CSE 396 Lecture Tue 4/9 Spring 2019

Top Hat 0371

A Turing Machine (TM) adds to a FA the capabilities to:

- change a char in the string on its tape.
- move a scanning head left (L) on the tape as well as right (R) or stay (S).

Text puts these in different instructions ("4-tuple notation") but we and Turing Kit combine them ("5-tuple notation").

Defn: A TM is a 7-tuple $M = (Q, \Sigma, \Gamma, \delta, \sqcup, s, F)$ where:

Q is a finite set of states

Σ is a finite input alphabet (or I/O alphabet)

Γ, which includes Σ and the blank $\sqcup$, is the work alphabet.

$s \in Q$ is the start state;

$F \subseteq Q$ is the set of desired final states, aka. accepting states.

Turing Kit allows F to be general, but we will design to the text's "normal form" where $F = \{q_{acc}\}$ and q_{rej} is the only other halting state.

And

$$\delta \subseteq \underbrace{Q \times \Gamma}_{\text{read}} \underset{\text{"loud comma"}}{\times} \underbrace{\Gamma}_{\text{write}} \times \underbrace{\{L, R, S\} \times Q}_{\text{execute.}}$$

Typical instruction: $(p, c; d, D, q)$ $p \in Q \setminus \{q_{acc}, q_{rej}\}$ $D \in \{L,R,S\}$ $q \in Q$. $c, d \in \Gamma$ D = "direction"

In a diagram $p \xrightarrow{(c/d,D)} q$

The machine is deterministic (a DTM) if the instructions define δ as a function

$$\delta : Q \setminus \{q_{acc}, q_{rej}\} \times \Gamma \longrightarrow \Gamma \times \{L, R, S\} \times Q$$

The rest of the lecture was a long demo of Turing machines with 1 and 2 tapes, incl. PDAs.

Given a DTM M and an input $x \in \Sigma^*$, $M(x)$ denotes a computation path that might

- end in q_{acc}: M accepts x, $x \in L(M)$,
- end in q_{rej}: M halts and rejects x, OR
- might never halt at all: $M(x)\uparrow$ ("diverges"), also $x \notin L(M)$