$L = \{ x\bar{x} \mid x \in \{0,1\}^* \}$ $\quad \bar{x}$ means "not"

if $x = 0^i y$,

Let $D_i = \frac{d}{d0^i} L$ $\quad$ then $x\bar{x} = 0^i y 1^i \bar{y}$

$D_i = \{ y 1^i \bar{y} \mid y \in \{0,1\}^* \}$

this has i 1's in the middle,

so these are all different.

Shown.

Grammars

Another general way to specify languages.

more complex than regex.

e.g. $\{a^n b^n\}$ has a grammar.

A grammar for english sentences like

I love mom

you want candy

we see candy

these are all NVO

As a grammar:

$$S \to NVO$$ means "or"

$$N \to I \mid you \mid we$$

$$V \to love \mid want \mid see$$

$$O \to mom \mid candy$$

Each $\to$ is a "production"

We can write derivations, like:

$$S \Rightarrow NVO \Rightarrow I\ VO \Rightarrow I\ love\ O \Rightarrow I\ love\ mom$$

Fancier one

$$S \to NVO \mid NVMO$$

$$N \to I \mid you \mid we$$

$$V \to love \mid want \mid see$$

$$O \to mom \mid candy$$

$$M \to the \mid my$$

now we can derive "I see my mom"

We'll usually do ones like

$$G: \quad \begin{aligned} S &\to aS \mid X \\ X &\to bX \mid \varepsilon \end{aligned}$$

We start with S, and do substitutions using $\to$'s.

we can derive aaabb

$$S \Rightarrow aS \Rightarrow aaS \Rightarrow aaaS \Rightarrow aaaX$$
$$\Rightarrow aaabX \Rightarrow aaabbX \Rightarrow aaabb\varepsilon$$
$$= aaabb$$

The set of all strings derivable using the grammar is <u>the language generated by G</u>,

$$L(G)$$

$$G: \quad S \to aSb \mid \varepsilon$$

some derivations

$$S \Rightarrow aSb \Rightarrow a\varepsilon b = ab$$
$$S \Rightarrow aSb \Rightarrow aaSbb \Rightarrow aabb$$
$$S \Rightarrow aSb \Rightarrow aaSbb \Rightarrow aaaSbbb \Rightarrow aaabbb$$

$$L(G) = \{a^n b^n\}$$

Fancier:

$$\underbrace{S \to aSb}, \mid \varepsilon$$

$$aS \to Sa$$

Now, we can get $a^n b^n$, but also $abab$

$$S \Rightarrow aSb \Rightarrow Sab \Rightarrow aSbab \Rightarrow abab.$$

$L(G)$ is hard to describe

Terms: Alphabet letters (a, b, etc) are called <u>terminal symbols</u>

The others $(S, X, Y, T, \ldots)$ are called <u>nonterminal symbols</u>

<u>Formal description of a grammar</u>

<u>Def</u> A <u>grammar</u> is a tuple $G = (V, \Sigma, S, P)$

where V is a nonempty set of nonterminal symbols
Σ . . . . terminal symbols

$S \in V$ the starting symbol

P is a set of productions,

each looks like $x \to y$

where x & y are strings on $V \cup \Sigma$

and x uses at least one nonterminal.

$aS \to b$ is allowed

$\cancel{a \to ba}$ is not allowed.

$$G: \begin{array}{l} S \to aS \mid X \\ X \to bX \mid \varepsilon \end{array}$$

Formally:

$$G = \Big(\{S, X\}, \{a, b\}, S, \{ S \to aS, S \to X, X \to bX, X \to \varepsilon \} \Big)$$

Formally $x \Rightarrow y$ means we can obtain y by doing a single production on some piece of x.

We write $x \Rightarrow^* y$ when we can obtain y from x after several derivation steps.

Def The language of G is

$$L(G) = \{ x \in \Sigma^* \mid S \Rightarrow^* x \}$$

all strings x

such that

we can start with S and derive x.