P and NP

P = set of decision problems solvable in polynomial time

$1 M question: $P \stackrel{?}{=} NP$

NP = set of decision problems with polynomial-time verification algorithms

P = set of decision problems solvable in polynomial time

\$1 M question: $P \stackrel{?}{=} NP$ $(NP \stackrel{?}{\subseteq} P)$

NP = set of decision problems with polynomial-time verification algorithms

$P \subseteq NP$: Let $X \in P$. Then there exists poly-time A that solves X. (def. P)

Define poly-time verifier for X: $\underline{\text{X-VERIFY}(x,y)}$
return $A(x)$

[Need: $X(x) = YES \rightarrow \exists y$ s.t. size(y) is poly in size(x) and X-VERIFY(x,y) = YES]

If $X(x) = YES$ then $\text{X-VERIFY}(x,y) \underset{\text{code}}{=} A(x) \underset{A \text{ solves } X \text{ so } A(x) = X(x)}{=} YES$ for all y

[Need: $X(x) = NO \rightarrow \forall y$, X-VERIFY(x,y) = NO for all y]

If $X(x) = NO$ then $\text{X-VERIFY}(x,y) = A(x) = NO$ for all y

NP-complete Problems

NP-complete: Problem X is NP-complete if 1) $X \in NP$

2) $Y \leq_p X$ for all $Y \in NP$] X is NP-hard

for all c, no A that solves X has worst case $O(n^c)$

To show $P \neq NP$, it suffices to prove that $X \notin P$ for some $X \in NP$.

NP-complete: Problem X is NP-complete if 1) $X \in NP$
2) $Y \le_p X$ for all $Y \in NP$] X is NP-hard

for all c, no A that solves X has worst case $O(n^c)$

To show $P \neq NP$, it suffices to prove that $X \notin P$ for some $X \in NP$.

To show $P = NP$, it suffices to find a polynomial time algorithm for some NP-complete X

Suppose X is NP-complete and A solves X in polynomial time

$NP \subseteq P$

Let $Y \in NP$

Then $Y \le_p X$ (def NP-complete)

So some B solves Y using poly-time + poly calls to alg for X (def $\le_p$)

Then $Y \in P$ (use A to solve X in B : $\underbrace{\text{poly}}_{\text{poly time in } B} + \underbrace{\text{poly}}_{\text{poly calls to } A} \cdot \underbrace{\text{poly}}_{\text{poly time in } A} = \text{poly}$)

Most computer scientists <u>think</u> $P \neq NP$

NP-complete: Problem X is NP-complete if 1) $X \in NP$
2) $Y \le_p X$ for all $Y \in NP$] X is NP-hard

for all c, no A that solves X has worst case $O(n^c)$

To show $P \neq NP$, it suffices to prove that $X \notin P$ for some $X \in NP$.

To show $P = NP$, it suffices to find a polynomial time algorithm for some NP-complete X

Suppose X is NP-complete and A solves X in polynomial time

$NP \subseteq P$

- Let $Y \in NP$
- Then $Y \le_p X$ (def NP-complete)
- So some B solves Y using poly-time + poly calls to alg for X (def $\le_p$)
- Then $Y \in P$ (use A to solve X in B : poly + poly · poly = poly)

poly time in B, poly calls to A, poly time in A

Most computer scientists <u>think</u> $P \neq NP$

NP-complete Problems

n is superprime if it is a positive integer with no divisors
there are no superprimes

NP-complete: Problem X is NP-complete if 1) $X \in NP$
2) $Y \leq_p X$ for all $Y \in NP$] X is NP-hard

To show X is NP-complete : 1) show $X \in NP$
2) show $Y \leq_p X$ for some NP-complete Y

(def NP-complete) (transitivity)

Let $Z \in NP$. Then $Z \leq_p Y$ and so $Z \leq_p X$

So $Z \leq_p X$ for all $Z \in NP$ and hence X is NP-complete

NP-complete Problems

n is superprime if it is a positive integer with no divisors

there are no superprimes

NP-complete: Problem X is NP-complete if 1) $X \in NP$

2) $Y \leq_p X$ for all $Y \in NP$] X is NP-hard

To show X is NP-complete: 1) show $X \in NP$

2) show $Y \leq_p X$ for some NP-complete Y

(def NP-complete) (transitivity)

Let $Z \in NP$. Then $Z \leq_p Y$ and so $Z \leq_p X$

So $Z \leq_p X$ for all $Z \in NP$ and hence X is NP-complete

If $Z \leq_p Y$ and $Y \leq_p X$ then $Z \leq_p X$

Then there is a B that solves Z using poly time + poly calls to alg for Y

And there is an A that solves Y using poly time + poly calls to alg for X

So there is an alg for Z using poly time + poly calls to X (use A in B)

poly time + poly · (poly time + poly calls to X) = poly + poly · poly time + poly · poly calls to X

= poly time + poly calls to X

n is superprime if it is a positive integer with no divisors
there are no superprimes

NP-complete: Problem X is NP-complete if 1) $X \in NP$
2) $Y \le_p X$ for all $Y \in NP$] X is NP-hard

To show X is NP-complete: 1) show $X \in NP$
2) show $Y \le_p X$ for some NP-complete Y

(def NP-complete) (transitivity)
Let $Z \in NP$. Then $Z \le_p Y$ and so $Z \le_p X$

So $Z \le_p X$ for all $Z \in NP$ and hence X is NP-complete

If $Z \le_p Y$ and $Y \le_p X$ then $Z \le_p X$

Then there is a B that solves Z using poly time + poly calls to alg for Y

And there is an A that solves Y using poly time + poly calls to alg for X

So there is an alg for Z using poly time + poly calls to X (use A in B)

poly time + poly · (poly time + poly calls to X) = poly + poly · poly time + poly · poly calls to X
= poly time + poly calls to X

NP-complete Problems

n is superprime if it is a positive integer with no divisors
there are no superprimes

NP-complete: Problem X is NP-complete if 1) $X \in NP$
2) $Y \leq_p X$ for all $Y \in NP$] X is NP-hard

To show X is NP-complete: 1) show $X \in NP$
2) show $Y \leq_p X$ for some NP-complete Y

(def NP-complete) (transitivity)
Let $Z \in NP$. Then $Z \leq_p Y$ and so $Z \leq_p X$

So $Z \leq_p X$ for all $Z \in NP$ and hence X is NP-complete

Hamiltonian Cycle $\leq_p$ Travelling Salesperson

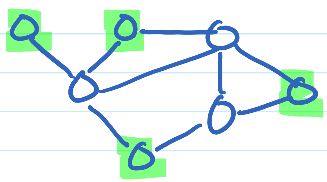
HC is NP-complete (trust me)

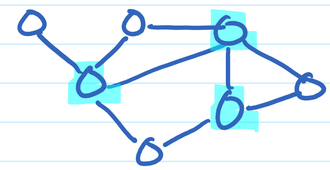
TSP $\in NP$ (previous video)

So Travelling Salesperson is NP-complete

Independent Set is NP-complete

VERTEX-COVER: Given undirected G and k, is there a $\overbrace{\text{vertex cover}}^{\text{subset of vertices } C \text{ s.t. all edges have } \geq 1 \text{ endpoint in } C}$ C with $|C| \leq k$?





INDEPENDENT SET: Given undirected G and k, is there an $\overbrace{\text{independent set}}^{\text{subset of verts s.t. } u,v \in S \rightarrow \text{no edge } (u,v)}$ S with $|S| \geq k$?

VERTEX-COVER is NP-complete (trust me)

INDEPENDENT SET is NP-complete

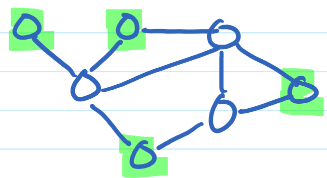
IS $\in$ NP:

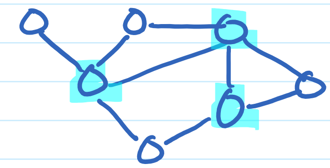
VC $\leq_p$ IS:

Independent Set is NP-complete

subset of vertices C s.t. all edges have ≥ 1 endpoint in C

VERTEX-COVER: Given undirected G and k, is there a vertex cover C with $|C| \leq k$?





vertex cover of size 3

subset of verts s.t. $u, v \in S \rightarrow$ no edge (u,v)

INDEPENDENT SET: Given undirected G and k, is there an independent set S with $|S| \geq k$?

VERTEX-COVER is NP-complete (trust me)

poly in size of G

INDEPENDENT SET is NP-complete

G has ind. set S w/ $|S| \geq k \rightarrow$ IS-VERIFY$(G,k,S) = Y$

G has no such ind. set $\rightarrow$ IS-VERIFY$(G,k,S) = N$ forall S

IS $\in$ NP:

IS-VERIFY(G, k, S)
- if $S \subseteq$ vertices in G — $O(n)$
  - for $u, v \in S$ — $O(n^2)$ iterations
    - if (u,v) is an edge in G then return NO — $O(n)$ per check
- return YES

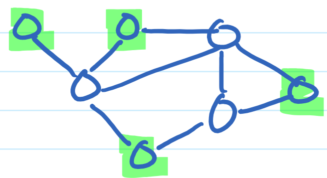
$O(n^3)$ total (polynomial)

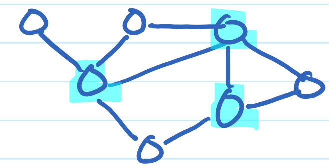
VC $\leq_p$ IS:

Independent Set is NP-complete

subset of vertices C s.t. all edges have ≥ 1 endpoint in C

VERTEX-COVER: Given undirected G and k, is there a vertex cover C with $|C| \leq k$?





vertex cover of size 3

subset of verts s.t. $u,v \in S \to$ no edge (u,v)

INDEPENDENT SET: Given undirected G and k, is there an independent set S with $|S| \geq k$?

VERTEX-COVER is NP-complete (trust me)

INDEPENDENT SET is NP-complete

poly in size of G

G has ind. set S w/ $|S| \geq k \to$ IS-VERIFY$(G,k,S) = Y$

G has no such ind. set $\to$ IS-VERIFY$(G,k,S) = N$ forall S

IS $\in$ NP:

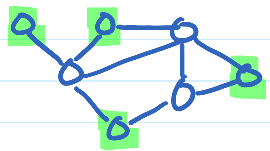
IS-VERIFY (G, k, S)

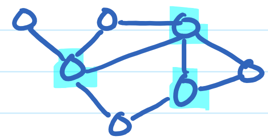
```
if  S ⊆ vertices in G                                  O(n)
   for u,v ∈ S                                         O(n²) iterations
      if (u,v) is an edge in G then return NO          O(n) per check
   return YES                                          O(n³) total (polynomial)
return NO
```

VC $\leq_p$ IS:

VC <= IS

largest independent set





smallest vertex cover

$VC \le_p IS$:

$\underline{VC(G, k)}$
return $IS(G, n-k)$

G has vertex cover of size $\le k$ $\iff$ G has independent set of size $\ge n-k$

$\Rightarrow$: Suppose G has vertex cover C with $|C| \le k$.

Let $S = V - C$. Then $|S| \ge n - k$. $(|S| = n - |C|;\ -|C| \ge -k)$

Also, S is an independent set:

Suppose $u, v \in S$ but $(u,v) \in E$

$u, v \notin C$ (choice of S)

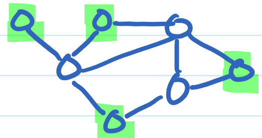
(u,v) is not covered by E (neither endpoint is in C)

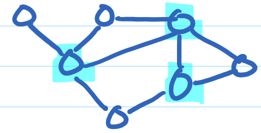
C is not a vertex cover $\Rightarrow\Leftarrow$ (doesn't cover (u,v))

$\therefore \forall u,v\ \ u, v \in S \rightarrow$ no edge (u,v)

VC <= IS

largest independent set





smallest vertex cover

$VC \leq_p IS$: $\underline{VC(G,k)}$
return $IS(G, n-k)$

G has vertex cover of size $\geq k$ $\Longleftrightarrow$ G has independent set of size $\geq n-k$

$\Rightarrow$: Suppose G has vertex cover C with $|C| \leq k$.

Let $S = V - C$. Then $|S| \geq n-k$.   ($|S| = n - |C|$; $-|C| \geq -k$)

Also, S is an independent set:

Suppose $u, v \in S$ but $(u,v) \in E$

$u, v \notin C$ (choice of S)

(u,v) is not covered by E   (neither endpoint is in C)

C is not a vertex cover $\Rightarrow\Leftarrow$ (doesn't cover (u,v))

$\therefore \forall u,v \;\; u, v \in S \rightarrow$ no edge (u,v)

$\Leftarrow$: Suppose G has independent set S with $|S| \geq n-k$

Let $C = V - S$ then $|C| \leq k$.   ($|C| = n - |S|$; $-|S| \leq -n+k$)

Also, C is a vertex cover:

Suppose $(u,v) \in E$ but $u \notin C$ and $v \notin C$

Then $u, v \in S$ (choice of C)

S is not an independent set $\Rightarrow\Leftarrow$ (has adjacent verts u,v)

$\therefore \forall u,v \;\; (u,v) \in E \rightarrow u \in C$ or $v \in C$