

chapter 02

Context free Languages.

what did we learn so far?

- DFA / NFA computers with extremely limited memory
- Regular expressions
 - Recipe to describe languages that are regular.

How about creating slightly powered computational model?

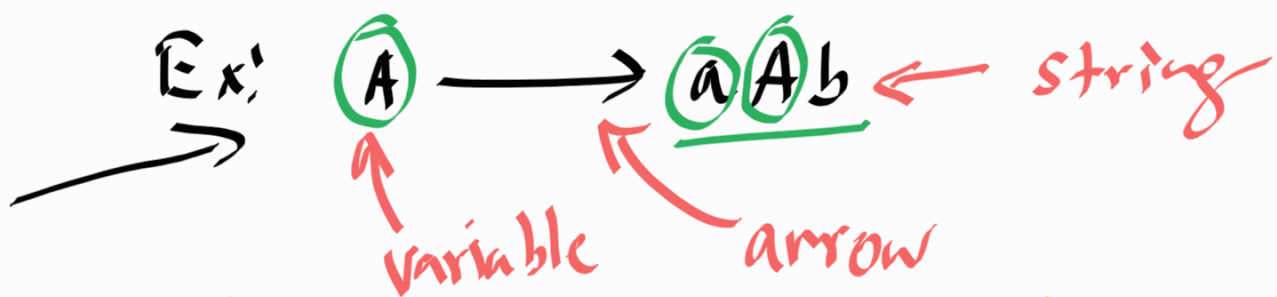
Instead going directly to the new computation model, let's try to learn about something called context free grammar.

Context free grammar to context free languages is what regular expression are to regular languages.

* context free grammar is a collection of substitution rules. also known as productions.

* Each rule appears as a line in the grammar.

* Each line comprises of a symbol, arrow, and a string.



* The string can consist of variables, and other symbols called terminals.

* Variables are often represented using capital letters, terminals are represented using lowercase letters, numbers or special symbols

$A \longrightarrow aAb$

Diagram illustrating a production rule in a context-free grammar. The variable A is circled in green and labeled "variable" with a red arrow. The string aAb is underlined in green and labeled "string" with a red arrow. The arrow between A and aAb is labeled "arrow" with a red arrow.

* One variable is designated starting variable.

what we know now

$F = \{0^n 1^n \mid n \geq 0\}$ is not regular.

1. $A \longrightarrow \underline{0A1}$
2. $A \longrightarrow B$
3. $B \longrightarrow \underline{\epsilon}$

sometimes we use
 Δ , $\#$

$$\underline{A} \Rightarrow \underline{0A1} \Rightarrow \underline{00A11}$$

\Downarrow

$$000B111 \Leftarrow 000A111$$

\Downarrow

$$000\epsilon111 \Rightarrow 000111$$