ORACLE TURING MACHINE

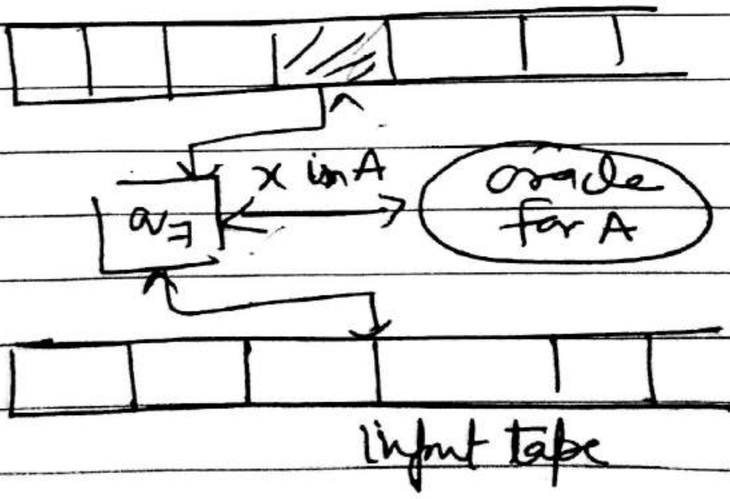
→ An Oracle Turing M/C (OTM) has a special read-write tape called M's oracle tape and three special states $q_{query}, q_{yes}, q_{no}$.

→ To execute M, we specify the input as usual; and a language $\boxed{O \subseteq \{0,1\}^*}$ that is used as an oracle for M.

→ While performing its computation, if M enters the state of q_{query} then M checks whether the contents of the oracle tape $w \in O$. if $w \in O$, M moves to the state q_{yes}, else it moves to q_{no}.

→ Regardless of the choice of O, a query like $w \in O$ counts for a single step of M

→ $M^O(x)$ denotes the output of the oracle Turing M/C M on input $x \in \{0,1\}^*$ with $O \subseteq \{0,1\}^*$ as ~~its~~ the oracle.



$M^O(x) =$

Def: A language $L \subseteq \{0,1\}^*$ is in the complexity class P^O (where O is an arbitrarily fixed language) if there exists a deterministic polytime oracle Turing M/C that decides L

Def: A language $L \subseteq \{0,1\}^*$ is in ~~the~~ complexity class NP^O (where O is an arbitrarily fixed language) if there exists a Non-deterministic polytime oracle Turing M/C that decides L.

Interactive proof System

→ Traditionally, the NP complexity class is defined as the set (class) of languages that are decided by non-deterministic polynomial-time Turing M/C's. but it can be equivalently thought of as the set of languages that have a polynomial-time verifier.

→ (Verifier, Certificate)

A verifier for a language L is a Turing M/C V, such that

$$L = \{ w \mid V \text{ accepts } \langle w, \underbrace{c}_{\text{Certificate for } w} \rangle \text{ for some string } c \}.$$

V is considered to be of polynomial time ~~if its~~

→ IPS were introduced by Goldwasser, Micali and Rackoff in their seminal 1985 paper "The knowledge complexity of Interactive proof system".

An Interactive ~~pa~~ Turing M/C (ITM) is a Turing M/C that has:

- * A read only input tape
- * A scratch ~~type~~ tape
- * A random tape
- * A read only communication tape
- * A write only communication tape

→ Contains an infinite sequence of random bits, which are read from left-to-right, and reading the next random bit is called flipping a coin.

BY COMBINING TWO ITM we can define a general two-party interactive protocol.

(Interactive protocol, Prover, Verifier)

An Interactive protocol is an orderd pair of ITMs (P,V), called the Prover and Verifier respectively such that:

→ P,V share the same input tape

→ p's write only communication tape is V's read-only communication tape and vice-versa.

→ p's computationally unbounded, whereas V's total internal computational time is polynomial in the length of the common input.

→ Two m|c's take turns being active, and V starts

→ During an active state, the m|c are some internal computation using the input, scratch, random and communication tapes writes to its write-only tape

→ Termination occurs when V either accepts or rejects the input by outputting accept or reject in its write tape and halting the protocol.

{Interactive proof system)

The Interactive protocol (P,V) is an IPS for language L if

* $w \in L \Rightarrow pr[(P,V) \text{ accepts } w] \geq 2/3$

* $w \notin L \Rightarrow pr[(P,V) \text{ accepts } w] \leq 1/3$

{IP class}

The interactive polynomial (IP) time class is the class of language that have an interactive proof system.