Instantaneous descriptions

- To fully specify the configuration of a TM at a moment in time, need to state:
  - What is written on the tape
  - What state the machine is in
  - What cell the tape-head is scanning
- Note that although the tape is infinite, the number of non-blank symbols on the tape is always finite (after a finite number of steps)
- An instantaneous description of a TM is a way of specifying these things
- **Definition:** Let $M = (Q, \Sigma, \Gamma, q_0, q_{accept}, q_{reject}, \delta)$ be a TM s.t. $\Gamma \cap Q = \emptyset$. An instantaneous description (ID) of M is a string AqB, where
  - $A, B \in \Gamma^*$
  - $q \in Q$
  - Neither the first symbol of A nor the last symbol of B are $\sqcup$
- Note: the requirement that $Q \cap \Sigma = \emptyset$ is without loss of generality: just rename the states so that it is true.
- An ID AqB denotes:
  - The tape contains $\ldots \sqcup \sqcup AB \sqcup \sqcup \ldots$
  - The TM is in state q
  - The tape-head is over the left-most symbol of B
- **Example:** If a TM has tape contents $\ldots \sqcup \sqcup 0101 \sqcup \sqcup \ldots$, state q, and the tape head is scanning the second 0, then the ID is $01q01$.
- **Definition:** Let M be a TM, and let $w \in \Sigma^*$. Then $I_M(w) = q_0 w$.
- $I_M(w)$ is the ID of the initial configuration of M on input w
- **Definition:** Let M be a TM, and let C, C' be IDs of M. We write $C \vdash_M C'$ (or just $C \vdash C'$) if C' is the ID obtained from C after one step of M.
- **Example:** If a TM M has a transition $\delta(q, 0) = (q', 1, L)$ then $01q01 \vdash_M 0q'111$
- **Definition:** If M is a TM and C, C' are IDs of M, write $C \vdash_M^* C'$ if C' can be reached in zero or more steps of from C; that is, if one of the following holds:
  - $C = C'$

  - $C \vdash_M C'$
  - There exist $k > 0$ and IDs $C_1, \ldots, C_k$ s.t. $C \vdash_M C_1$, $C_k \vdash_M C'$, and for all $1 \leq i < k$, $C_i \vdash_M C_{i+1}$.

- **Definition:** An ID AqB is accepting if $q = q_{accept}$. It is rejecting if $q = q_{reject}$.

- **Definition:**
  - A TM M accepts $w \in \Sigma^*$ if $I_M(w) \vdash^* C$ for some accepting ID C.
  - A TM M rejects $w \in \Sigma^*$ if $I_M(w) \vdash^* C$ for some rejecting ID C.
  - A TM M halts on input $w \in \Sigma^*$ if M accepts w or if M rejects w.

Languages

- **Definition:** A language over an alphabet Σ is a (finite or infinite) set of strings over Σ. In other words, a language over Σ is a subset of Σ^*.
- **Definition:** For a TM M, the language of M is $L(M) = \{w \mid M \text{ accepts } w\}$
- **Definition:** A language L is recognizable if there is a TM M s.t. $L = L(M)$
- **Definition:** A language L is decidable if there is a TM M s.t.
  - $L = L(M)$
  - M halts on every input
- Questions: are there languages that are not recognizable? Are there languages that are recognizable but not decidable?

Functions

- **Definition:** A function $f : \Sigma^* \mapsto \Sigma^*$ is computable if there exists a TM M s.t. for all $w \in \Sigma^*$, $I_M(w) \vdash^* q_{accept} f(w)$
- Note: we could just as easily have used q_{reject} in the above definition. The important thing is that at some point M halts, and when it does so the tape contains the result of the function evaluated at w.
- Question: are there functions that are not computable?

The Church-Turing thesis

- Many ways of stating the thesis
- One way: "Anything that can be computed by an algorithm (in the intuitive sense of the word) can be computed by a Turing machine"

- Note: this is not a mathematical statement (it can't be proved)
- In the next week or so, I will attempt to convince you that it is true
- One convincing argument is the following fact: anything that can be computed by a random access machine can be computed by a TM (A RAM is like a real-world computer with infinite memory)
- Eventually, we will assume that the Church-Turing thesis is true, and adopt the following definition:
  - **Definition:** An algorithm is a Turing machine