Def. \( P \) = set of polynomial time predicates

Input \( x \in \{0,1\}^n, \ |x| = n \)

Runtime \( p(n) \)

Def. \( FP \) = set of polynomial time functions.

Def. \( NP \) = non-deterministic polynomial time
- some configurations have 2 successors

\[ L \in NP \iff \exists \text{ poly time TM } M(x) \]
\[ \forall x \ (x \in L \iff M(x) \text{ has an accepting computation} \]

Def. A language \( L \subseteq \{0,1\}^* \)

And the worst-case polynomial time Turing Machines (TM) have a "clock" attached which make it explicitly poly time.

An alternate Def. A language \( L \in NP \iff \exists \text{ poly time predicate } R(x,y), \)
\( \forall p(n) \text{ polynomial}, \forall x \ (x \in L \iff (\exists y, |y| = p(|x|)) \ R(x,y) \)

Notation:
\[ NP = \Sigma^p_1 \]
\[ \text{coNP} = \Pi^p_1 = \{ L \in NP^p \mid \neg L \subseteq \{0,1\}^* \} \]
\( (\text{and } x \in L \iff (\forall y, |y| = p(|x|)) \neg R(x,y)) \)

- we can generalize \( \Sigma^p_i, \Pi^p_i, i \geq 1 \text{ polynomial time hierarchy} \)

\[ \text{PH} = \bigcup \Sigma^p_i = \bigcup \Pi^p_i \]

\( L \in \Sigma^p_i \iff \exists \text{ formula} \)
\( \exists y_1, \ldots, y_i, |1 \leq p_i(|x|) \quad \forall x, |y_2| \leq p_2(|x|) \quad \ldots \quad Qy_i, |y_i| = p_i(|x|) \ R(x,y_i) \)

where \( p_i \)'s are poly's

\[ \forall x \ (x \in L \iff \varphi(x) \text{ is true}) \]
L ∈ P (⇒) L ∈ P

Open: NP = coNP?

**Def. Probabilistic TM (PTM)**
- Some configurations have 2 successors
- Choice on successor is made at random

**Def. Bounded Probabilistic Poly Time (BPP)**
Let BPP ⇔ ∃ poly time PTM, M(x), s.t.
∀x (x ∈ L ⇔ Pr (M(x) accepts) ≥ \frac{2}{3})
and ∀x (x \notin L ⇔ Pr (M(x) accepts) ≤ \frac{1}{3})

Alternatively,
Let BPP ⇔ ∃ deterministic TM (DTM) poly time predicate R(x, y), ∃ poly p(n)

s.t. ∀x (x ∈ L ⇔ (∃ y ∈ \{0, 1\}^{\leq p(n)}) R(x, y) ≥ \frac{2}{3} \cdot 2^{p(n)})
and ∀x (x \notin L ⇔ (∃ y ∈ \{0, 1\}^{\leq p(n)}) R(x, y) ≤ \frac{1}{3} \cdot 2^{p(n)})

• BPP is unchanged if we replace \frac{1}{3}, \frac{1}{2} by \frac{2^{-n}}{2 \cdot \frac{n}{2}}, \frac{2^{r(n)}}{2 \cdot r(n)} or
  \frac{\frac{1}{2} \cdot \frac{n}{2}}{2 \cdot r(n)} \frac{2^{\frac{n}{2}}}{2 \cdot r(n)}

for any poly r(n)
Def. \(#P\) class of functions

\[
\text{f.e. } \#P \iff f : \{0,1\}^* \rightarrow \mathbb{N} \times \{0,1\}^*
\]

\[
f \in \#P \iff \exists \text{ poly time predicate } R(x, y), \exists \text{ poly p(n) s.t.}
\]

\[
\forall x \left( f(x) = (\# y \in \{0,1\}^* \mid R(x, y)) \right)
\]

Equivalently, \(\exists\text{ poly time NTM } M(x)\) s.t.

\[
\forall x \left( f(x) = (\#\text{ accepting computations of } M(x)) \right)
\]

- \(#P \leq \text{PSPACE}\)
  - \text{Open:} \text{P = PSPACE?}

Def. \(P^{\#P}\) (or more generally \(P^{\#X}\) where \(X \in \{0,1\}^*\)), where \(#P\) \((\text{or } X)\) is an oracle.

Poly time TM that is allowed to "query" \(f \in \#P\) (or \(X\)).

for \(#P\), query is some \(z \in \{0,1\}^*\) and answer is \(f(z)\)

**Easy Thm's**

- \(NP \leq P^{\#P}\)
- \(coNP \leq P^{\#P}\)
- \(BPP \leq P^{\#P}\)

**Today's Thm.** Polynomial Hierarchy (all classes) \(PH \leq P^{\#P}\)

**Overall Picture**:

\[
P \leq \text{NP} \leq \text{coNP} \leq \Sigma_2^P \leq \text{PH} \leq P^{\#P} \leq \text{PSPACE}
\]