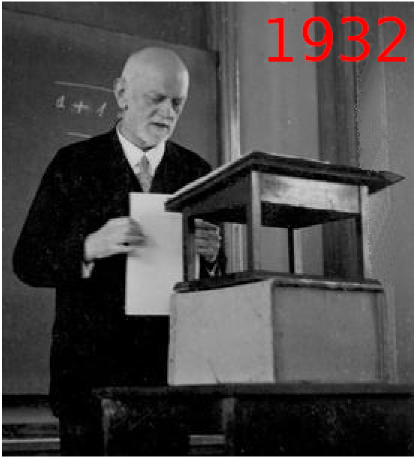
What is...P versus NP?

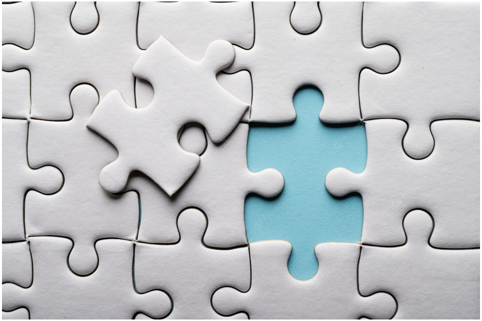
Or: Building a chair versus recognizing it

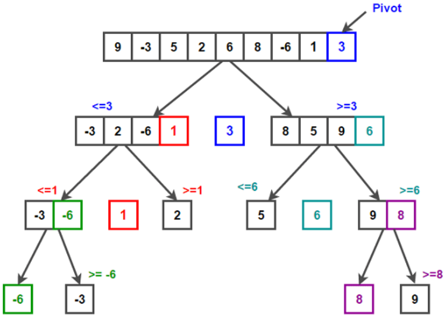
Hilbert's 1900 speech



- Hilbert Is there a "purely mechanical procedure" to "check statements"?
- The $P \stackrel{?}{=} NP$ problem is a modern refinement of Hilbert's question
- $P = NP$ $\leftrightsquigarrow$ fast way to address all questions with mechanically verifiable answers

Jigsaw versus sorting





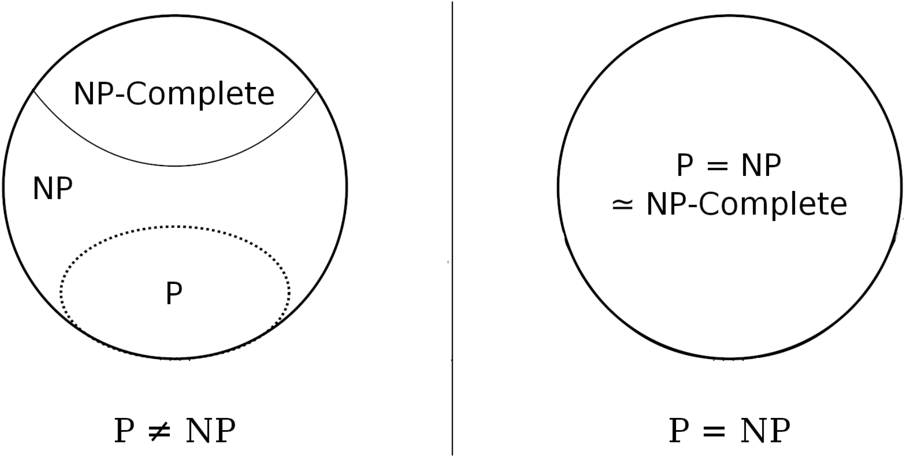
NP = easy to check that one has a solution & brute-force gives solutions + techinicalities

P = easy to find a solution, P⊂NP

P=NP "=" anyone who can recognize a chair can also build one

- It is not necessarily easy to find a solution to a jigsaw puzzle but it is easy to check whether one has a solution **Jigsaw in NP**
- Sorting can be done $\leq n^2$, and thus it is easy to find a solution **Sorting in P**

NP-complete

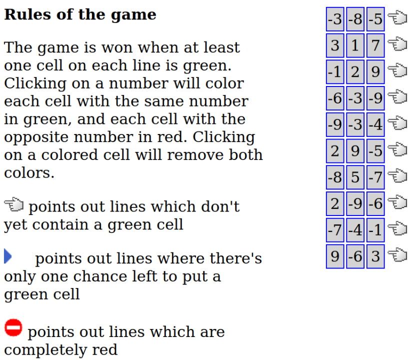


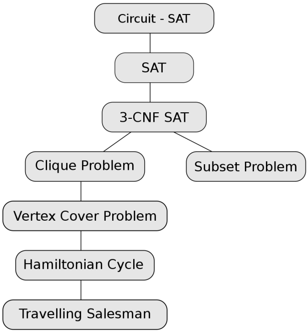
- NP-complete = problems that can be used to simulate every NP problem
- $(P \cap NP\text{-complete} \neq \emptyset) \Leftrightarrow (P = NP)$
- Task Find NP-complete problems

Enter, the theorem(s)

3SAT is NP-complete, and 3SAT can be solved in $O(1.3^n)$

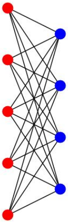
- SAT: the problem of determining if there exists an interpretation that satisfies a given Boolean formula; 3SAT = SAT with three inputs
- Most known NP-complete problems are reduced to some form of SAT



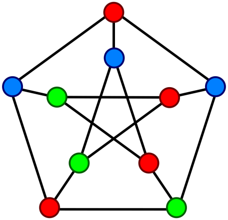


Why is $P \neq NP$ so hard to prove?

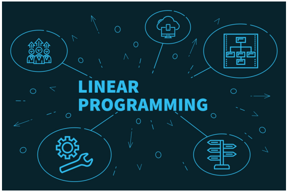
2-colorability $\in P$



3-colorability $\in NP$-complete



All LP problems are $\in P$



Matrix mult. is in $O(n^{2.4})$

$$\begin{bmatrix} a & b & c \\ d & e & f \end{bmatrix} \begin{bmatrix} u & v \\ w & x \\ y & z \end{bmatrix} = \begin{bmatrix} au + bw + cy & av + bx + cz \\ du + ew + fy & dv + ex + fz \end{bmatrix}$$

- 1) Variations of problems differ a lot in their complexity
- 2) There are amazingly clever ways to avoid brute-force approaches
- 3) Complexity classes are hard to determine precisely

Thank you for your attention!

I hope that was of some help.