Turing Machines

- more powerful than PDA's
- what could it have?
Example: $A = \{ a^i b^i c^i \mid i \geq 0 \}$
Example: \( B = \{ w\#w \mid w \in \{0,1\}^* \} \)
Def 3.3:

A **Turing machine** (TM) is a 7-tuple \((Q, \Sigma, \Gamma, \delta, q_0, q_{\text{accept}}, q_{\text{reject}})\), where

- \(Q\) is the (finite) set of states
- \(\Sigma\) is the (finite) input alphabet, not containing \(\square\)
- \(\Gamma\) is the (finite) tape alphabet, \(\square \cup \Sigma \subseteq \Gamma\)
- \(\delta: Q - \{q_{\text{reject}}\} \times \Gamma \rightarrow Q \times \Gamma \times \{L, R\}\) is the transition function
- \(q_0 \in Q\)
- \(q_{\text{accept}} \in Q\) is the accept state
- \(q_{\text{reject}} \in Q - \{q_{\text{accept}}\}\) is the reject state
Computation of Turing machines

- first we define a configuration:
  
  $uvq$ - means the tape contains $uv$, the state is $q$, and the machine reads the first symbol of $v$

- suppose configuration is $uaqbv$ and $\delta(q,b) = (p,c,R)$

We say that $uaqbv$ yields $uacpv$
Computation of Turing machines

- first we define a configuration:
  
uqv - means the tape contains uv, the state is q, and the machine reads the first symbol of v

- start configuration:

- accepting configuration:

- rejecting configuration:

Note: accepting/rejecting configurations are halting
Computation of Turing machines
- first we define a configuration:
  \[ uvq \] means the tape contains \( uv \), the state is \( q \), and the machine reads the first symbol of \( v \).

A TM \( M \) accepts \( w \) if there exists a sequence of configurations \( C_0, C_1, \ldots, C_k \) s.t.

1) \( C_0 \) starting config.
2) \( C_i \) yields \( C_{i+1} \), \( \forall i \in \{0, \ldots, k-1\} \)
3) \( C_k \) accepting config.

The language of a TM \( M \) is the set of strings that \( M \) accepts/recognizes.
Def 3.5: A language is **Turing-recognizable** if there is some TM that recognizes it.

(also known as recursively enumerable)

- accepts the strings in the language
- rejects or loops on strings not in the language

Def 3.6: A language is **Turing-decidable** if there is some TM that decides it.

(also known as recursive)

→ for every input, ends in the accepting or the rejecting state
   (never infinite loop)

E.g. \{a^i b^i c^i \mid i \geq 0\} is T-decidable
and T-recognizable

[Diagram showing relationships between Turing-decidable and Turing-recognizable sets]
Example: \( A = \{ 0^n \mid n=2^k \text{ for some } k \geq 0 \} \)