PDA: Equivalence with CFG

PDA: Equivalence with CFG

- Theorem:

  A language L is a CFL iff $\exists$ PDA M s.t. M recognizes L.

  - $\Rightarrow$: L context free, then $\exists$ PDA M recognizes L.
  - Proof: Convert a CFG into an equivalent PDA.
    Mimic derivations in CFG.

Procedure of Converting CFG to PDA

- Push special mark \$ and the start symbol S to the stack
- Repeat forever

  If the top of the stack is

  - variable A: (apply a rule for A)

    substitute A by RHS of a rule.
  - terminal a: (check applicability)

    compare the next input symbol with a

    * If match, repeat.
    * If not match, stop the branch.
  - symbol \$: enter the accept state.

Example of Converting CFG to PDA

- Very straightforward.

  $S \to aTb \mid b$

  $T \to Ta \mid \epsilon$

Non-Context-Free Languages

- Properties of CFL: strings longer than *pumping length* can be pumped.
- Theorem: **Pumping lemma for CFL**

  Let L be context-free. Then $\exists$ pumping length n such that $\forall s \in L$ with $|s| \geq n$, there exists $u, v, x, y, z \in \Sigma^*$ such that

  1. $s = uvxyz$
  2. $|vxy| \leq n$
  3. $|vy| \geq 1$
  4. $\forall i \geq 0, uv^ixy^iz \in L$
- Proof: