TODAY: NP-completeness
- NP-hardness & -completeness
- 3SAT
  \[\rightarrow\] Super Mario Bros.
  \[\rightarrow\] 3-Dimensional Matching
  \[\rightarrow\] Subset Sum
  \[\rightarrow\] Partition
  \[\rightarrow\] Rectangle Packing
  \[\rightarrow\] 4-Partition
  \[\rightarrow\] Rectangle Packing
  \[\rightarrow\] Jigsaw puzzles

Recall: (from 6.006)
- \(P = \{\text{problems solvable in polynomial time}\}\)
  \(\substack{\rightarrow \text{size } n \\ \leftarrow n^{O(1)}}\)
- \(NP = \{\text{decision problems solvable in polynomial} \\
  \substack{\rightarrow \text{output is YES or NO} \\
  \rightarrow \text{nondeterministic time} \\
  \rightarrow \text{in } O(1) \text{ time can “guess” among polynomial number of choices} \\
  \text{if any guess leads to YES, then will make such a guess “lucky”}}\} \)
- can assume all guessing is done first
  \[ \Rightarrow \text{equivalent to polynomial-time verifier of polynomial-size certificates for YES answers} \]
- note asymmetry between YES & NO
- problem X is
  - \underline{NP-complete} if X \in NP & X is NP-hard
  - \underline{NP-hard} if every problem Y \in NP reduces to X
    - if P \neq NP then X \notin P \quad (NP \setminus P \rightarrow X)

- reduction from problem A to problem B =
  - polynomial-time algorithm converting A inputs into equivalent B inputs
  \[ \Downarrow \text{same YES/NO answer} \]
- if B \in P then A \in P \quad \Leftarrow A \rightarrow B \rightarrow \text{solve}
- if B \in NP then A \in NP
- if A is NP-hard then B is NP-hard

How to prove X is NP-complete:
\[ \text{(1) } X \in NP \text{ via nondeterministic algorithm} \]
\[ \quad \text{or certificate + verifier} \]
\[ \text{(2) reduce from known NP-complete problem Y to X} \]
\[ \Rightarrow \text{any } Z \in NP \rightarrow Y \rightarrow X \Rightarrow X \text{ is NP-hard} \]
\[ \quad \text{poly-time conversion from Y inputs to X inputs} \]
A if Y answer is YES then X answer is YES
B if X answer is YES then Y answer is YES
C if X answer is YES then Y answer is YES
3SAT: given Boolean formula of the form:

\[(x_1 \lor x_3 \lor \overline{x}_6) \land (\overline{x}_2 \lor x_3 \lor \overline{x}_7) \land \cdots\]

\[\text{clause} \]

\[\text{occurrences of variable } x_i\]

i.e. \text{formula} = \text{AND of clauses}
\text{clause} = \text{OR of 3 literals}
\text{literal} \in \{x_i, \overline{x}_i\}

is there a variable \rightarrow \{T, F\} assignment
such that formula = T (satisfying assignment)

- \text{NP-complete} \quad \text{[Cook 1971]}
- \in \text{NP}: guess \( x_1 \) is T or F
  \quad \text{guess } x_2 \text{ is T or F}
  \quad \text{check formula} \quad \in O(\#\text{variables})
  \quad \text{nondeterministic}
- \text{NP-hard: intuition}
  \quad \text{convert algorithm into a circuit}
  \quad \text{convert circuit into a formula}
  \quad \text{convert formula into 3CNF (as above)}
Super Mario Bros. is NP-hard

- generalized to arbitrary screen size \((n \times n)\)
- reduction from 3SAT:

\[
\begin{align*}
&x_1 &\quad &T &\quad &F \\
&x_2 &\quad &T &\quad &F \\
&x_3 &\quad &T &\quad &F \\
\end{align*}
\]

Also mushroom check

variable:

clause:

cross-over:

For many more cool examples, check out 6.890: “Fun with Hardness”

http://courses.csail.mit.edu/6.890/fall14/
3-Dimensional Matching: (3DM)

given disjoint sets $X, Y, Z$ each of $n$ elements, &
triples $T \subseteq X \times Y \times Z$, is there a subset $S \subseteq T$ such that
each element $e \in X \cup Y \cup Z$ is in exactly one set $s \in S$?

- $\in \text{NP}$: guess which triples $e \in S$ – $O(T)$ nondet.
  check for exact coverage – $O(T)$

- $\text{NP-hard by reduction from 3SAT}$:
  [Garey & Johnson 1979 book] → # occurrences of $x$ or $\bar{x}$
  - variable $x \rightarrow 2n_x$ chain:
    - exactly 2 solutions
    - either $x$'s or $\bar{x}$'s left
  - clause $x \lor y \lor z \rightarrow$
    - solvable if $x$ or $y$ or $z$'s left
  - garbage collection: $\odot$ all $x_i$ & $\bar{x}_i$'s
    - shared (per repeat)

repeated $\sum_{i=1}^{n_x} n_x$ – # clauses times

- can cover exactly all unused $x_i$'s & $\bar{x}_i$'s

- satisfying assignment $\rightarrow$ 3DM
  $(x = T \rightarrow$ leave $x$; $x = F \rightarrow$ leave $\bar{x}$; satisfy clauses; 
  cover remaining with garbage collector)$

- 3DM $\rightarrow$ satisfying assignment
  $(x$ left $\rightarrow x = T$; $\bar{x}$ left $\rightarrow x = F$; satisfy clauses)$
**Subset Sum:** given \( n \) integers \( A = \{a_1, a_2, \ldots, a_n\} \) & a target sum \( t \), is there a subset \( S \subseteq A \) such that \( \sum S = \sum_{a \in S} a = t \)?

- \( \in \text{NP} \): guess \( S \)
- **polynomial algorithm via DP** (like knapsack)
- weakly \( \text{NP-hard} \) by reduction from 3DM
  - hard when numbers exponential in \( n \)
    - but still only polynomial number of bits
  - view numbers in base \( b = 1 + \max_i n_{x_i} \)
    - never overflow/carry
  - triple \((x_i, x_j, x_k)\) → \(000100100001000\)
    - \( t = 11 \cdots 1 = \sum_i b^i \)
**Partition**: given $n$ positive integers $A = a_1, a_2, \ldots, a_n, 3\alpha$, is there a subset $S \subseteq A$ with $\sum S = \sum (A \setminus S) = \frac{1}{2} \sum A$?

- special case of Subset Sum ($t = \frac{1}{2} \sum A$)
  $\Rightarrow \in \text{NP} \& \text{pseudopolynomial algorithm}$
- weakly NP-hard by reduction from Subset Sum
  - let $\sigma = \sum_{i=1}^{n} a_i$
  - add $a_{n+1} = \sigma + t$ & $a_{n+2} = 2\sigma - t$
    $\Rightarrow$ exactly one is $\in S$ (else $3\sigma$ vs. $\sigma$)
  $\Rightarrow$ partition must add $t$ to $a_{n+1}$ & add $\sigma - t$ to $a_n$

**Rectangle packing**: given $n$ rectangles $R_1, R_2, \ldots, R_n$ & target rectangle $T$ of area $\sum \text{area}(R_i)$ can you pack $R_i$'s into $T$ without overlap?

- $\in \text{NP}$ because areas match
  $\Rightarrow$ can only rotate by int. $\times 90^\circ$
  $\Rightarrow$ can guess rotation & integer translation
- weakly NP-hard by reduction from Partition:
  - $a_i \rightarrow 1 \times 3a_i$ rectangle $R_i$
  - $t \rightarrow 2 \times 3t$ rectangle $T$
  - $3 > 2 \Rightarrow$ can't rotate $90^\circ$
  $\Rightarrow$ packing must find partition
4-Partition: given $n$ integers $A = \{a_1, a_2, \ldots, a_n\}$, is there a partition into $n/4$ subsets of $4$ each with the same sum $t = \sum A / (n/4)$?

- $\in$ NP: guess $A \rightarrow$ subset mapping
- strongly $\text{NP}$-hard by reduction from 3DM $[G&J]$

$\Rightarrow$ NP-hard even when number values polynomial in $n$

- write numbers in base $r = 100 \cdot \sum (x_i \vee y_j \vee z_k)$
- element $x_i \in X$ $\rightarrow$ (10, 0, 0, 1, 1) $\times (n_{x_i} - 1)$ copies
- element $y_j \in Y$ $\rightarrow$ (10, 0, 0, 2) $\times (n_{y_j} - 1)$ copies
- element $z_k \in Z$ $\rightarrow$ (10, 0, 0, 4) $\times (n_{z_k} - 1)$ copies

- target sum $t = (40, 0, 0, 0, 15)$ $= 40r^4 + 15$
- no carries ($r$ large enough)
- $\mod r \Rightarrow$ use one $x_i$, one $y_j$, one $z_k$, one triple
- $[\frac{\Sigma}{r}] \mod r \Rightarrow z_k$ & triple match
- $[\frac{\Sigma}{r^2}] \mod r \Rightarrow y_j$ & triple match
- $[\frac{\Sigma}{r^3}] \mod r \Rightarrow x_i$ & triple match
- $[\frac{\Sigma}{r^4}] \mod r \Rightarrow 4 \cdot 10$ $\Rightarrow$ chosen triple $\in S$

or $11 + 11 + 8 + 10 \Rightarrow$ unused triple $\notin S$

- primary (10) form of $x_i$ (or $y_j$ or $z_k$) must appear in exactly one chosen triple (and elements of triple must all match)
Rectangle packing:
- strongly NP-hard by reduction from 4-Partition
  - \( a_i \rightarrow 1 \times n a_i \) rectangle \( R_i \)
  - \( t \rightarrow \frac{n}{3} \times n t \) rectangle \( T \)

Jigsaw puzzles:
- model: square tiles (no pattern)
  each side tab, pocket, or boundary tabs & pockets must have matching shape target rectangular shape
- NP-hard by reduction from 4-Partition:
  (similar to reduction to Rectangle Packing)
  - \( a_i \rightarrow \)
  - \( t \rightarrow \) frame: