Turing Machines – A Standard Model

The evolution of machines/automata

- finite memory (encoded in state) = DFAs/NFAs
- unbounded stack – NPDAs/DPDAs
- unbounded memory – TMs.

Def. A *Turing machine* $T = \langle Q, \Sigma, \Gamma, \frac{\delta}{\Delta}, q_0, \square, F \rangle$

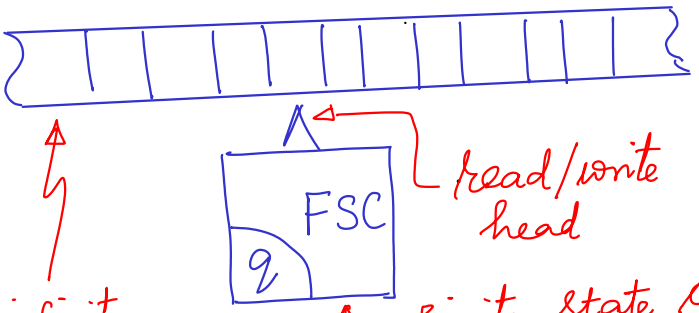
$F \subseteq Q$

$\Sigma \subseteq \Gamma - \{\square\}$

$\square$: blank symbol

$\delta : Q \times \Gamma \to Q \times \Gamma \times \{L, R\}$ (partial function)

$\Delta \subseteq_f (Q \times \Gamma) \times (Q \times \Gamma \times \{L, R\})$



infinite tape

read/write head

finite state control

(which is used for storing input, as scratch pad and also (in the case of turing machines as transducers) for storing the output.

Def. An **instantaneous description** (ID) of a TM is a **sequence of the form** xqy where $x, y \in \Gamma^*$ such that $x \neq \square^* \neq y$ and (i) there is no non-blank symbol either to the left of x or to the right of y (ii) the first symbol of x and the last symbol of y are both non-blank and (iii) the first symbol of y is the symbol immediately under the read/write head. (iv) q is the current state of the FSC. An ID is also called a **configuration**.

Def A **move** of the TM from an ID $x_1 q_1 y_1$ to another ID $x_2 q_2 y_2$ is denoted $x_1 q_1 y_1 \vdash x_2 q_2 y_2$ where $x_1 = a_1 \cdots a_k$, $y_1 = b_1 \cdots b_\ell$ is possible provided one of the following holds.

(i) $\delta(q_1, b_1) = (q_2, c, R)$ and $x_2 = a_1 \cdots a_k c$, $y_2 = b_2 \cdots b_\ell$ or

(ii) $\delta(q_1, b_1) = (q_2, c, L)$ and $x_2 = a_1 \cdots a_{k-1}$, $y_2 = a_k c b_1 \cdots b_\ell$

Def. The **language** accepted by a Turing machine T is

$$L(T) = \{ x \in \Sigma^* \mid q_0 x \vdash^* y q_F z, \ y, z \in T^*, \ q_F \in F \}$$