EASY & HARD PROBLEMS

Tuesday, November 27, 2018 5:09 PM

WANT: a formal definition of hard / easy problems

Intuitively, a problem is "easy" if it requires "little" time to solve.

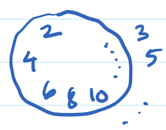
Def: A DECISION PROBLEM is a problem whose OUTPUT is either TRUE(1) or FALSE(0)

Note: A decision problem can be associated with a set A, whose elements are inputs whose outputs are TRUE.

Ex: EVEN is a decision problem

INPUT: a positive integer n

OUTPUT: TRUE if n is even and FALSE if not.



Ex: SORTING is NOT a decision problem.

INPUT: n positive integers $a_1, a_2, \dots, a_n$

OUTPUT: a rearrangement π of $a_1, \dots, a_n$ such that

$$a_{\pi(1)} \le a_{\pi(2)} \le \dots \le a_{\pi(n)}$$

IS_SORTED

INPUT: n positive integers $a_1, a_2, \dots, a_n$

OUTPUT: TRUE if $a_1 \le a_2 \le \dots \le a_n$ and FALSE otherwise

Def: A decision problem is EASY if there is a POLY-TIME algorithm to FIND the output.

EVEN is easy because there is an $O(n)$ alg: TRUE if RIGHTMOST bit is 0

IS_SORTED is easy because there is an $O(n)$ alg:

IS_MEDIAN:

INPUT: a list of n integers $a_1, \dots, a_n$; an integer m

OUTPUT: TRUE if m is the (left) median of $a_1, \dots, a_n$

ARE_RELATIVELY_PRIME

INPUT: 2 positive integers a, b

OUTPUT: TRUE if a, b have no common factors > 1.

Ex: 2, 3 TRUE

4, 6 FALSE

Easy because TRUE if $\gcd(a,b) = 1$ $O(\log(\max(a,b)))$

Def: The class of decision problems with a poly-time alg to FIND the output is called P

PARTITION : NOT KNOWN TO BE IN P

INPUT: n positive integers $s_1, s_2, \dots, s_n$

OUTPUT: true if $\{s_1, s_2, \dots, s_n\} = A \cup B$ for some sets A, B such that $A \cap B = \emptyset$ and $\text{sum}(A) = \text{sum}(B)$

Ex: INPUT: 3, 1, 20, 7, 6, 1, 4 $\quad 1+20 = 3+7+6+1+4$

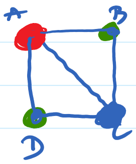
OUTPUT: TRUE

K
3-COLOR NOT KNOWN TO BE IN P

INPUT: a graph (V, E)

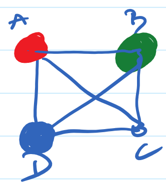
OUTPUT: TRUE if there is a function $c: V \to \{R, G, B\}$ such that if $\{a, b\} \in E$, then $c(a) \neq c(b)$

Ex:



TRUE

Ex:



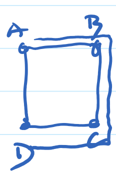
FALSE Brute-force: $3 \times 3 \times 3 \times \cdots \times 3 = 3^n$

HAMILTONIAN PATH / CYCLE

INPUT: a graph (V, E)

OUTPUT: TRUE if there is a path (cycle) (tour) that goes through each vertex once

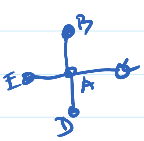
Ex:



TRUE

A B E C D

□ □ □ □ □ $\quad n!$



FALSE

LONGEST PATH

INPUT: graph (V, E); a positive integer B

OUTPUT: TRUE if there is a SIMPLE PATH that has at least B edges.

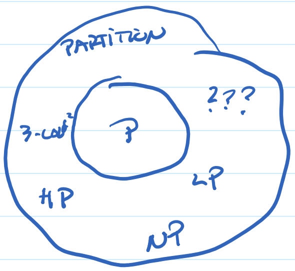
$a^2 + b^2 = c^2$

$a^3 + b^3 = c^3$

$a^4 + b^4 = c^4$ ⋮

$a^n + b^n = c^n$ Fermat

Def: A decision problem is in NP if there is a poly-time alg to VERIFY the solution.

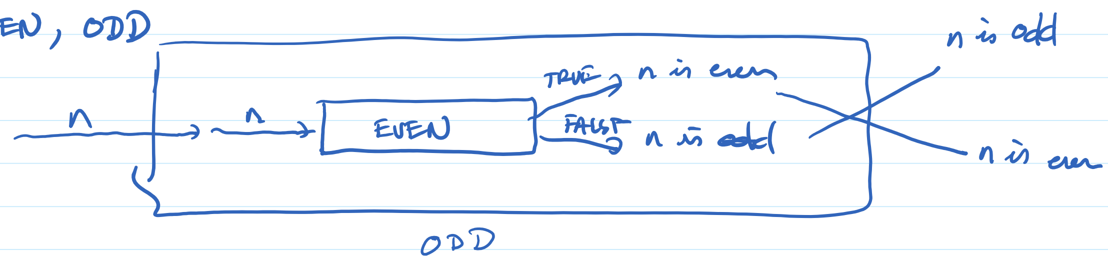


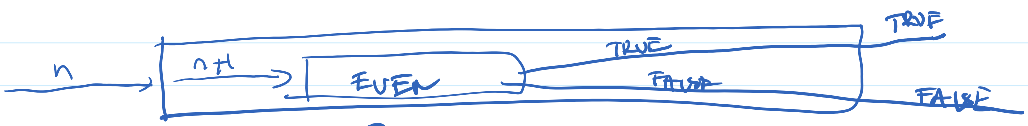
$P \overset{?}{=} NP$

Note: 3-COLOR, PARTITION, HP, LP, ... are considered HARD problems.
We can use them to show that other problems are LIKELY TO BE HARD by reduction.

REDUCTION:

EVEN, ODD





ODD $\leq$ EVEN

n is odd iff n+1 is even

Def: Let A, B be decision problems (or equivalently, sets).
We say A reduces to B and write $A \leq B$ if there is a POLY-TIME alg f such that

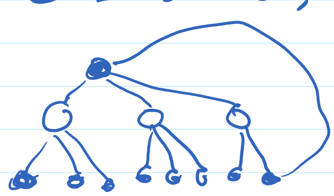
$$a \in A \iff f(a) \in B$$

Ex: ODD $\leq$ EVEN because
$n \in \text{ODD} \iff f(n) = n+1 \in \text{EVEN}$

Def: A decision problem H is NP-hard if
$C \leq H$ where C = 3-COLOR, HP, LP, PARTITION

Ex:

3-COLOR is hard
2-COLOR is easy
4-COLOR



CLAIM: 4-COLOR is NP-hard

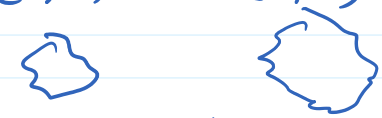
Proof:

We need to reduce a known NP-hard problem to 4-COLOR

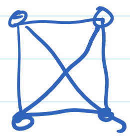
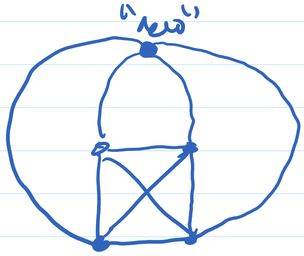
$$3\text{-COLOR} \leq 4\text{-COLOR}$$

n is odd? n+1 even?

We need to find a poly-time function $f(V,E) \longrightarrow (V', E')$



3-colorable $\Longleftrightarrow$ 4-colorable

G   G'

3-colorable? 4-colorable

```
f(V,E)
{ E' := E
  V' := V ∪ {new}
  for each v ∈ V
     E' += {new, v}
} return (V', E')
```

poly transformation

Claim: G is 3-colorable $\Longleftrightarrow$ G' is 4-colorable

Proof:

Suppose G is 3-colorable. Then G' can be 4-colored by using the 4th on new

Conversely suppose G' is 4-colorable. We claim that color of "new" is unique since $\{new, v\} \in E'$ for any $v \in V$.

In otherwords, the vertices in V are colored using only 3 colors. So G is 3-colorable.