Definition 1 (Turing machine). *A (non-deterministic)* Turing machine *(*NTM*) is given by a 7-tuple* $M = \langle Z, \Sigma, \Gamma, \delta, z_0, \Box, E \rangle$ *with:*

- Z *finite, non-empty set of* states
- $\Sigma \neq \emptyset$ *finite* input alphabet
- $\Gamma \supset \Sigma$ *finite* tape alphabet
- $\delta : (Z \setminus E) \times \Gamma \to \mathcal{P}(Z \times \Gamma \times \{\mathrm{L}, \mathrm{R}, \mathrm{N}\})$ transition function
- $z_0 \in Z$ initial state
- $\Box \in \Gamma \setminus \Sigma$ blank symbol
- $E \subseteq Z$ final states

Definition 2 (Configuration of an NTM). *A* configuration *of a Turing machine* $M = \langle Z, \Sigma, \Gamma, \delta, z_0, \Box, E \rangle$ *is given by a word* $k \in \Gamma^* Z \Gamma^+$.

Configuration $w_1 z w_2$ *means that the non-empty (i.e., visited) part of the tape contains the word* $w_1 w_2$*, the read/write head is on the first symbol of* w_2*, and the Turing machine is in state* z.

Definition 3 (Step of an NTM). *An NTM* $M = \langle Z, \Sigma, \Gamma, \delta, z_0, \Box, E \rangle$ *can go from configuration* k *to* k' *in one step* $(k \vdash_M k')$ *according to the following rules:*

$$a_1 \dots a_m z b_1 \dots b_n \vdash_M \begin{cases} a_1 \dots a_m z' c b_2 \dots b_n & \textit{if } (z', c, N) \in \delta(z, b_1), m \geq 0, n \geq 1 \\ a_1 \dots a_m c z' b_2 \dots b_n & \textit{if } (z', c, R) \in \delta(z, b_1), m \geq 0, n \geq 2 \\ a_1 \dots a_{m-1} z' a_m c b_2 \dots b_n & \textit{if } (z', c, L) \in \delta(z, b_1), m \geq 1, n \geq 1 \\ a_1 \dots a_m c z' \Box & \textit{if } (z', c, R) \in \delta(z, b_1), m \geq 0, n = 1 \\ z' \Box c b_2 \dots b_n & \textit{if } (z', c, L) \in \delta(z, b_1), m = 0, n \geq 1 \end{cases}$$

Definition 4 (Reachable configuration). *Configuration* k' *is* reachable *from configuration* k *with NTM* M *(written as* $k \vdash_M^* k'$*), if* $k = k'$ *or there are configurations* $k_0, \dots, k_n$ $(n \geq 1)$*, with*

- $k_0 = k$,
- $k_i \vdash_M k_{i+1}$ *for* $i \in \{0, \dots, n-1\}$*, and*
- $k_n = k'$.

Definition 5 (Accepted word of an NTM). *NTM* $M = \langle Z, \Sigma, \Gamma, \delta, z_0, \Box, E \rangle$ accepts the word w *if and only if* M *can reach a configuration with a* final state *from the* start configuration $z_0 w$ *in a finite amount of steps:*

$$M \textit{ accepts } w \textit{ iff. } z_0 w \vdash_M^* w_1 z w_2 \textit{ with } z \in E, w_1 \in \Gamma^*, w_2 \in \Gamma^+.$$

Special case: *for* $w = \varepsilon$ *the start configuration is* $z_0 \Box$.

Definition 6 (Accepted language of an NTM). *Let* M *be an NTM with input alphabet* Σ*. The* language accepted by M *is defined by*

$$\mathcal{L}(M) = \{w \in \Sigma^* \mid w \textit{ is accepted by } M\}.$$

Definition 7 (Deterministic Turing machine). *A* deterministic Turing machine *(DTM) is defined in almost the same way, but the transitions are deterministic, i.e., there is only one "effect" of each transition:* $\delta : (Z \setminus E) \times \Gamma \to Z \times \Gamma \times \{\mathrm{L}, \mathrm{R}, \mathrm{N}\}$.

The definition of a step of a DTM is almost the same as for an NTM, just replace every $\in$ *with* $=$*. The remaining definitions are all exactly analogous to those of NTMs.*