**Definition 1.** An oracle Turing machine is a TM $M$ that has a special (read-write) tape that we call $M$'s oracle tape and three special states $q_{\text{query}}, q_{\text{yes}}, q_{\text{no}} \in Q$. To execute $M$, we specify in addition to the input a language $O \subseteq \{0, 1\}^*$ that is used as the oracle for $M$. Whenever during the execution, $M$ enters the state $q_{\text{query}}$—instead of using the transition function to determine the next configuration—for the next configuration the machine moves into the state $q_{\text{yes}}$ if $q \in O$ and into the state $q_{\text{no}}$ if $q \notin O$, where $q$ denotes the contents of the special oracle tape, and the tape contents and the positions of the tape heads do not change.

Thus, regardless of the choice of $O$, a membership query to $O$ only counts as a single computation step.

If $M$ is an oracle machine, $O \subseteq \{0, 1\}^*$ is a language, and $x \in \{0, 1\}^*$ is an input, then we denote the output of $M$ on input $x$ and with oracle $O$ by: $M^O(x)$.

**Definition 2.** Let $O \subseteq \{0, 1\}^*$ be a language.

We let $P^O$ be the set of all languages that can be decided by a polynomial-time deterministic Turing machine with oracle access to $O$.

We let $NP^O$ be the set of all languages that can be decided by a polynomial-time nondeterministic Turing machine with oracle access to $O$.

We will use similar notation for variants of other complexity classes that are based on Turing machines with bounds on the running time, e.g., $\text{EXP}^O$. 