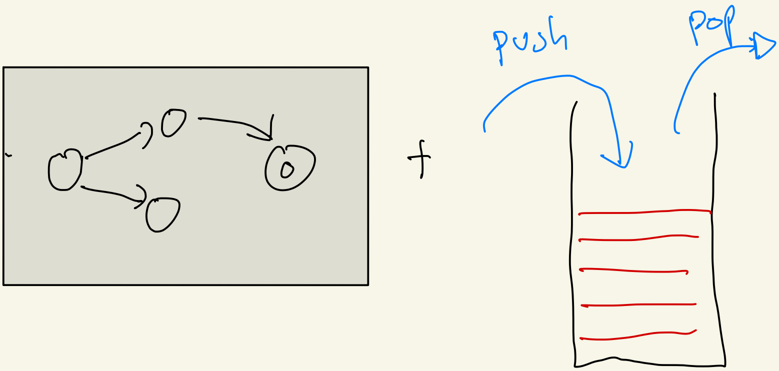
Push down automata (PDA)

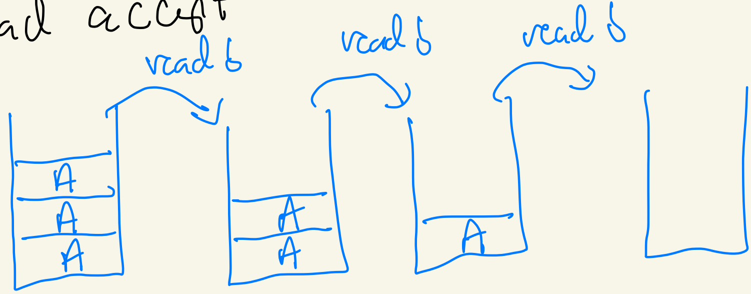
like an NFA with an ∞-stack



Idea for $L=\{a^n b^n \mid n\geq 0\}$:

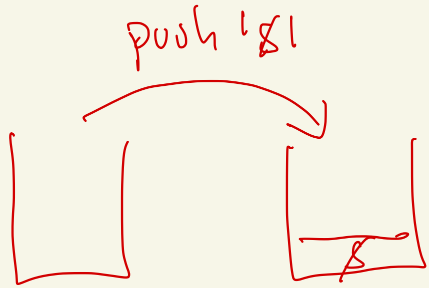
while reading a's: push 'A' on stack
when seeing a b: pop 'A' from stack
while reading b's: pop 'A' from stack
If stack becomes empty and w is read accept

aaabbb $\rightsquigarrow$



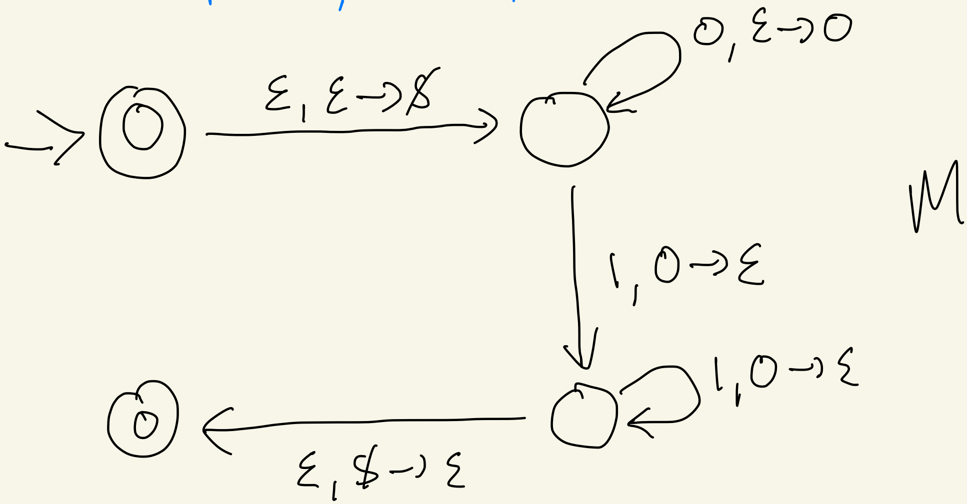
Problem: how do we detect that the stack is empty?

Solution: push a special symbol $\$ \notin \Sigma$ on stack initially



NB: we need $ as we cannot ask for 'empty stack'

PDA for $L = \{0^n 1^n \mid n \geq 0\}$



Claim $L(M) = L$

Formal definition of a PDA:

$$M = (Q, \Sigma, \Gamma, \delta, q_0, F)$$

as NFA

stack alphabet

$$\delta : Q \times (\Sigma \cup \varepsilon) \times (\Gamma \cup \varepsilon) \to P(Q \times (\Gamma \cup \varepsilon))$$

Non-deterministic!

M accepts w if $w = w_1 w_2 \cdots w_m$ where $w_i \in \Sigma_\varepsilon$ $(\Sigma \cup \varepsilon)$
and $\exists$ states $r_0, r_1, r_2, \ldots, r_m \in Q$ and strings $s_1, s_2 \cdots s_m \in \Gamma^*$

1. $r_0 = q_0$ and $s_0 = \varepsilon$ ← start with empty stack
2. For $i = 0, 1, 2, \ldots, m-1$: $(r_{i+1}, b) \in \delta(r_i, w_{i+1}, a)$
   where $s_i = at$ and $s_{i+1} = bt$ for some $a, b \in \Gamma_\varepsilon$ $t \in \Gamma^*$

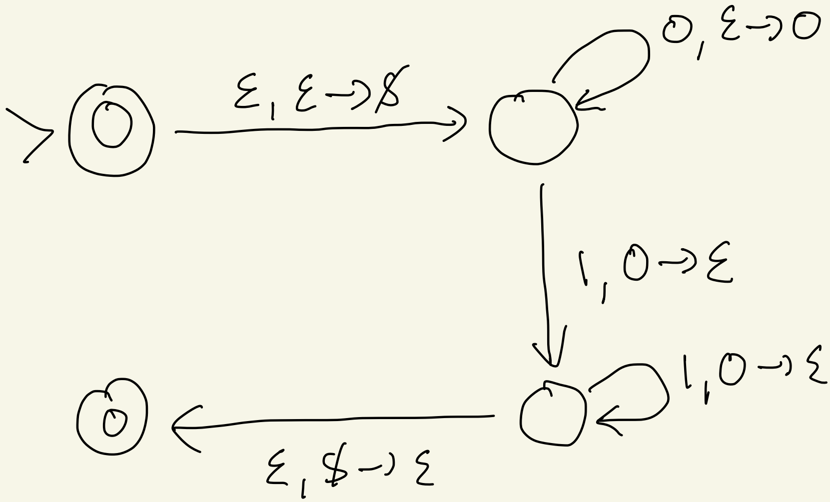
   i.e M moves properly according to state, stack and next input symbol (or ε)
3. $r_m \in F$ (so M reaches an accepting state after processing w)

$$L(M) = \{ w \in \Sigma^* \mid M \text{ accepts } w \}$$

Digraph representation:

$a, b \to c$ $\iff$ when reading $a \in \Sigma_\varepsilon$ from input
M may replace b on the top of the stack by c

each of a, b, c may be the empty string!



M

Claim $L(M) = L$

M can only reach an accept state <u>and</u> at the same time have read all of w if $w \in L$

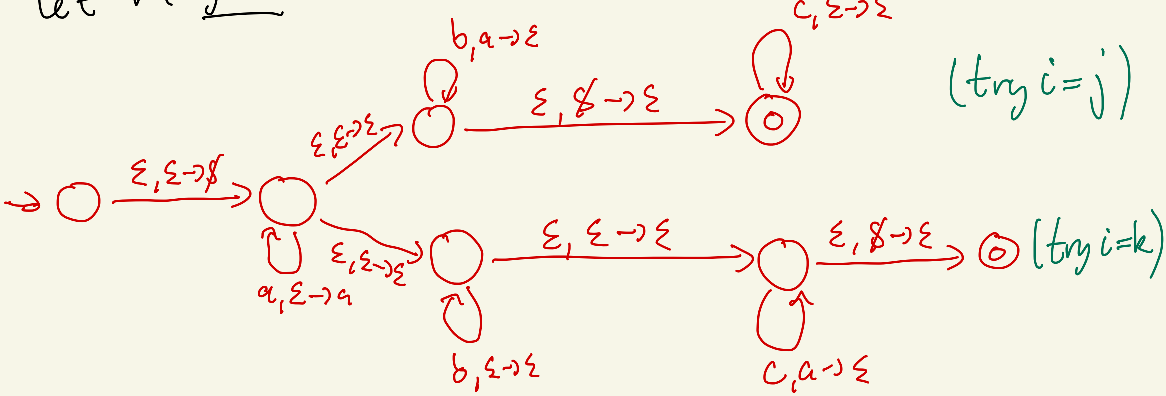
Example 2.16

$L' = \{ a^i b^j c^k \mid i,j,k \geq 0 \text{ and } i=j \text{ or } i=k \}$

NB L' is not regular: $L' \cap a^* b^* = \{ a^n b^n \mid n \geq 0 \}$

How to construct PDA M with $L(M) = L'$?

Let M <u>guess</u> whether $i=j$ or $i=k$



Check for $L(M) = L'$:

- $L' \subseteq L(M)$
- $L(M) \subseteq L'$

$a^3 b^3 c^2 \in L$	$a^3 b^2 c^2 \notin L$
$a^3 b c^3 \in L$	$a^3 b c^2 \notin L$
$\varepsilon \in L$	$a^2 b^3 c^4 \notin L$

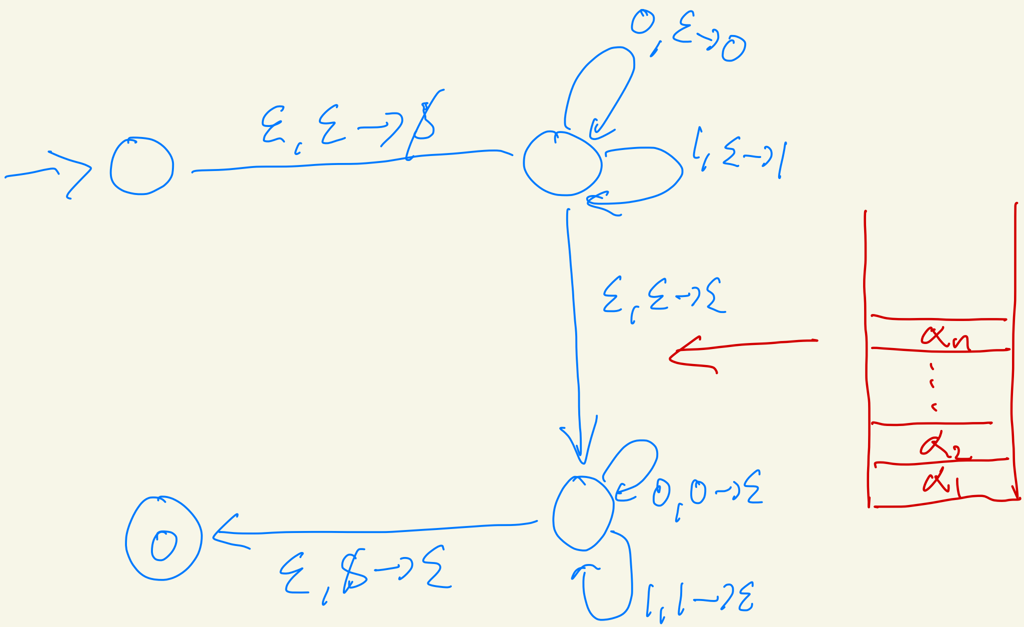
Example 2.18

$L'' = \{ww^R \mid w \in \{0,1\}^*\}$

easy to show via pumping lemma that L'' is not regular:

- suppose M is a DFA with $L(M) = L''$ and let $p = \#$states in M.
- let $s = \underbrace{0^p 1^p}_{w} \underbrace{1^p 0^p}_{w^R}$
- adversary gives us x, y, z s.t. $|xy| \le p$, $|y| = j > 0$ and $s = xyz$
- Then $xz = 0^{p-j} 1^p 1^p 0^p \notin L''$

$$L'' = \{ w w^R \mid w \in \{0,1\}^* \}$$



$$w = \alpha_1 \alpha_2 \cdots \alpha_n \alpha_n \cdots \alpha_2 \alpha_1$$

Theorem 2.20

L is context-free

$\Updownarrow$

$L = L(M)$ for some PDA

we prove this later