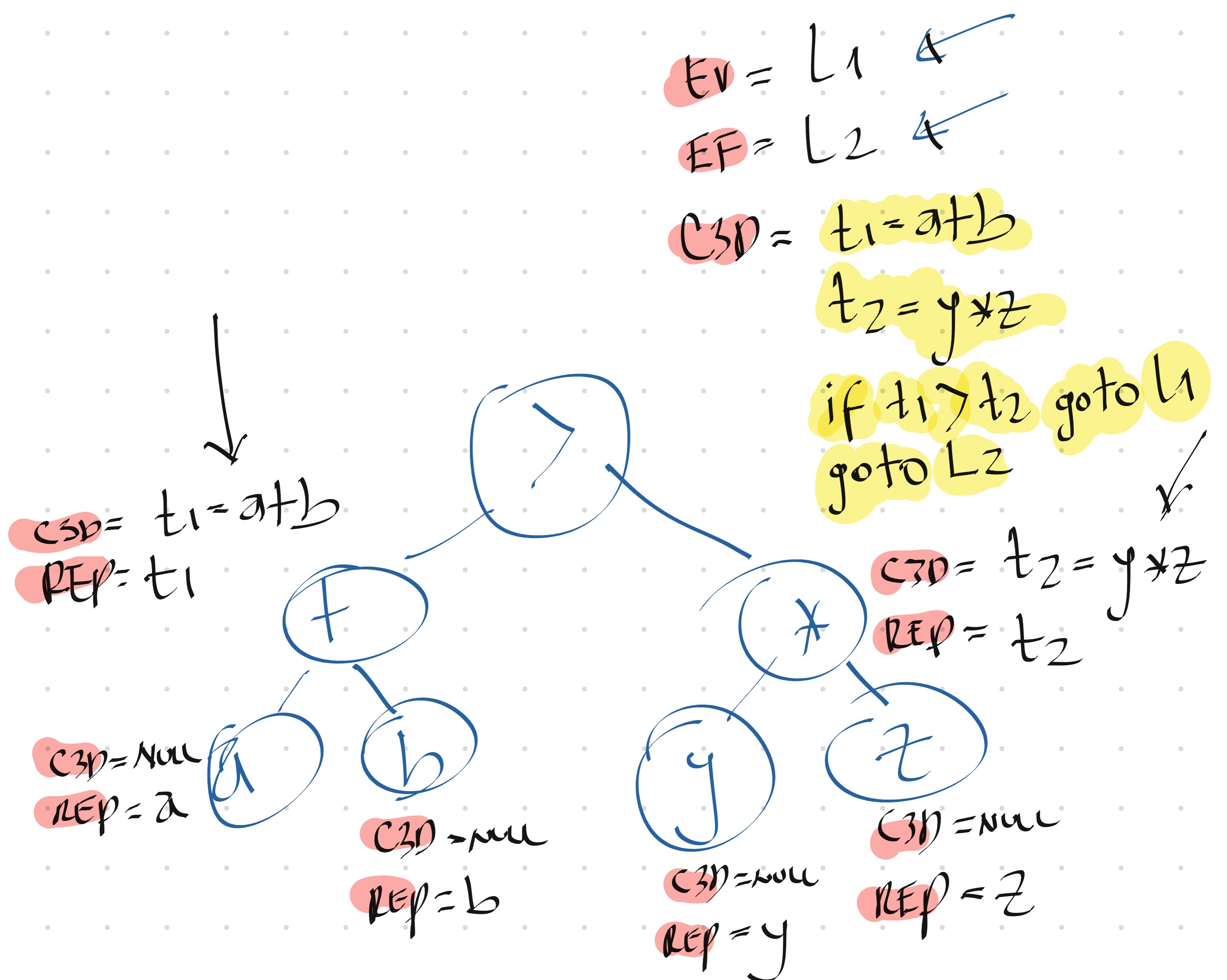
$$t_2 = y * z$$

C3D y REP

1 if $t_1 > t_2$ goto l1
go to l2

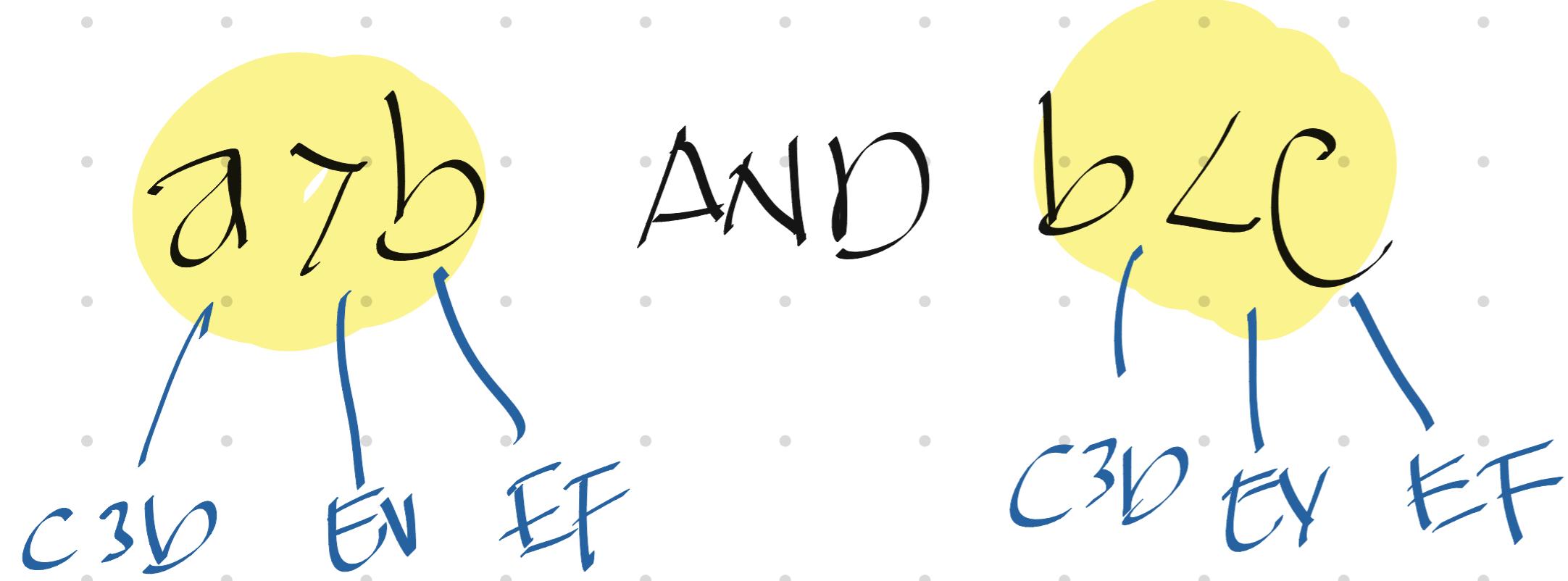
C3D
EV
EF

$L_1, L_2, L_3, \dots, L_n \rightarrow \text{getLabel}()$



Operaciones Booleanas

AND



P q	P AND q	
1 1	1	L3 *
1 0	0	L4 *
0 1	0	L2 *
0 0	0	L2 *

c1d if a > b go to L1
go to L2

c2d if b < c go to L3
go to L4

EV L1
EF L2

EV L3
EF L4

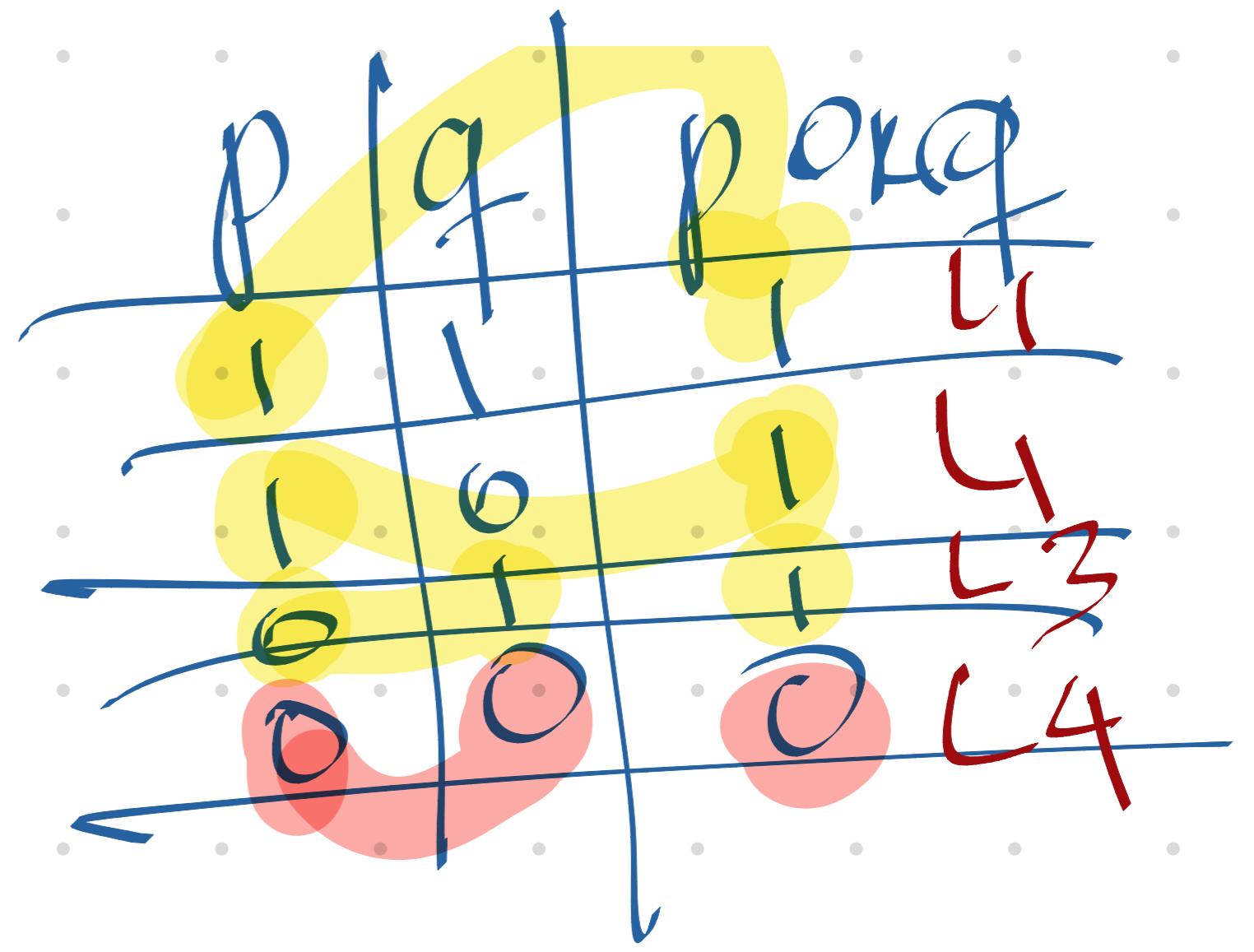
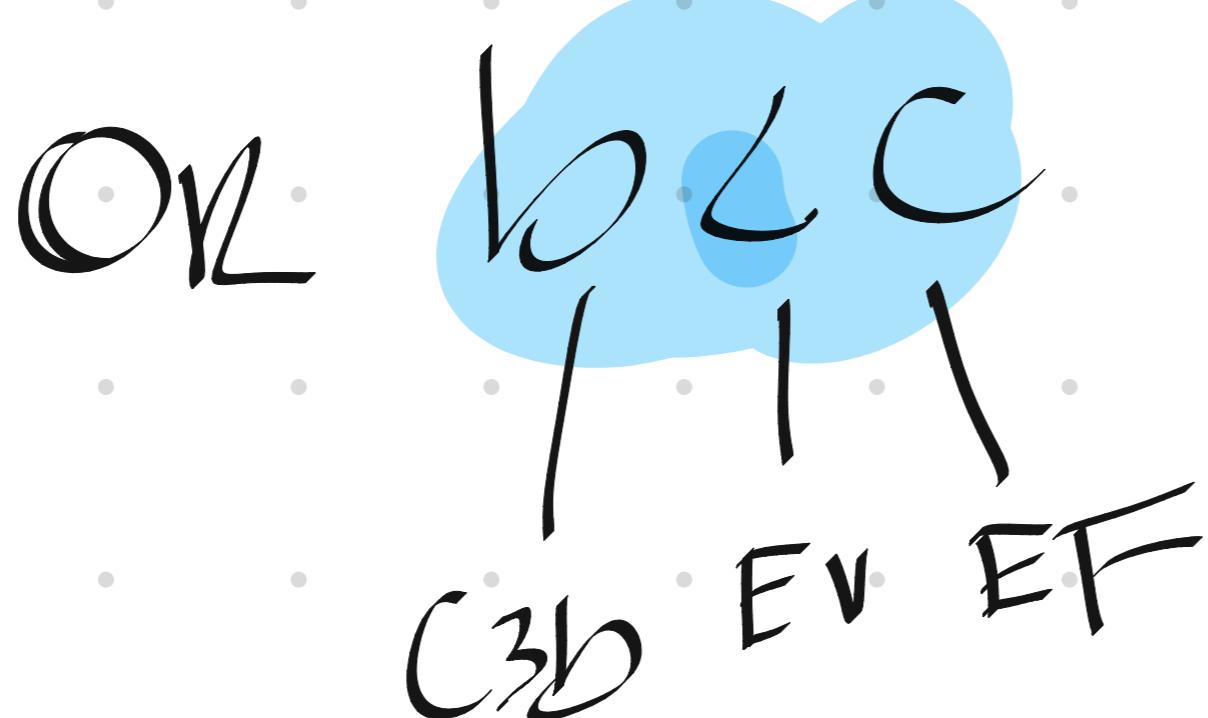
c3d if a > b go to L1
go to L2
L1:

EV = L3

if b < c go to L3
go to L4

EF = L2, L4

OR



if a**b** goto L1
go to L2

C3D
EV
EF
L1
L2

if b**c** goto L3
go to L4

C3D
EV
EF
L3
L4

if a**b** goto L1
go to L2

C3D
L2:

if b**c** goto L3
go to L4

$$EV = L_1, L_3$$

$$EF = L_4$$

$a+b > x \times y$ AND $(a+b) \times z < x$

$$\begin{cases} t_1 = a+b \\ t_2 = x \times y \end{cases}$$

if $t_1 > t_2$ goto L1
goto L2

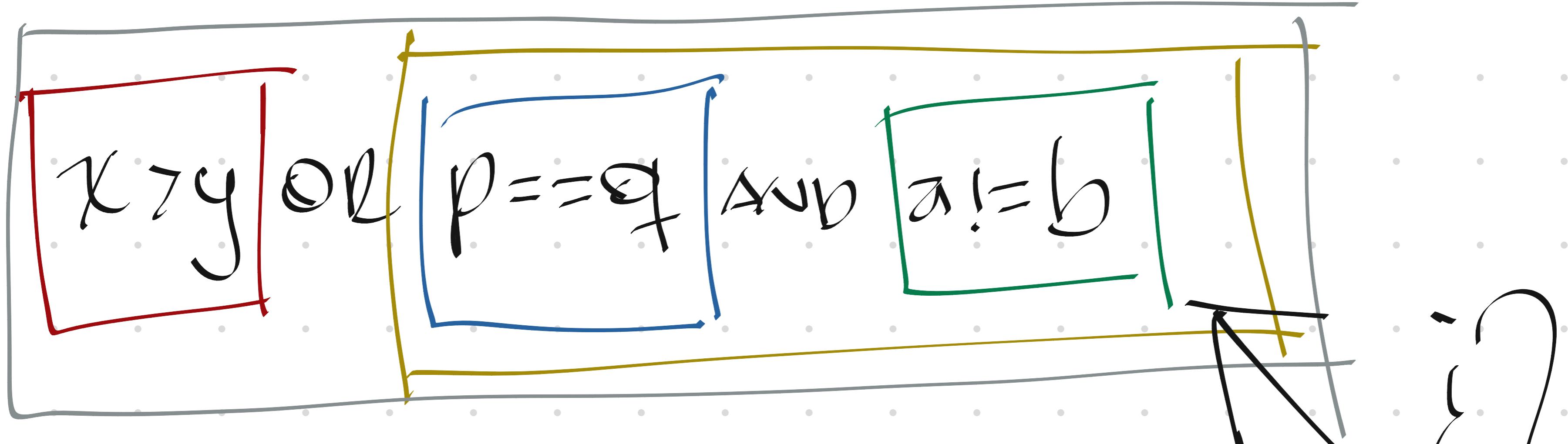
$$EV = L_3$$

$$EF = L_2, L_4$$

L1:

$$\begin{cases} t_3 = a+b \\ t_4 = t_3 \times z \end{cases}$$

if $t_4 < x$ goto L3
go to L4



if $x > y$ go to l_1

go to l_2

l_2 :

$EV = l_1, l_5$

$EF = l_4, l_6$

if $p == q$ go to l_3

go to l_4

l_3 :

if $a != b$ goto l_5

goto l_6

incorrect

NOT (a > b AND b < c)

Class 12
07/03/25

NOT ($a > b$ AND $b < c$)

if $a > b$ go to L1

go to L2

L1:

if $b < c$ go to L3

go to L4

L4:

EV =

EF =

L3

L2, L4

NOT

EV = L2, L4

EF = L3

EV = L3

EF = L2, L4

$(x > y \text{ OR } b < c) \text{ AND } (x = z \text{ OR } z \neq q)$

if $x > y$ go to L_1
go to L_2

$L_2:$

if $b < c$ goto L_3
go to L_4

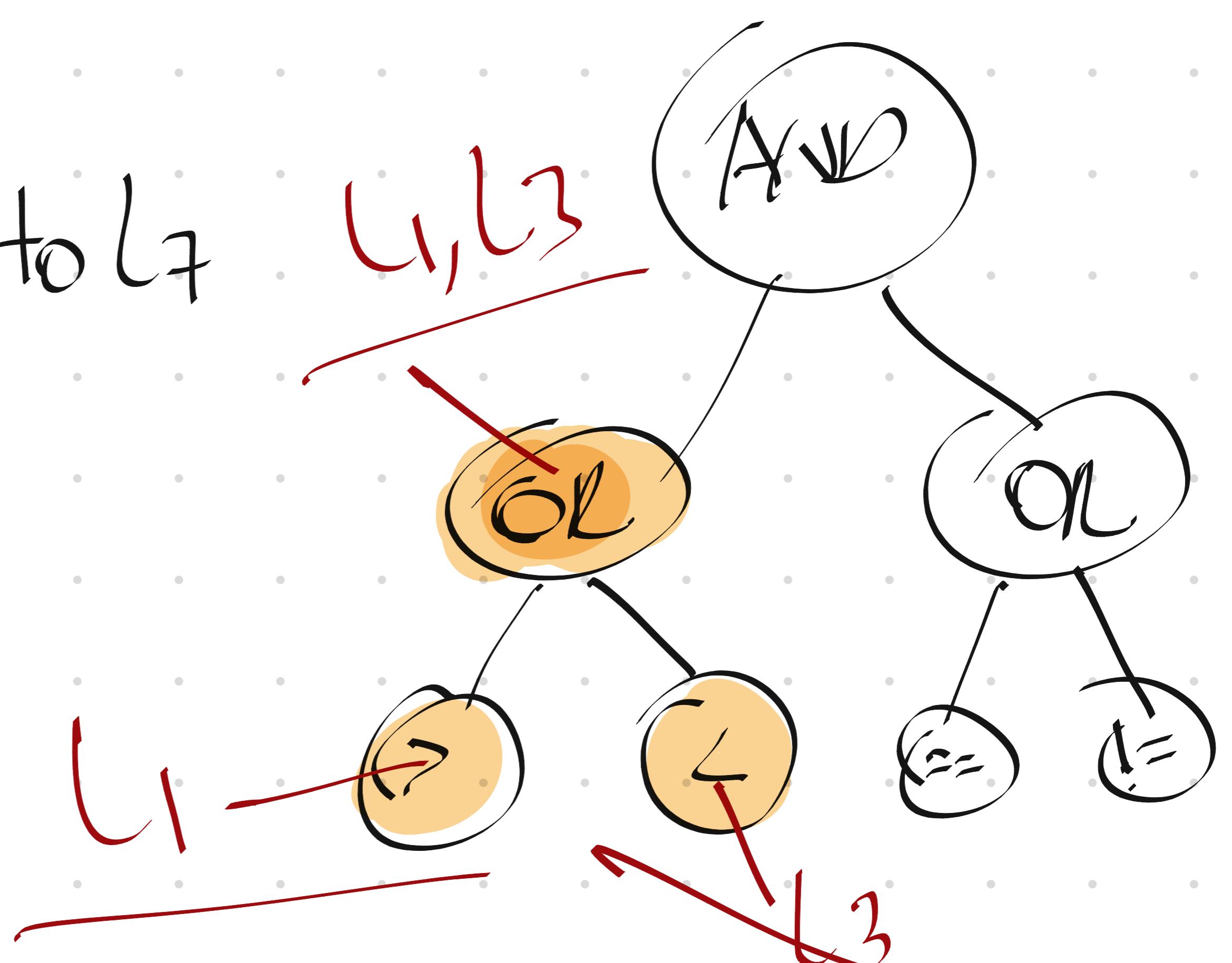
$$EV = L_5, L_7$$
$$EF = L_4, L_8$$

$L_1, L_3:$

if $x = z$ goto L_5
go to L_6

$L_6:$

if $z \neq q$ goto L_7
goto L_8



$\text{NOT}(\text{NOT}(x > b \text{ AND } \text{NOT}(c \neq d \text{ OR } g < y))) \text{ AND } u \neq t$

L3D

if $x > b$ go to L1
go to L2

L1:

if $c \neq d$ goto L3
go to L4

L4:

if $g < y$ goto L5
go to L6

L6:

if $u \neq t$ goto L7
go to L8

EV = L7

EF = L2, L3, L5, L8

↑

$$B \rightarrow B \text{ OR } P \left\{ \begin{array}{l} B.CD = B_1.CD + B_1.EF + P.CD \\ B.EV = B_1.EV + ' + P.EV \\ B.EF = P.EF \end{array} \right\}$$

$$| P \left\{ \begin{array}{l} B.CD = P.CD \\ B.EV = P.EV \\ B.EF = P.EF \end{array} \right\}$$

$$P \rightarrow P \text{ AND } Q \left\{ \begin{array}{l} P.CD = P_1.CD + P_1.EV + Q.CD \\ P.EV = Q.EV \\ P.EF = P_1.EF + ' + Q.EF \end{array} \right\}$$

$$| Q \left\{ \begin{array}{l} P.CD = Q.CD \\ P.EV = P.EV \\ P.EF = P.EF \end{array} \right\}$$

$$Q \rightarrow OR \left\{ \begin{array}{l} Q.CD = OR.CD \\ Q.EV = OR.EV \\ Q.EF = OR.EF \end{array} \right\}$$

$$| (B) \left\{ \begin{array}{l} Q.CD = B.CD \\ Q.EV = B.EV \\ Q.EF = B.EF \end{array} \right\}$$

OR \rightarrow CD
 \rightarrow EV
 \rightarrow EF

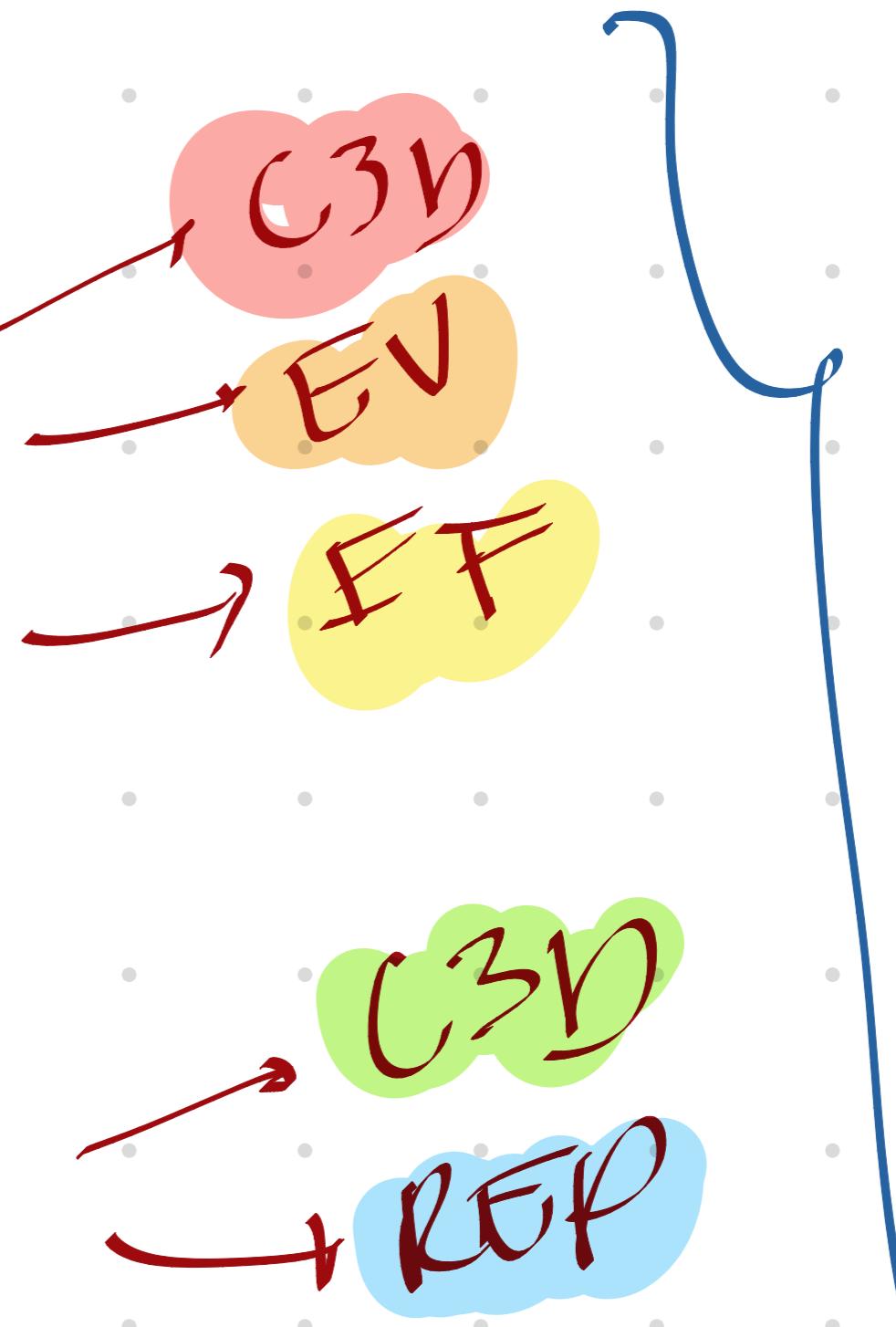
$$X \rightarrow NOT(B) \left\{ \begin{array}{l} X.CD = B.CD \\ X.EV = B.EF \\ X.EF = B.EV \end{array} \right\}$$

FLUJOS DE CONTROL

if

if $a > b$ {
 $t = a + b$

→ Condición



Sentencias

if $a > b$ goto l1
goto l2

l1:
 $t = a + b$
 $a = t$

l2:

if $x > b$ then
 $x = x + 1$
end

if $x > b$ then
 $x = x + 1$

if $x > b$ then
begin
 $x = x + 1$
end

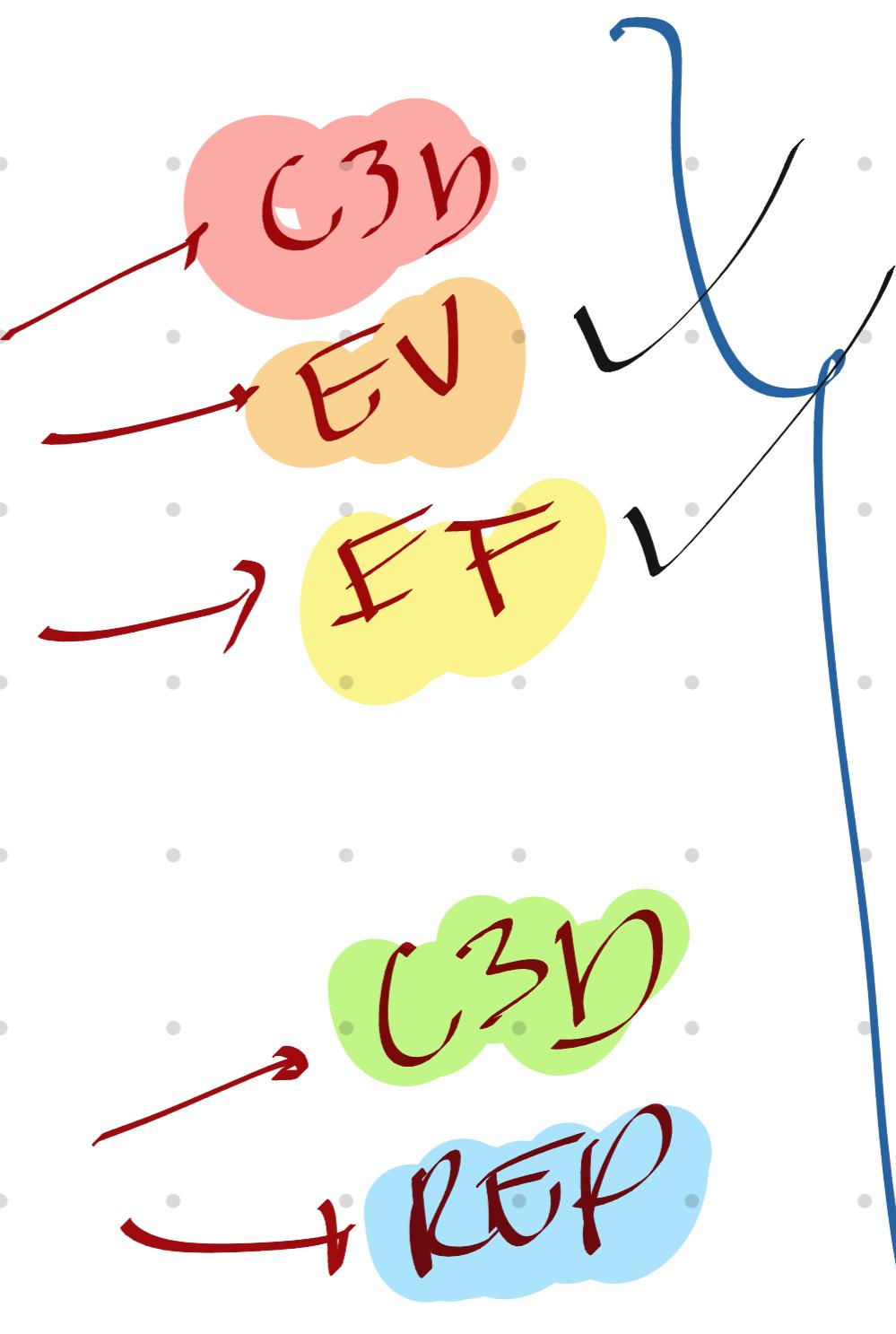
FLUJOS DE CONTROL

~~if~~

if $a > b$ {

$t = a + b$

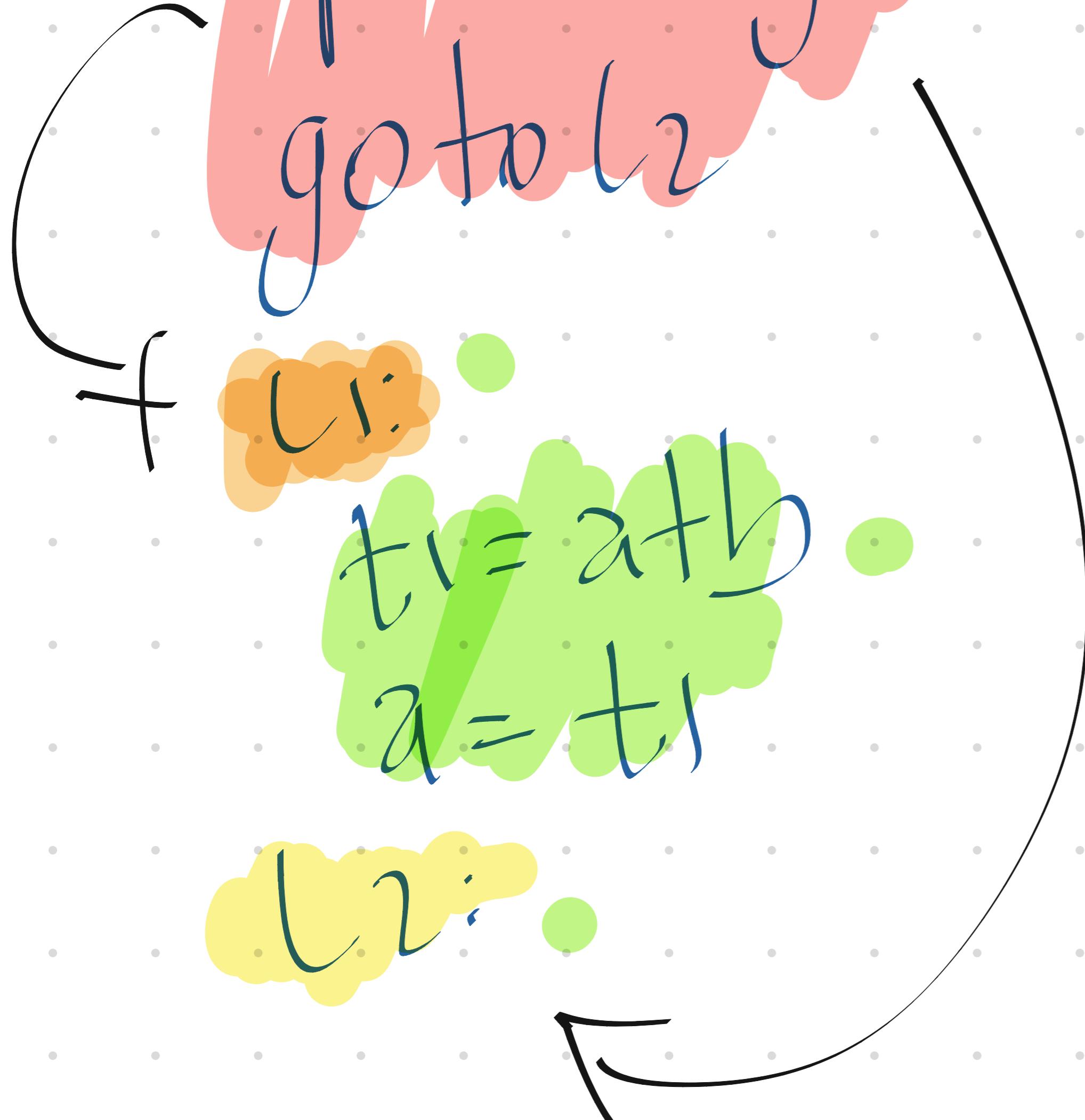
→ Condición



Sentencias

}

if $a > b$ goto l1
goto l2



if $a > b$ AND $x \neq y$ {

$x = x + 1$

}

if $a > b$ go to L1

go to L2

L1:

if $x \neq y$ go to L3

go to L4

L3:

$t_1 = x + 1$

$x = t_1$

L2, L4

if $x > y$ L1

if $p == q$ L2

if $a \leq b$ L3

$x = p + a$

L1:

L2:

if $x > y$ goto L4
goto L2

L1:

if $p == q$ goto L3

goto L4

L3:

if $a \leq b$ goto L5
goto L6

L5:

$t_1 = p + a$

$x = t_1$

L6:

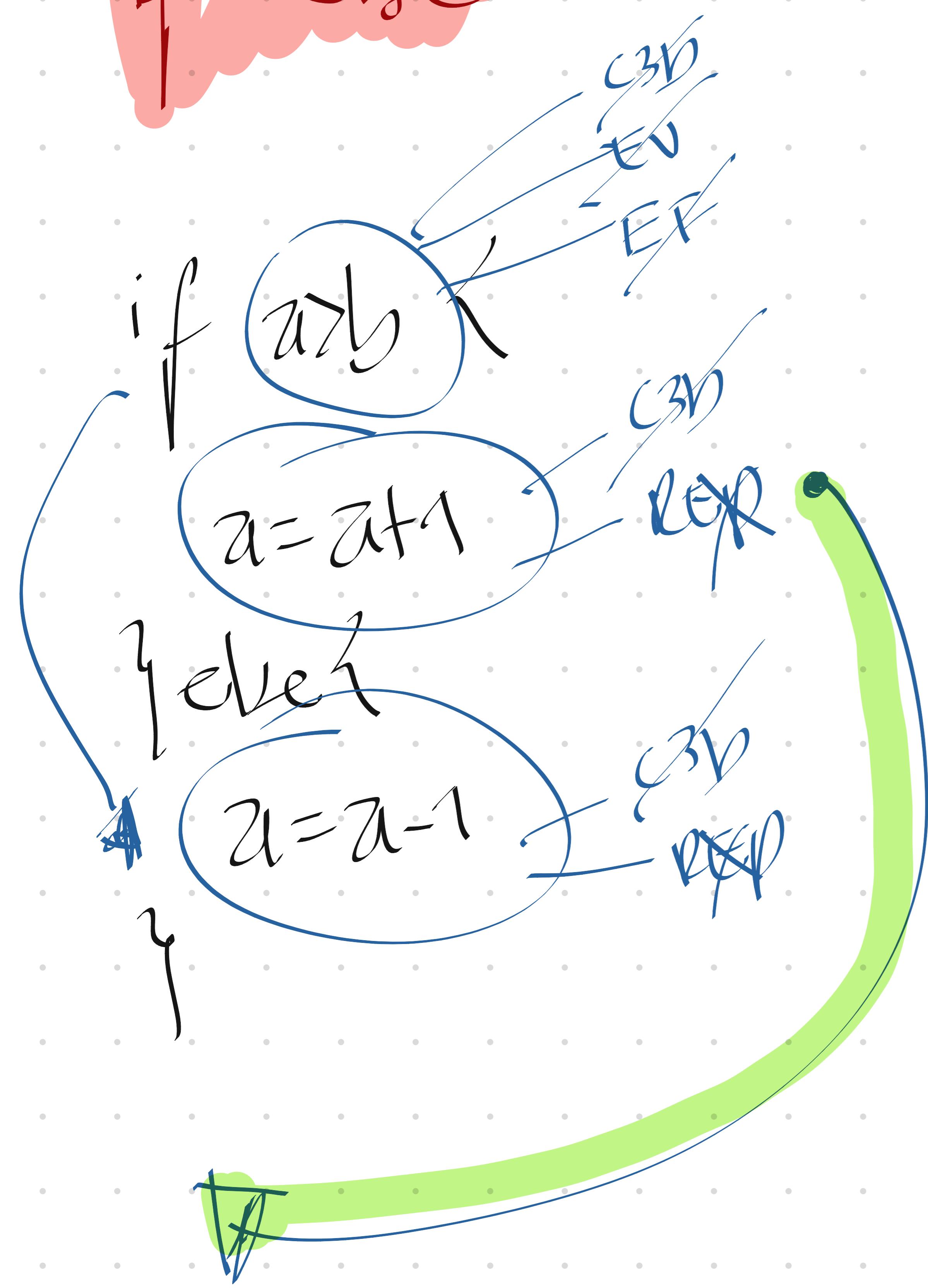
L4:

L2:

if $\sim S$

if $\rightarrow S$ if

if-else



if $a > b$ go to l1

go to l2

l1:

$t_1 = a + 1$

$a = t_1$

goto l3

l2:

$t_2 = a - 1$

$a = t_2$

l3:

if $x > y$ {

 if $x == y$ {

$x = x + 1$

 }

 else {

 if $p != q$ {

$a = p + q$

 }

}

if $x > y$ goto l1
goto l2

l1:

if $x == y$ goto l3
goto l4

l3:

$t_1 = x + 1$

$x = t_1$

l4:

goto l7

l2:

if $p != q$ goto l5
goto l6

l5:

$t_2 = p + q$

$a = t_2$

l6:

l7:

if $a > b$ {

$a = a + 1$

} else if $x > y$ {

$x = x + 1$

} else {

$a = b + 1$

}

if $a > b$ {

$a = a + 1$

} else {

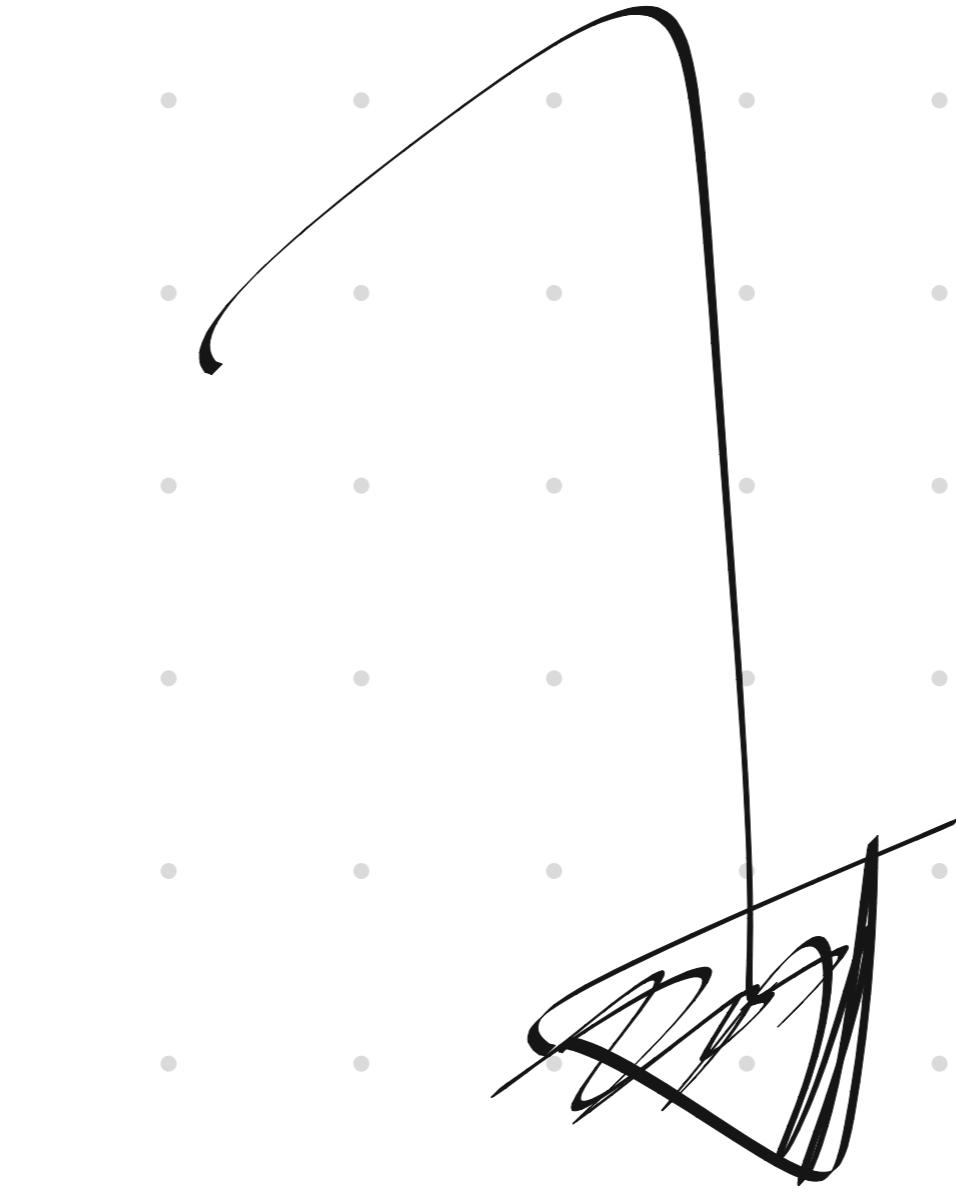
if $x > y$ {

$x = x + 1$

} else {

$a = b + 1$

}



CSN

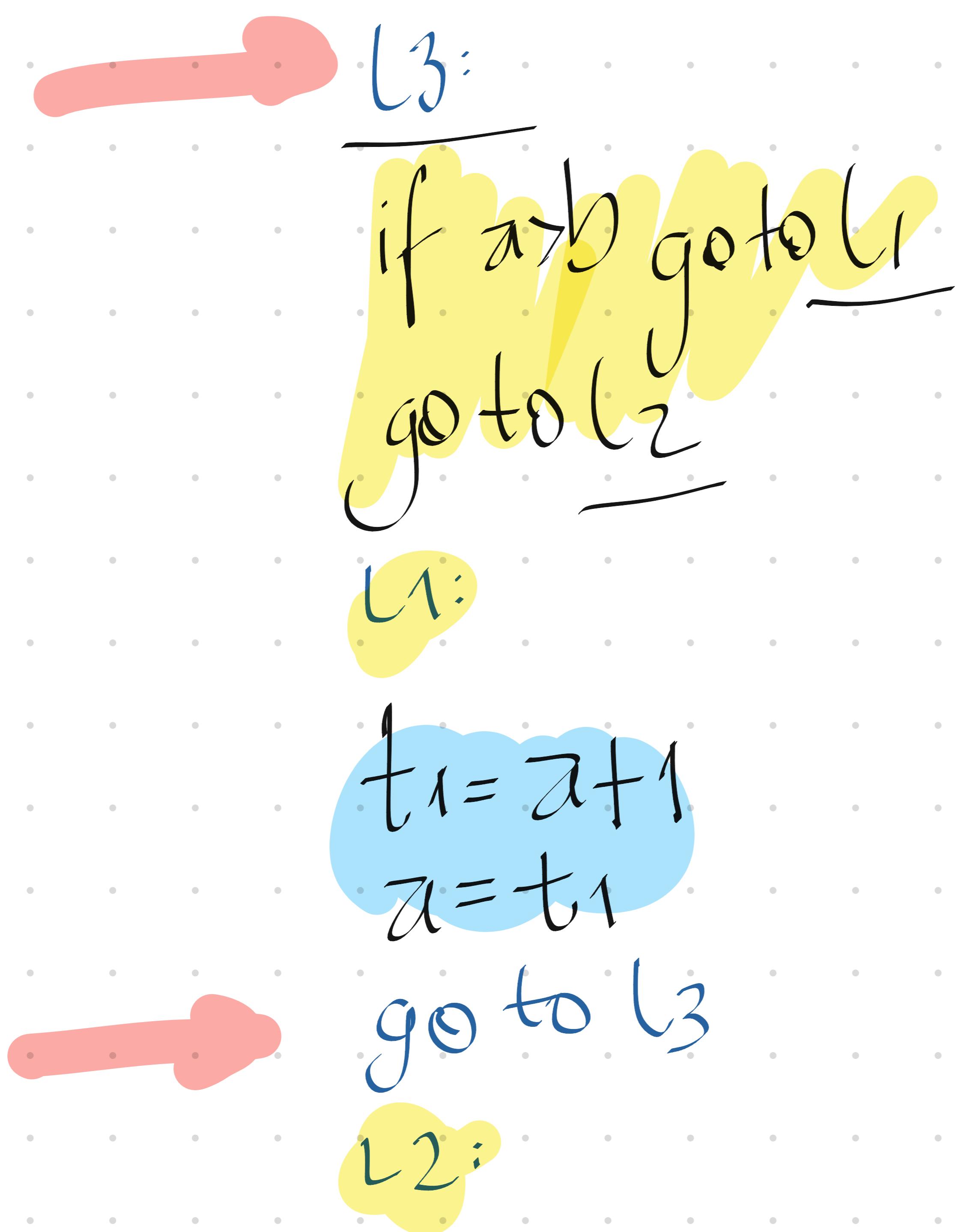
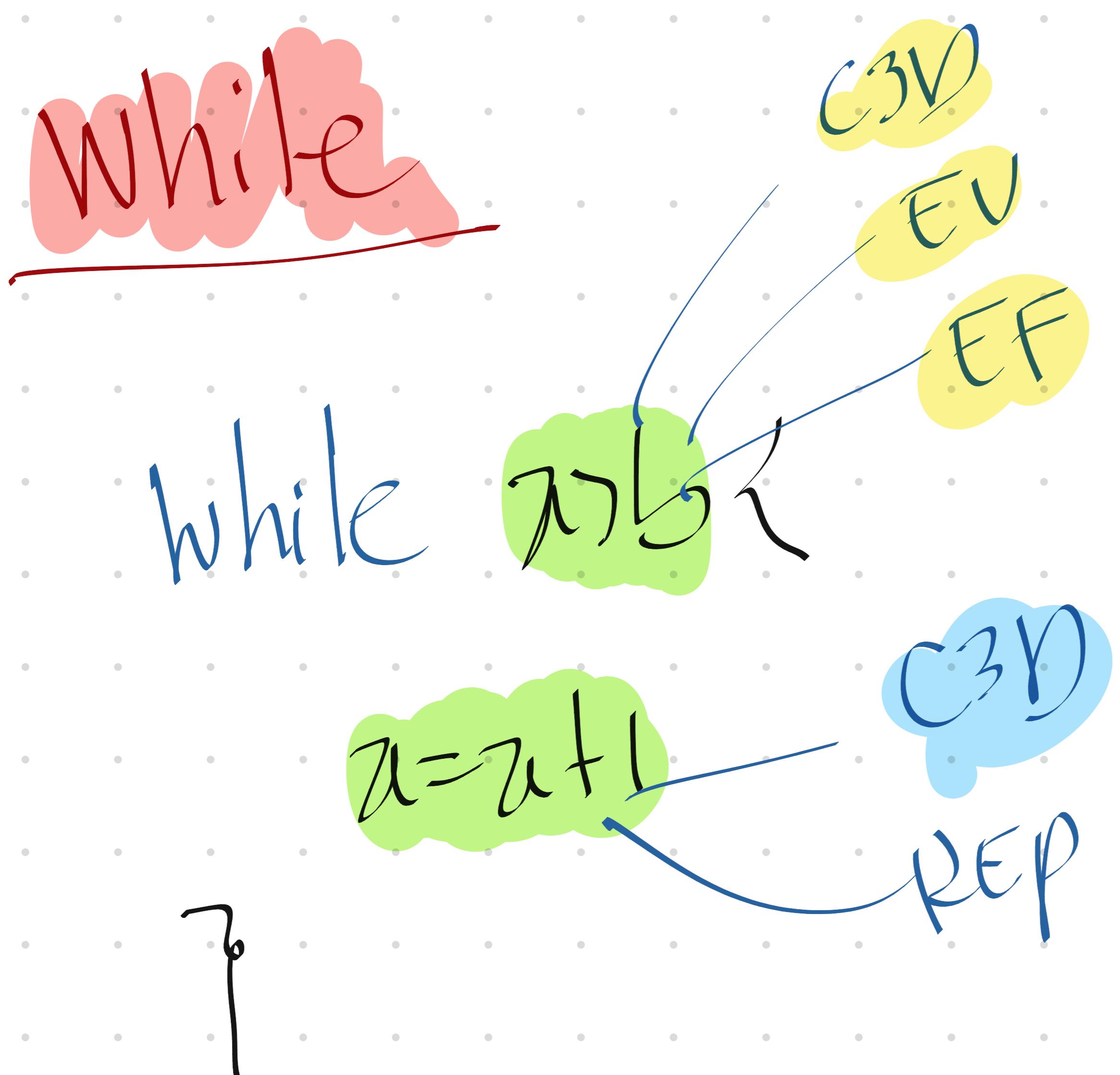
White

while $\pi \neq \emptyset$

$a = \pi[1]$

π

Clare 14
13/03/2025



while $a > b$ L

while $x == y$ L

while $p != q$ L

$a = x + p$

}

L9:

if $a > b$ goto L1
goto L2

L1:

L8:

if $x == y$ goto L3
goto L4

L3:

L7:

if $p != q$ goto L5
goto L6

L5:

$t_1 = x + p$
 $a = t_1$
goto L7

L6:

goto L8

L4:

goto L9

L2:

Do - While

Do {

$x = x + 1$

} while

$x \leq 10$

C3D

REP

C3D

EV

EF

L1:

$t1 = x + 1$

$x = t1$

if $x \leq 10$ go to L1
go to L2

L2:

switch / Select Case

switch x {

if $x == 1$ {

case 1 {

$x = x + 1$

}

case y {

$x = y + 1$

}

case z {

$x = z + 1$

}

default {

$x = x - 1$

}

}

if $x == 1$ {

$x = x + 1$

}

if $x == y$ {

$x = y + 1$

if $x == z$ {

$x = z + 1$

}

$x = x - 1$

Class 15
20/03/2015

if $a > b \text{ And } p \neq q$ {

 while $a < p$ {

$a = a + 2$

 }

$p = p + a$

 }

 if $x > q$ {

$x = x + y$

 }

}

if $a > b$ goto L1

go to L2

L1:

L5:

if $a < p$ goto L3

go to L4

L3:

$t_1 = a + 2$

$u = t_1$

goto L5

L4:

$t_2 = p + n$

$p = t_2$

go to L8

L2:

if $x > q$ goto L6

go to L7

L6:

$t_3 = x + y$

$x = t_3$

L7:

L8:

AND
if $x \neq b$ goto L1
go to L2

L1:
if $p \neq q$ go to L3
go to L4

{J:

L7:
if $x \neq p$ go to L5
go to L6

L5:

$t_1 = a + z$
 $a = t_1$

go to L7

L6:

$t_2 = p + z$
 $p = t_2$

go to L10

$$EW = \cancel{L_3}$$
$$EF = \cancel{L_2, L_4}$$



if $x \neq q$ go to L8
go to L9

L8:

$t_3 = x + y$

$x = t_3$

L9:

L10:

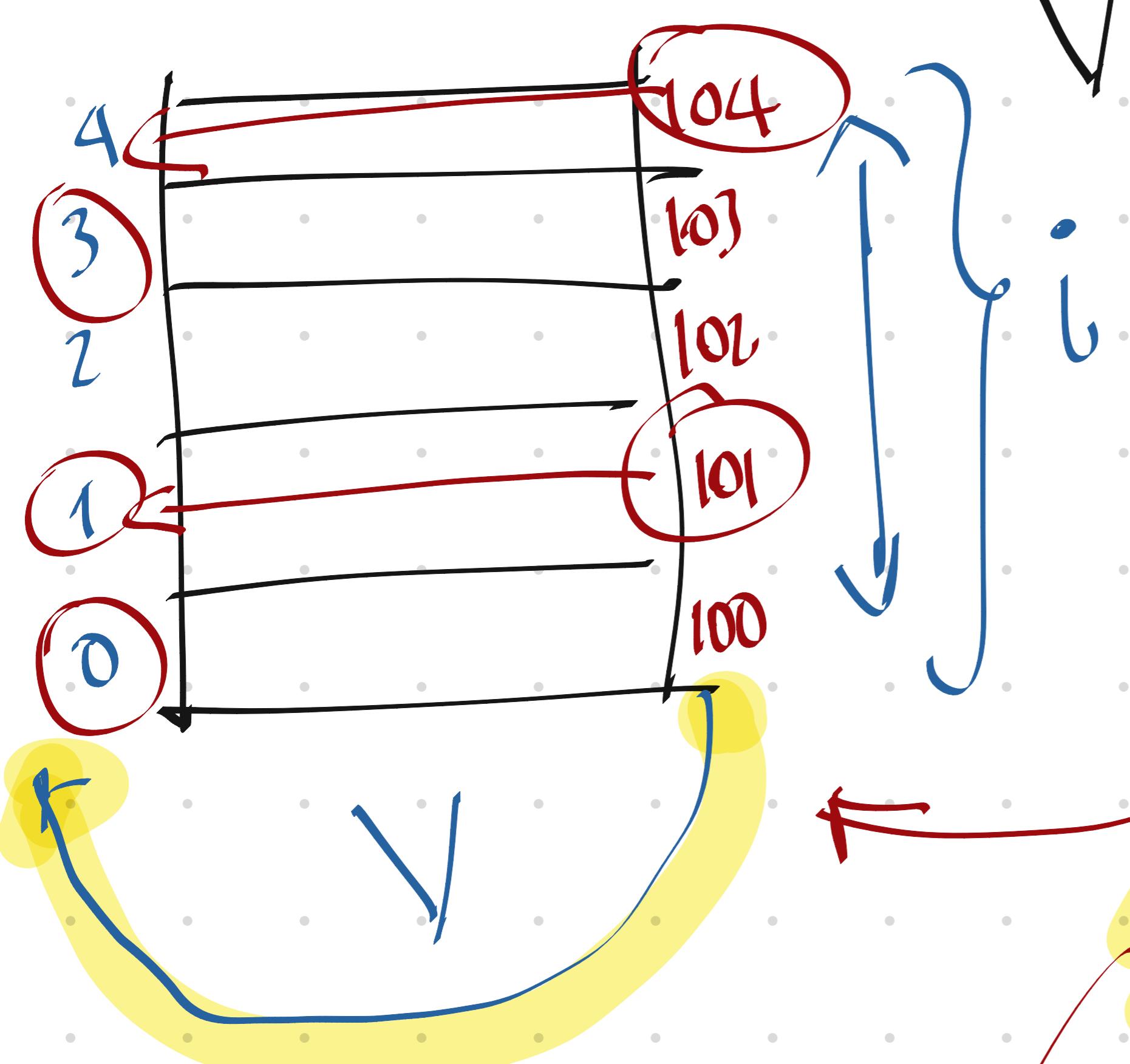
ARREGLOS (ARRAY)

C3D \rightarrow $x = y[i]$
 $x[i] = y$

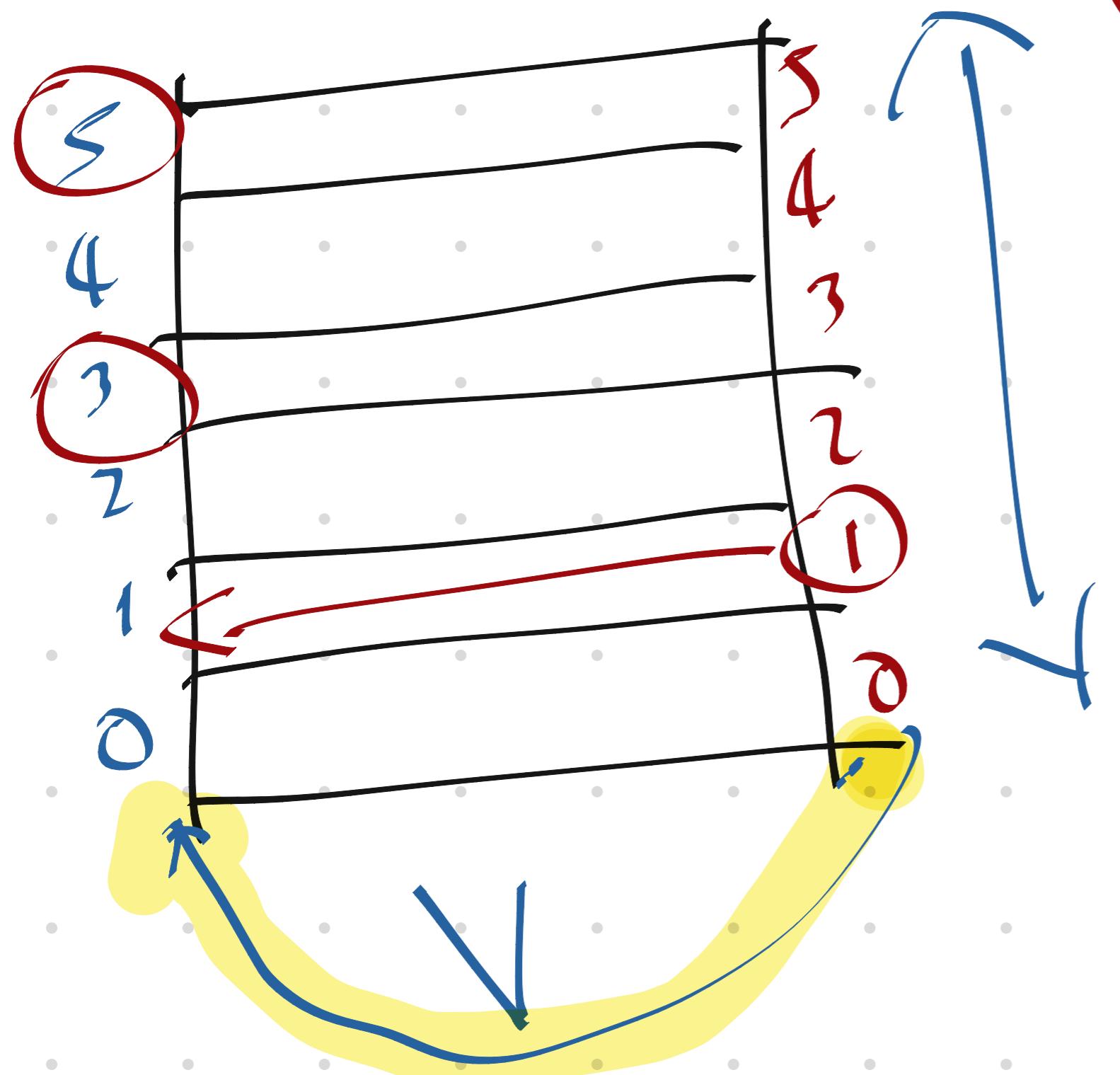
~~$x[i] = y[j]$~~
 ~~$x[i, j] = y$~~

ARREGLOS 1 DIMENSIÓN

$V : \text{ARRAY}[100..104] \text{ of int}$



$i_{C3D}(i) = i - \inf$

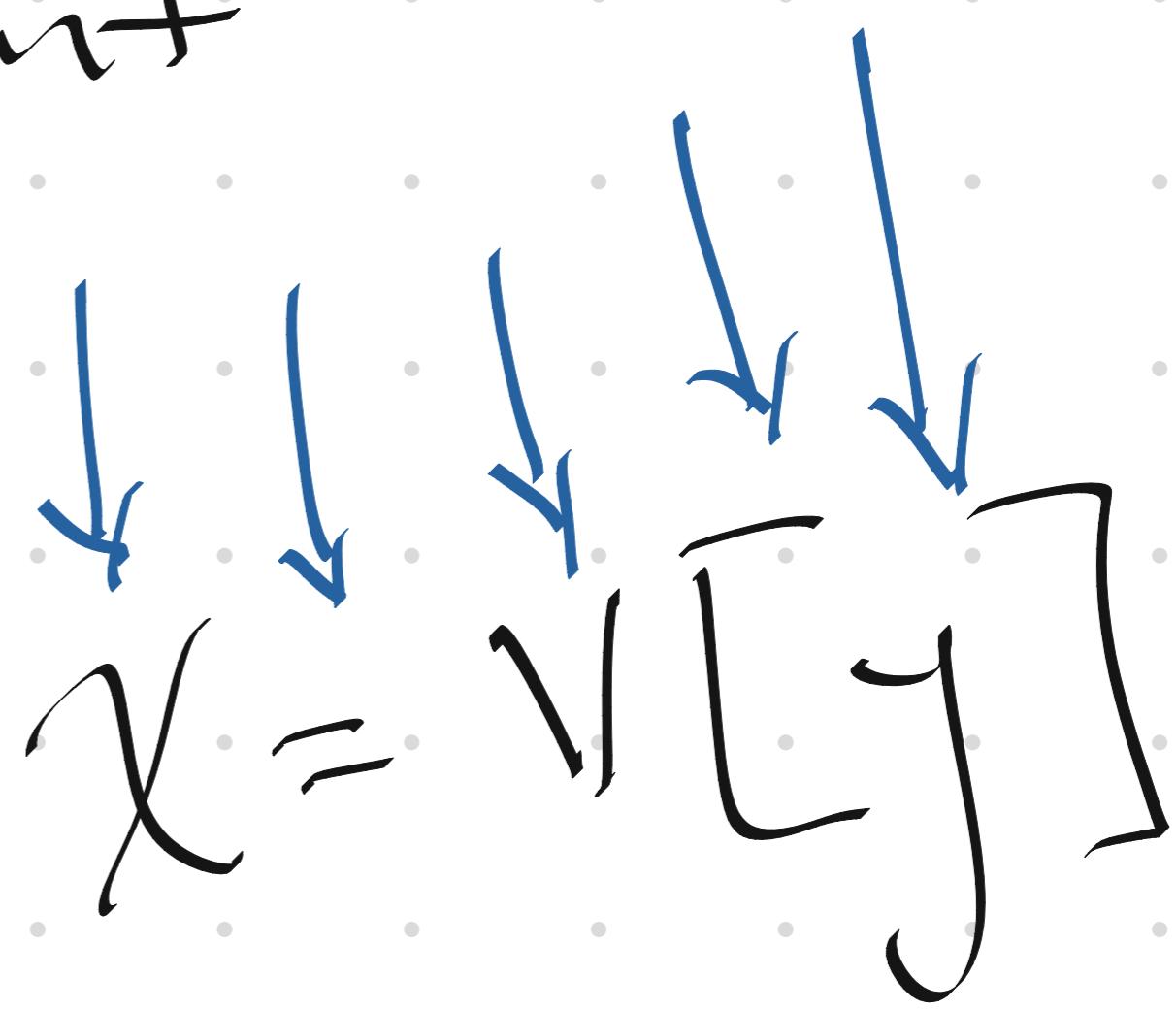
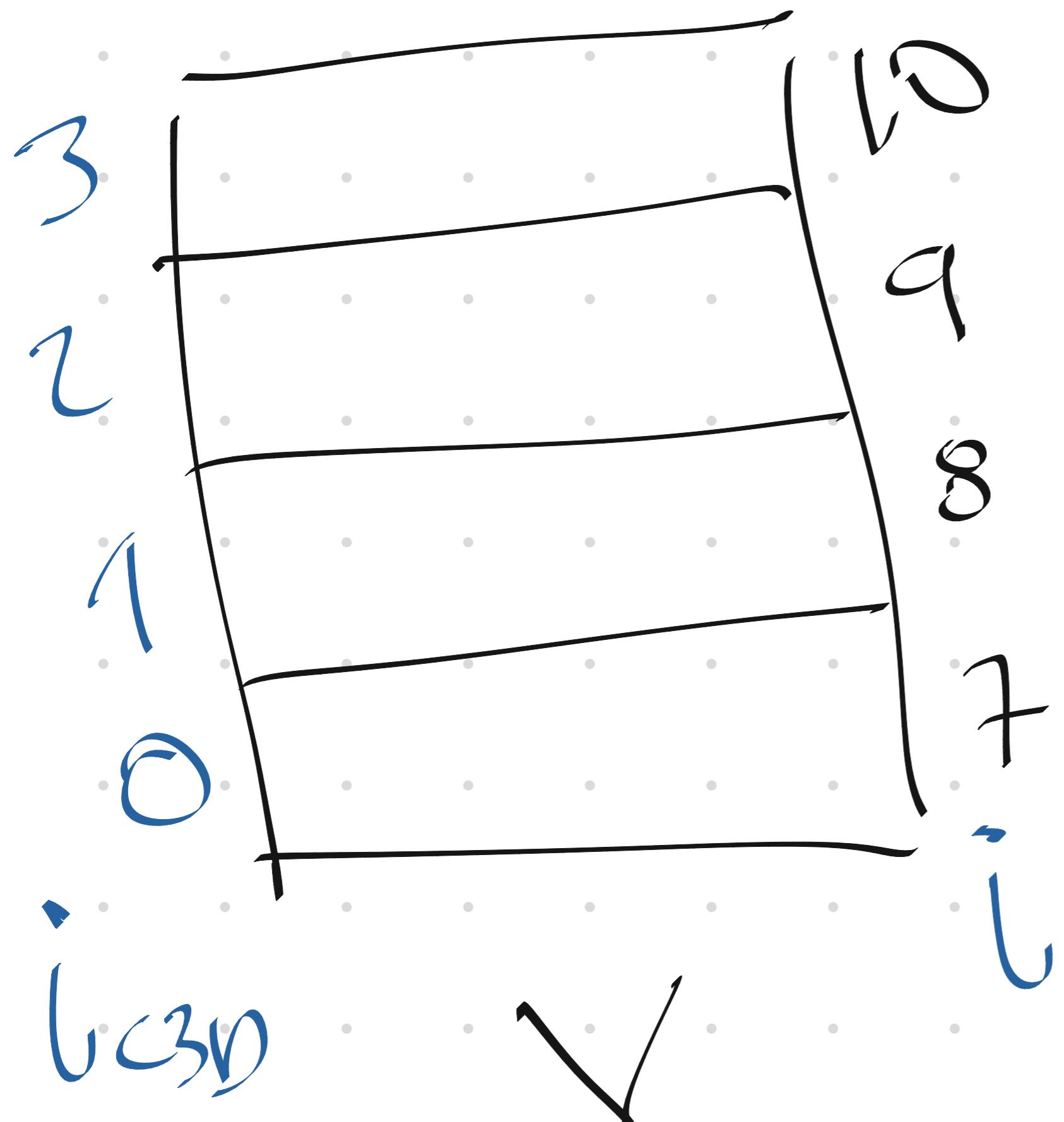


$V : \text{ARRAY}[0..5] \text{ of int}$

inf sup

ACCESO

$y : \text{ARRAY}[7..10] \text{ of int}$



$$t_1 = y - 7$$

$$t_2 = \sqrt{t_1}$$

$$x = t_2$$

$$x = \sqrt{y} + \sqrt{z}$$

$$t_1 = y - 7$$

$$t_2 = \sqrt{t_1}$$

$$t_3 = z - 7$$

$$t_4 = \sqrt{t_3}$$

$$t_5 = t_2 + t_4$$

$$x = t_5$$

$$x = \sqrt{y+z} + \sqrt{\sqrt{a}}$$

$$t_1 = y + z$$

$$t_2 = t_1 - 7$$

$$t_3 = \sqrt{t_2}$$

$$t_8 = t_3 + t_7$$

$$x = t_8$$

$$t_4 = a - 7$$

$$t_5 = \sqrt{t_4}$$

$$t_6 = t_5 - 7$$

$$t_7 = \sqrt{t_6}$$

ASIGNACIÓN

$y : \text{ARRAY}[11..15] \text{ of int}$

$$\downarrow \quad \downarrow \quad \downarrow$$
$$v[x] = x + y$$

$$v[v[x+y]] = 1$$

$$t_1 = x - 11$$

$$t_2 = x + y$$

$$v[t_1] = t_2$$

$$t_1 = x + y$$

$$t_2 = t_1 - 11$$

$$t_3 = v[t_2]$$

$$t_4 = t_3 - 11$$

$$v[t_4] = 1$$

$$v[v[x] + v[y]] = v[v[x * y]]$$

$$t_1 = x - 11$$

$$t_2 = v[t_1]$$

$$t_5 = t_2 + t_4$$

$$t_6 = t_5 - 11$$

$$t_8 = t_7 - 11$$

$$t_9 = v[t_8]$$

$$v[t_6] = t_{11}$$

$$t_3 = y - 11$$

$$t_4 = v[t_3]$$

$$t_7 = x * y$$

$$t_{10} = t_9 - 11$$

$$t_{11} = v[t_{10}]$$

Talib
de Simbolos

Tema 1

Escriba el código de tres direcciones de las siguientes expresiones relacionales, indique su forma estándar en c3d y su código correspondiente:

1. B in [A,D,C]
2. B in [1..10]
3. B in [A, D, 1..10, C]
4. B > C or H in [K+D*F, P, L*D] and J < L

Tema 2

Escriba la apariencia del código de tres direcciones para la sentencia cíclica *switch*, donde existe una variable selectora *Var* y valores constantes *val1*, *val2*,.. *valn*, que la variable podría recibir. Después de ejecutar las sentencias asociadas al valor de coincidencia de la variable se debe regresar a verificar si el valor de la variable fue modificado y ejecutar las sentencias del nuevo valor de coincidencia. La única forma de salir del ciclo es con la sentencia *Break*. La sintaxis de la sentencia Switch se muestra a continuación:

```
Switch {  
Case val1: Sentencias 1  
Case val2: Sentencias 2  
:  
:  
Case valn: Sentencias n  
}(Var)
```

Tema 3

Para la implementación de un ciclo while, con el formato de entrada while(Cond){S}, de tal manera que las sentencias S se ejecuten mientras la condición Cond se cumple. Considere que dentro de las sentencias S, se permiten:

Break: instrucción que envía hacia afuera del ciclo

Continue: Instrucción que regresa a evaluar la condición

- a. Muestre la apariencia del código de tres direcciones la siguiente entrada:

```
while(Cond1)
{
    Continue; Break;
    while(Cond2){ Break; Break; Break; Break; Continue;} Break;
}
```

Escriba un ejemplo del uso del while y las sentencias break y continue

Tema 4

El manejo de valores TRUE y FALSE en código tres direcciones no está definido de manera explícita, por lo cual se solicita:

- a) Escribir una propuesta de traducción a código tres direcciones de los valores TRUE y FALSE
- b) Con base a la propuesta traduzca las siguientes operaciones booleanas e indique los valores finales de las etiquetas en cada caso.
 - a. TRUE AND TRUE
 - b. TRUE AND FALSE
 - c. FALSE OR FALSE
 - d. FALSE OR TRUE

Tema 5

El código de tres direcciones en corto circuito para condiciones trata de disminuir al mínimo el número de saltos al momento de la ejecución para conocer el resultado de la expresión condicional, en tal sentido puede resultar de mucho beneficio sustituir la condición por su complemento. Escriba la definición dirigida por la sintaxis para sustituir la condición por su complemento al momento de general el código de tres direcciones, de acuerdo con el ejemplo siguiente:

if a < b then goto L1

goto L2

por:

if a>= b then goto L2

goto L1

Tema 6

Considere que existe la sentencia de alto nivel: Loop {Sentencias} que sirve para hacer un ciclo que se repite hasta que la sentencia de escape Break sea alcanzada. Ambas sentencias deben estar relacionadas ya que se debe asegurar que la secuencia de escape siempre aparezca, para ello se debe asegurar que cada ocurrencia de la sentencia Break se encuentre asociada a la sentencia If. Escriba la propuesta de código estándar de tres direcciones para estas sentencias (Loop y Break) y escriba el esquema de traducción que genere el código y además detecte todos los posibles errores de semántica.