2-1

BNF – Backus-Naur Form (variant of CFG context-free grammar)

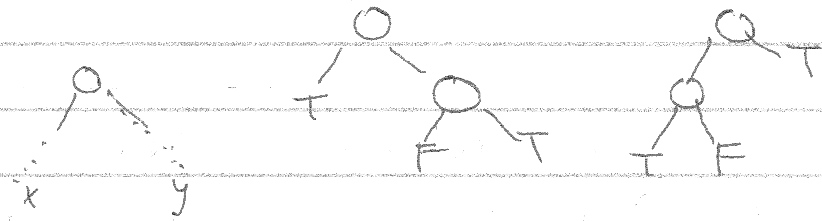
$$
\begin{aligned}
e &= T \\
&\mid F \\
&\mid e \otimes e
\end{aligned}
$$

terminals

non terminal

"$T \otimes F \otimes T$" $\in e$

$x \otimes y$



---

semantics – what P. Languages mean

compiler – translate A into B

$$
\begin{tikzcd}
A \arrow[r, "com"] \arrow[dr, "run"'] & B \arrow[d, "run"] \\
& ans
\end{tikzcd}
$$

interpreter semantics – an A program that runs C programs

denotational – a math program that maps C program to math objects

$$f(x) = \left(\int \pi^{-2} \cdot y \, dy\right) \times x$$

big-step semantics – a "simple" math program that maps C programs to C answers

small-step semantics – a really simple math program that simulates C programs turning into simpler C programs and ultimately C answers.

2-2

$2 + 1 \times 3 + 6$ big step $\rightarrow$ 11

$\Downarrow$

$2 + 3 + 6$

$\Downarrow$ — small-step

$5 + 6$

$\Downarrow$

11

---

r — the base relation on terms

$(T \otimes F) \; r \; F$ $\quad$ "n" + "m" $r \; \ulcorner n + m \urcorner$

$\rightarrow_r$ — compatible closure of r

$F \otimes (T \otimes F) \rightarrow_r F \otimes F$

$\twoheadrightarrow_r$ — the reflexive & transitive closure of $\rightarrow_r$

$x \twoheadrightarrow_r x$ $\quad$ $x \twoheadrightarrow_r y$

$y \twoheadrightarrow_r z$

then $x \twoheadrightarrow_r z$

$=_r$ — the symmetric closure of $\twoheadrightarrow_r$

$x \; R \; y$

$\Rightarrow y \; R \; x$

$eval_r$ — a "function" from terms to answers

$e = T$ $\quad$ $a = T$

$\mid F$ $\quad$ $\mid F$

$eval_r(e) = x$ $\quad$ $\mid e \otimes e$

s.t. $e \twoheadrightarrow_r x \in$ answer

$$eval_r(e) = \begin{cases} T & \text{if } e =_r T \\ F & \text{if } e =_r F \end{cases}$$

I ~~&~~ have program X, what will it do?

"$\text{evalr}(X) = ?$"

My compiler says X is A, but is that right?

"$\text{evalr}(X) = A$"?

Is it always the case that every program has one answer?

$$\forall e.\ \exists a.\ \text{evalr}(e) = a$$

"All programs have at least one answer"

$$\forall e.\ \forall a, a'.\ \text{evalr}(e) = a \wedge \text{evalr}(e) = a' \rightarrow a = a'$$

= eval is a function

"All programs have zero or one answer"

$T \Rightarrow T = T$
$T \Rightarrow F = F$
$F \Rightarrow T = T$
$F \Rightarrow F = T$

Do X and Y do the same thing?

"$\text{evalr}(X) = \text{evalr}(Y)$"

"$\forall \text{input}.\ \text{evalr}(X\ \text{in}) = \text{evalr}(Y\ \text{in})$"

2-4/

compatible closure means "find work"

$$
\begin{tikzcd}
 & M(X+Y) \arrow[dl] \arrow[dr] & \\
M'(X'+Y) \arrow[dr, dotted, two heads] & & (X+Y') \arrow[dl, dotted] \\
 & A &
\end{tikzcd}
$$

$\forall M, N \in$ terms. If $M =_r N$, exists L,
$M \twoheadrightarrow_r L$ and $N \twoheadrightarrow_r L$

Church-Rosser

the diamond property

$$
\begin{tikzcd}
 & M \arrow[dl] \arrow[dr, two heads] & \\
A \arrow[rr, phantom, "=_r"] \arrow[dr, two heads] & & B \arrow[dl, two heads] \\
 & L &
\end{tikzcd}
$$