The Class $\mathbb{P}$

\[ \mathbb{P} = \bigcup_{k} \text{TIME}(n^k) \]
RELPRIME = \{ \langle x, y \rangle \mid x \text{ and } y \text{ are relatively prime} \}

RELPRIME \in P
PATH = \{ \langle G, s, t \rangle \mid G \text{ is a directed graph that has a path from } s \text{ to } t \}\}

PATH \in P
Def: \( \text{NTIME}(t(n)) \)

set of all languages decidable by an \( O(t(n)) \) time nondeterministic TM

\[ \bigcup_{k} \text{NTIME}(n^k) \]
HAMPATH = \{ \langle G, s, t \rangle \mid G \text{ has a Hamiltonian path (that visits each node exactly once) from } s \text{ to } t \\}
HAMPATH = \{ \langle G, s, t \rangle \mid G \text{ has a Hamiltonian path (that visits each node exactly once) from } s \text{ to } t \}
\[
\text{PATH} = \{ \langle G, s, t \rangle \mid G \text{ is a directed graph that has a path from } s \text{ to } t \} \\
\text{PATH} \in P
\]
G is a directed graph
Does it have a Hamiltonian path from s to t?
G is a directed graph
Does it have a Hamiltonian path from s to t?

Step 1: Mark s
Step 2: Repeat until no additional node is marked
1. Scan through all edges
2. Locate edge of form (marked, unmarked)
3. Mark unmarked node
Step 3: Check if t is marked.
\text{HAMPATH} = \{ \langle G, s, t \rangle \mid G \text{ has a Hamiltonian path from } s \text{ to } t \} \\

\textbf{Finding} a Hamiltonian path seems to be \textit{hard}  \\
Given a path, \textit{easy} to \textbf{check} that it is  
\begin{itemize}
  \item valid  
  \item Hamiltonian
\end{itemize}
The Class \textbf{NP}

class of languages that have \textit{polynomial time verifiers}
Def: (polynomial) verifier

Language $A$ with verifier $V$

$A = \{ w \mid \exists c \text{ such that } V \text{ accepts } \langle w, c \rangle \}$

$V$ is a decider that runs in $O(n^k)$ time

Certificate
\[ \text{HAMPATH} = \{ \langle G, s, t \rangle \mid G \text{ has a Hamiltonian path from } s \text{ to } t \} \]

\[ \text{HAMPATH} \in \text{NP} \]
HAMPATH = \{ \langle G, s, t \rangle \mid G \text{ has a Hamiltonian path from } s \text{ to } t \}\}

certificate is the Hamiltonian path from \( s \) to \( t \)
A language is in $\text{NP}$ iff it is decided by some nondeterministic polynomial time $\text{TM}$ has a polytime verifier.
polytime NTM for HAMPATH

N: On input \( \langle G, s, t \rangle \)
1. Nondeterministically generate a permutation of \( n \) nodes of \( G \)
2. Check if the first node is \( s \) and the last node is \( t \)
   2.1. If not, \textbf{reject} this branch
3. Check if every adjacent node pair is connected by an edge
   3.1. If not, \textbf{reject} this branch
3.2. If yes, \textbf{accept}
If a language is in $\text{NP}$ then it is decided by some nondeterministic polynomial time $\text{TM}$ has a polytime verifier.
If a language is in \textit{NP} then it is decided by some \textit{nondeterministic polynomial time} TM has a polytime verifier.
Given: $A \in \text{NP}$ has polytime verifier $V$

Construct: nondeterministic polytime TM $N$ that decides $A$

$V$ runs in $O(n^k)$

$N$: On input $w$

1. Nondeterministically pick a string $c$ in $\Sigma^n$
2. Run $V$ on $\langle w, c \rangle$.
   2.1. If it accepts, accept
   2.2. If it rejects, reject this branch
If a language is decided by some nondeterministic polynomial time TM, then it is in NP and has a polytime verifier.
Given: nondeterministic polytime TM $N$ that decides $A$

Construct: polytime verifier $V$ for $A$

Every branch of $N$ halts in $O(n^k)$

$V$: On input $\langle w, c \rangle$

1. Deterministically simulate the branch of $N$ indicated by $c$
   1.1. If it accepts, accept
   1.2. If it rejects, reject