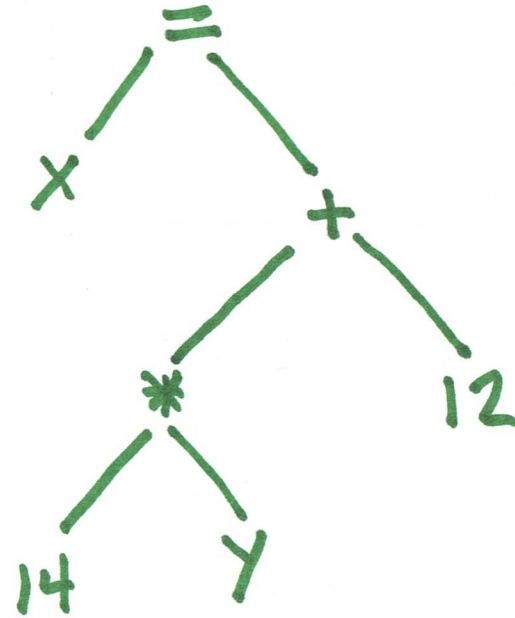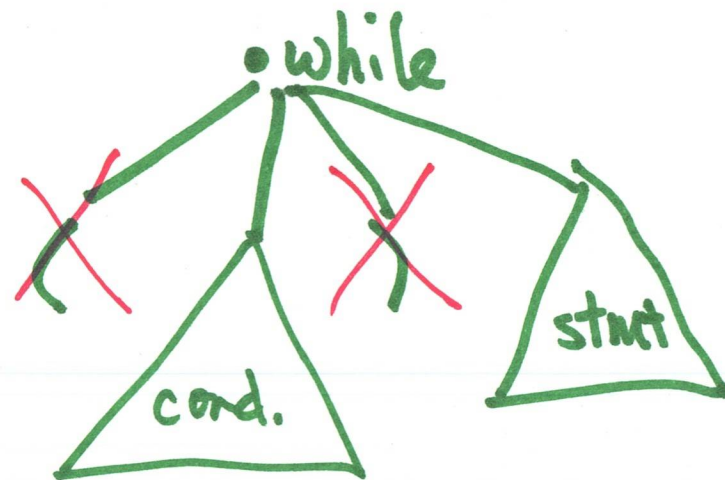$x = 14 * y + 12$
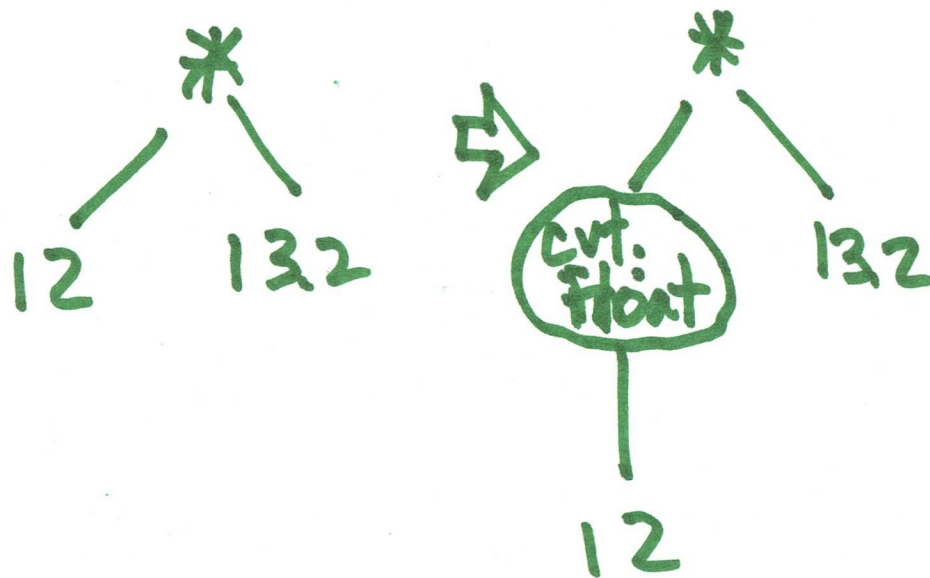

while (cond) stmt

parse
tree
AST

$12 * 13.2$


```
LOADI   12, r1
CVTF    r1
LOADI   13.2, r2
~~ADD~~   r1, r2, r3
MUL
```

$x = \underset{\text{bool}}{\text{true}} + \underset{\text{float}}{31.2}$

$y = \text{true}$

$z = 31.2$

$\boxed{x = y + z}$

```
switch (c) {
  case 'a':

  case 'b':

  case 'a': ✗
}
```

regular expressions

alphabet Σ = set.

elements of an alphabet are <u>characters</u> or <u>symbols</u>
or <u>letters</u>

$$\Sigma = \{0,1\} \qquad \Sigma' = \{1\}$$

string sequence of symbols (with possible repeats) from an
alphabet

010010 string on $\Sigma = \{0,1\}$

111111 string on $\Sigma = \{1\}$ or $\Sigma = \{0,1\}$

$$\Sigma^* = \text{set of all strings on } \Sigma$$

ε (epsilon) = string of 0 characters

$$\varepsilon \in \Sigma^* \text{ for any } \Sigma$$

__language__ is a set of strings on Σ. $L \subseteq \Sigma^*$

$$L = \{01, 001, 0001, \ldots\} \qquad \text{on } \Sigma = \{0, 1\}$$

$$L = \{a, b, ab\} \qquad \text{on } \Sigma = \{a, b, c\}$$

$$L_0 = \{0\} \quad L_1 = \{1\}$$

$$L_a = \{a\} \qquad \text{for any } a \in \Sigma$$

$$L_\varepsilon = \{\varepsilon\}$$

$\overset{L \cup M}{L_a \cup L_b} = \{a, b\} = \{a\} \cup \{b\}$

$\underset{L \cdot M}{L_a L_b} = \{\alpha \beta \mid \alpha \in \underset{L}{L_a}, b \in \underset{M}{L_b}\}$

concatenation

$$L = \{a, ab\}$$
$$M = \{c, cd\}$$
$$LM = \{ac, acd, abc, abcd\}$$

$$L^k = \underbrace{LLL \cdots L}_{k \text{ times}}$$

$$L^1 \quad L_1 = L$$
$$L^0 \quad L_0 = \{\varepsilon\}$$
$$L^2 \quad L_2 = LL$$
$$L^3 \quad L_3 = LLL$$

$$L^k_k = LL^{k-1}$$

$$L^* = L_0 \cup L_1 \cup L_2 \cup L_3 \cup \cdots$$

$$\Sigma = \{0, 1\}$$
$$L_{\{0,1\}} = \{0, 1\}$$

$$L^*_{\{0,1\}} = \{\varepsilon, 0, 1, 00, 01, 10, 11, 000, 001, 010, \cdots\}$$
any string of 1's and zeroes.

regular expression

$$\varepsilon \qquad L_\varepsilon$$
$$a \qquad L_a \qquad \forall a \in \Sigma$$

$$\alpha \mid \beta \qquad \underline{\alpha \not\beta} \qquad L_\alpha \cup L_\beta$$

$$\alpha\beta \qquad\qquad L_\alpha L_\beta$$

$$\alpha^k \qquad\qquad L_\alpha^k$$

$$\alpha^* \qquad\qquad L_\alpha^*$$

$\underbrace{}$ reg. exp.

$\underbrace{}$ languages.

( ) for grouping.

$$a(b|c)d^* = L = \{ab, ac, abd, acd, abdd, abcdd, \dots\}$$

while(tom)

to*m

$$\{tm, tom, toom, tooom, \dots\}$$

expressivity extensions

$\alpha?$ or $\alpha?$ $(\alpha|\varepsilon)$

α^+ $(\alpha\alpha^*)$

α^k $\overset{k}{L\alpha}$
_____

⟶ cannot use k twice as variable ▲

$\alpha \bowtie \beta$ $\left(\varepsilon + \alpha(\beta\alpha)^*\right)$

β-separated list of α's.

L^*L^k order of precedence

LM

$L|M$
_____

$(L|M)^*$ $\{\ell m\ell, \ell, m, \dots\}$

$\{\ell\}$ $\{m\}$ any string of ℓ's & m's.

$\alpha^k\beta^k$

$\alpha^2\beta^2$

$X\bowtie_2$ ~~(scribbled out)~~
_____

ε X X,X X X,X,X
X,X,X,X