Theorem: Every string is perfectly cromulent

Proof: Let w be an arbitrary string.
Assume, for every string x such that $|x| < |w|$, that x is perfectly cromulent.
There are two cases to consider.

- Suppose $w = \varepsilon$.

  Therefore, w is perfectly cromulent.

- Suppose $w = ax$ for some symbol a and string x.
  The induction hypothesis implies that x is perfectly cromulent.

  Therefore, w is perfectly cromulent.

In both cases, we conclude that w is perfectly cromulent. □

Lemma: For all strings w, y, z: $\underline{(w \circ y)} \circ z = w \circ (y \circ z)$

Proof: Let w, y, z be arbitrary strings.

IH: Assume $(x \circ y) \circ z = x \circ (y \circ z)$ for all strings x shorter than w.

There are two cases:

- $w = \varepsilon$

$$
\begin{aligned}
(w \circ y) \circ z &= (\varepsilon \circ y) \circ z && [w = \varepsilon] \\
&= y \circ z && [\text{def} \circ] \\
&= \varepsilon \circ (y \circ z) && [\text{def} \circ] \\
&= w \circ (y \circ z) && [w = \varepsilon]
\end{aligned}
$$

- $w = ax$ for some symbol a + string x

$$
\begin{aligned}
(w \circ y) \circ z &= ((a \cdot x) \circ y) \circ z && [w = ax] \\
&= (a \cdot \overset{u}{\boxed{(x \circ y)}}) \circ z && [\text{def} \circ] \\
&= a \cdot (\overset{u}{\boxed{(x \circ y)}} \circ z) && [\text{def} \circ] \\
&= a \cdot (x \circ (y \circ z)) && \text{IH} \\
&= (ax \circ (y \circ z)) && [\text{def} \circ] \\
&= w \cdot (y \circ z) && [w = ax]
\end{aligned}
$$

Therefore, $(w \circ y) \circ z = w \circ (y \circ z)$

LANGUAGES = sets of strings over Σ

$\emptyset$

$\{\varepsilon\}$ $\qquad$ Σ^* = all strings over Σ

$\cancel{\varepsilon}$

$\{w \in \{0,1\}^* \mid w \text{ has even \# of 1s}\} = \{\varepsilon, 00, 101, \ldots\}$

$\{\text{BMO}\}$

$\{\text{FINN}, \text{JAKE}, \text{ICEKING}\}$

$\{w \in \{0,1\}^* \mid w \text{ is binary for prime \#}\}$

$L = A \cup B$ $\qquad$ All Python programs

$L = A \cap B$ $\qquad$ All Python programs that ∞ loop

$L = \overline{A} = \Sigma^* \setminus A$

$L = A \bullet B = \{x \bullet y \mid x \in A \text{ and } y \in B\}$

$\{\text{FIRST}, \text{SECOND}, \text{THIRD}\} \bullet \{\text{BASE}, \text{PLACE}\}$

$\{0\}^* \bullet \{1\}^*$

$\emptyset \bullet L = \emptyset$ $\qquad$ $\{\varepsilon\} \bullet L = L$

L^* – Kleene star $= \{\varepsilon\} \cup L \cup L \bullet L \cup L \bullet L \bullet L \cup \ldots$

$w \in L^* \iff w = \varepsilon$ or $w = xy$
for some $x \in L$
$y \in L^*$

Is L^* always infinite?

$\emptyset^* = \{\varepsilon\} \cup \emptyset \cup \emptyset \bullet \emptyset \cup \ldots = \{\varepsilon\}$

$\{\varepsilon\}^* = \{\varepsilon\} \cup \{\varepsilon \bullet \varepsilon\} \cup \ldots = \{\varepsilon\}$

Lemma 2.1. *The following identities hold for all languages A, B, and C:*

(a) $A \cup B = B \cup A$.

(b) $(A \cup B) \cup C = A \cup (B \cup C)$.

(c) $\varnothing \bullet A = A \bullet \varnothing = \varnothing$.

(d) $\{\varepsilon\} \bullet A = A \bullet \{\varepsilon\} = A$.

(e) $(A \bullet B) \bullet C = A \bullet (B \bullet C)$.

(f) $A \bullet (B \cup C) = (A \bullet B) \cup (A \bullet C)$.

(g) $(A \cup B) \bullet C = (A \bullet C) \cup (B \bullet C)$.

Lemma 2.2. *The following identities hold for every language L:*

(a) $L^* = \{\varepsilon\} \cup L^+ = L^* \bullet L^* = (L \cup \{\varepsilon\})^* = (L \setminus \{\varepsilon\})^* = \{\varepsilon\} \cup L \cup (L^+ \bullet L^+)$.

(b) $L^+ = L \bullet L^* = L^* \bullet L = L^+ \bullet L^* = L^* \bullet L^+ = L \cup (L^+ \bullet L^+)$.

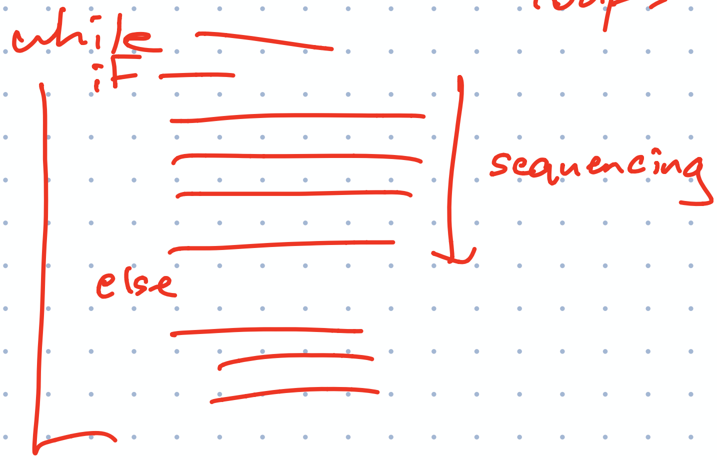
(c) $L^+ = L^*$ *if and only if* $\varepsilon \in L$.

Lemma 2.3 (Arden's Rule). *For any languages A, B, and L such that $L = A \bullet L \cup B$, we have $A^* \bullet B \subseteq L$. Moreover, if A does not contain the empty string, then $L = A \bullet L \cup B$ if and only if $L = A^* \bullet B$.*

Regular languages

L is regular <u>means</u> either

	$L = \varnothing$	$\to \varnothing$	
	$L = \{w\}$	$\to w$	for some string w
if then else	$L = A \cup B$	$\to A+B$	A, B regular
seq:	$L = A \bullet B$	$\to AB$	A, B regular
loops	$L = A^*$	$\to A^*$	A regular



Regular expressions

$0 + 10^*$

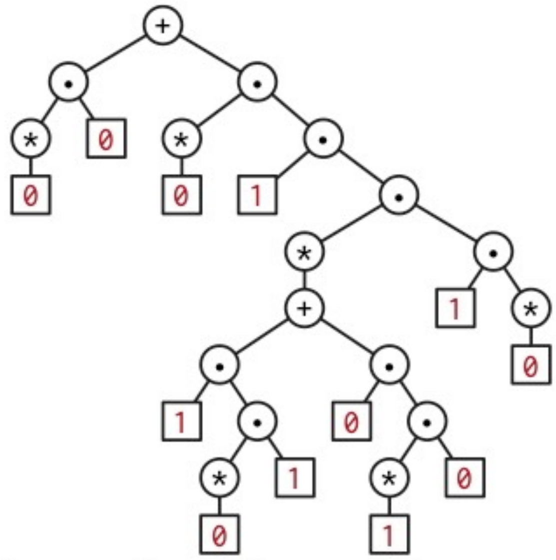
$= \{0\} \cup (\{1\} \bullet (\{0\})^*)$

Alternating 0's and 1s

Good: ε, 1, 0, 101, 010101, 01010, ...

Bad: 11, 0100, 01101, ...

$$\begin{aligned}&\varepsilon\\+\ &0\,(10)^*(1+\varepsilon)\\+\ &1\,(01)^*(0+\varepsilon)\end{aligned} \quad = (0+\varepsilon)(10)^*(1+\varepsilon)$$



A regular expression tree for $\mathtt{0}^*\mathtt{0} + \mathtt{0}^*\mathtt{1}(\mathtt{1}\mathtt{0}^*\mathtt{1} + \mathtt{0}\mathtt{1}^*\mathtt{0})^*\mathtt{1}\mathtt{0}^*$

Proof: Let R be an arbitrary regular expression.
Assume that **every regular expression smaller than** R is perfectly cromulent.
There are five cases to consider.

- Suppose $R = \varnothing$.

  Therefore, R is perfectly cromulent.
- Suppose R is a single string.

  Therefore, R is perfectly cromulent.
- Suppose $R = S + T$ for some regular expressions S and T.
  The induction hypothesis implies that S and T are perfectly cromulent.

  Therefore, R is perfectly cromulent.
- Suppose $R = S \bullet T$ for some regular expressions S and T.
  The induction hypothesis implies that S and T are perfectly cromulent.

  Therefore, R is perfectly cromulent.
- Suppose $R = S^*$ for some regular expression. S.
  The induction hypothesis implies that S is perfectly cromulent.

  Therefore, R is perfectly cromulent.

In all cases, we conclude that w is perfectly cromulent. □