Thm 7.27 [Cook-Levin]: SAT is in P \iff P = NP.
Def 7.29: Language $A$ is **polynomial-time reducible** to language $B$, written $A \leq_{P} B$, if a polynomial-time computable function $f: \Sigma^* \rightarrow \Sigma^*$ exists such that for every $w$,

$$w \in A \iff f(w) \in B$$

The function $f$ is called **polynomial-time reduction** of $A$ to $B$.

Thm 7.31: If $A \leq_{P} B$ and $B \in P$, then $A \in P$. 
Thm: HAMPATH is polynomial-time reducible to LONGESTPATH, where

\[
\text{LONGESTPATH} = \{ <G,s,t,k> \mid G \text{ is digraph and there exists a path from } s \text{ to } t \text{ of length } \geq k \}.
\]
Thm: HAMCYCLE is polynomial-time reducible to TSP, where

\[ \text{HAMCYCLE} = \{ <G> \mid G \text{ is a graph that contains a cycle through all vertices} \} \]

\[ \text{TSP} = \{ <G_w,k> \mid G_w \text{ is a complete weighted graph that contains a cycle through all vertices of length } \leq k \} \]
Thm 7.32: 3SAT is polynomial-time reducible to CLIQUE, where

\[ 3SAT = \{ <\phi> \mid \phi \text{ is a satisfiable 3-cnf formula} \}. \]
Def 7.34: A language $B$ is **NP-complete** if it satisfies both conditions:

- $B$ is in NP, and
- every $A$ in NP is polynomial-time reducible to $B$. 
Def 7.34: A language B is **NP-complete** if it satisfies both conditions:
- B is in NP, and
- every A in NP is polynomial-time reducible to B.

Thm 7.35: If B is NP-complete and B ∈ P, then P = NP.
Def 7.34: A language B is **NP-complete** if it satisfies both conditions:
- B is in NP, and
- every A in NP is polynomial-time reducible to B.

Thm 7.36: If B is NP-complete and $B \leq_P C$ for some $C \in \text{NP}$, then $C$ is NP-complete.
Def 7.34: A language B is **NP-complete** if it satisfies both conditions:
- B is in NP, and
- every A in NP is polynomial-time reducible to B.

Thm 7.37 [Cook-Levin]: SAT is NP-complete.

**Note:** a long list of known NP-complete problems.