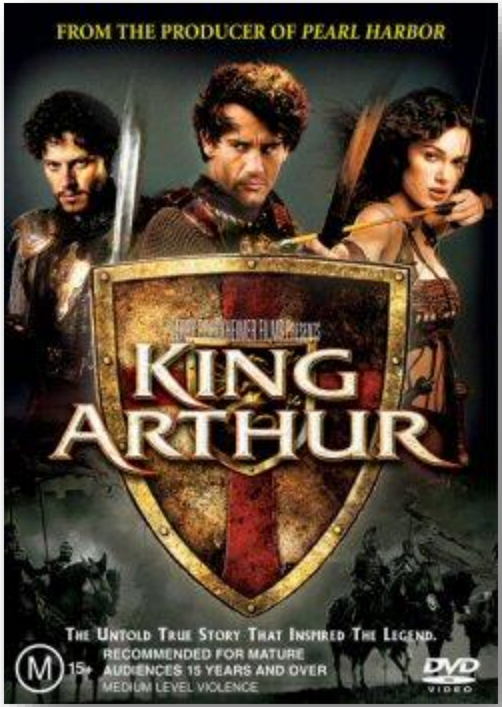
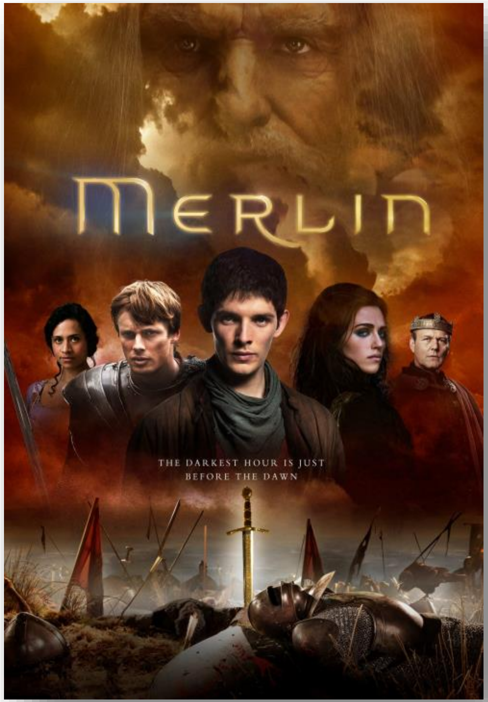
CMU 15-251

Interactive proofs

Teachers:
Anil Ada
Ariel Procaccia (this time)

Our protagonists





NP, REVISITED

Prover wants to convince Verifier that $x \in A$; cooks up a proof and sends it to Verifier

Verifier checks in polynomial time that the proof is legit

NP, REVISITED

- This strategy works, by definition, for any language in NP
- But what about languages like ¬3COL?
- We will relax the assumptions:
  - Make the protocol interactive
  - Make the verifier probabilistic

Coke vs. Pepsi

IP for Coke vs. Pepsi

Verifier fills a cup with a random choice of Coke/Pepsi, passes it to Prover

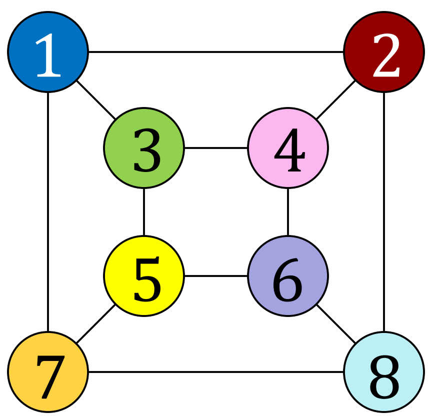
Prover tastes the contents of the cup, says whether it is Coke or Pepsi

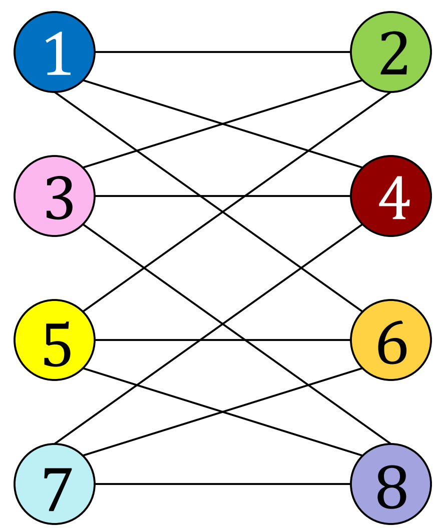
If Prover is correct Verifier accepts, otherwise verifier rejects

Graph nonisomorphism

- Two graphs $G_0 = (V_0, E_0)$ and $G_1 = (V_1, E_1)$ are isomorphic iff there is a permutation $\pi: V_0 \to V_1$ such that

$$(u, v) \in E_0 \Leftrightarrow \big(\pi(u), \pi(v)\big) \in E_1$$





Graph nonisomorphism

- Observation: If G_0 is isomorphic to G_1, and G_1 is isomorphic to G_2, then G_0 is isomorphic to G_2
- Graph isomorphism is clearly in **NP** (unknown if it's **NP**-complete)
- But how do we prove that two graphs are not isomorphic?
- We will give an interactive protocol!

IP for Graph nonisomorphism

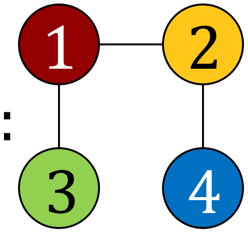
Verifier chooses $b \in \{0,1\}$ and permutation π at random, and sends $\pi(G_b)$ to prover

Prover sends a bit b'

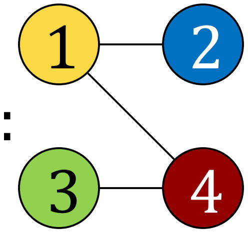
If $b = b'$ verifier accepts, otherwise verifier rejects

IP for Graph nonisomorphism

G_0:



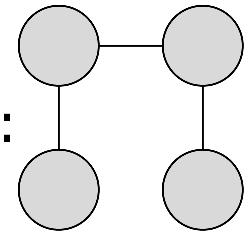
$\pi(G_0)$:



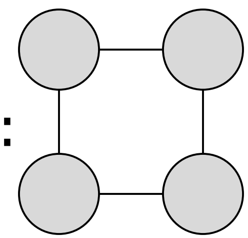
0

Accept

G_0:



G_1:



IP for Graph nonisomorphism

1. Verifier chooses $b \in \{0,1\}$ and permutation π at random, and sends $\pi(G_b)$ to prover
2. Prover sends a bit b'
3. If $b = b'$ verifier accepts, otherwise it rejects

- Poll 2: Probability that prover will get the verifier to accept, when the graphs are nonisomorphic and isomorphic, respectively?
  1. 1 and $1/n!$
  2. 1 and $1/2$
  3. $1/2$ and $1/n!$
  4. $1/2$ and $1/2$

Interactive proofs

- An interactive proof system for problem L is a protocol between a computationally unbounded prover P and a probabilistic polynomial-time verifier V such that:
  - Completeness:
    $\forall x \in L, \Pr[(V \leftrightarrow P)(x) \text{ accepts}] = 1$
  - Soundness:
    $\forall x \notin L, \forall P', \Pr[(V \leftrightarrow P')(x) \text{ accepts}] \leq 1/2$

Interactive proofs

- Graph Nonisomorphism has an interactive proof system

But being fooled with probability ½ is still pretty bad! What can we do about it?

Interactive proofs

- Poll 1: What is the relation between **NP** and **IP**?
  1. $\mathbf{NP} \subset \mathbf{IP}$
  2. $\mathbf{IP} \subset \mathbf{NP}$
  3. $\mathbf{IP} = \mathbf{NP}$
  4. They are incomparable

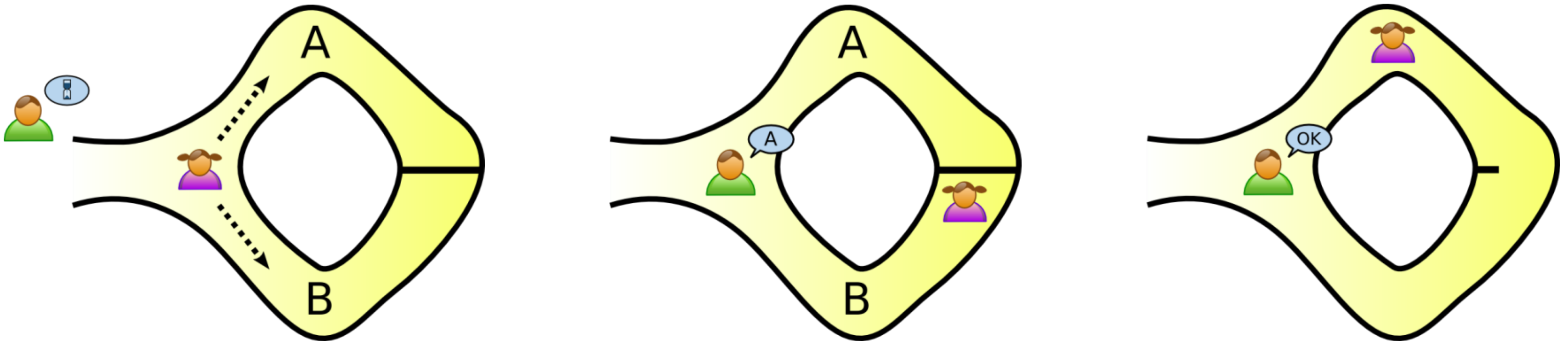
Zero knowledge proofs

- Graph isomorphism clearly has an interactive proof: Prover sends a solution, verifier checks it
- But can the prover convince the verifier that there is a solution without revealing the solution?
- This is called a zero knowledge proof

Why do we need ZKPs?

Merlin, prove that you are who you say you are!

Intuition for ZKPs



ZKP for graph isomorphism

Prover chooses $b \in \{0,1\}$ and permutation π at random, and sends $H = \pi(G_b)$ to Verifier

Verifier sends a random bit b' to Prover

Prover picks a permutation π' and sends it to Verifier

Verifier accepts iff $H = \pi'(G_{b'})$

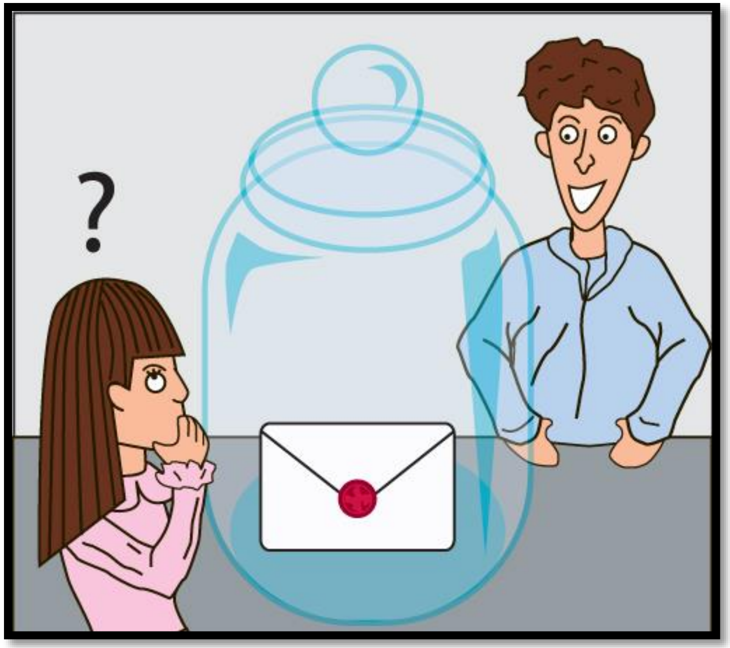
ZKP for graph isomorphism

1. Prover chooses $b \in \{0,1\}$ and permutation π at random, and sends $H = \pi(G_b)$ to Verifier
2. Verifier sends a random bit b' to Prover
3. Prover picks a permutation π' and sends it to Verifier
4. Verifier accepts iff $H = \pi'(G_{b'})$

- This is an interactive proof protocol:
  - It is complete (why?)
  - It is sound (why?)
- The verifier learns nothing about the solution!

ZKP FOR 3-COLORING

- Next, we want to design an zero knowledge proof system for 3-COLORING
- We will rely on a cryptographic construction known as bit commitment
- Prover can put bits in envelopes and send them to Verifier; Verifier can only open an envelope if Prover tells him how to do it



ZKP for 3-coloring

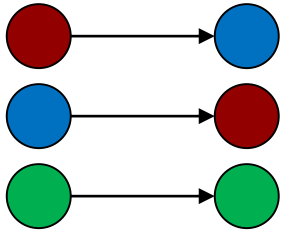
Prover selects random permutation π of $\{R, G, B\}$, commits to $\pi(\gamma(v))$ for all $v \in V$ and sends

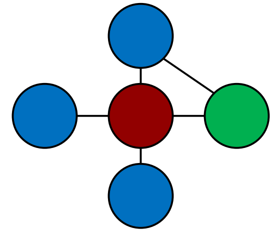
Verifier selects an edge $(u, v) \in E$ uniformly at random and sends

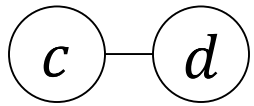
Prover reveals $a = \pi(\gamma(u))$ and $b = \pi(\gamma(v))$

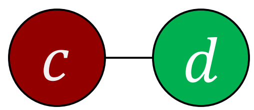
Verifier accepts iff $a \neq b$

ZKP for 3-coloring

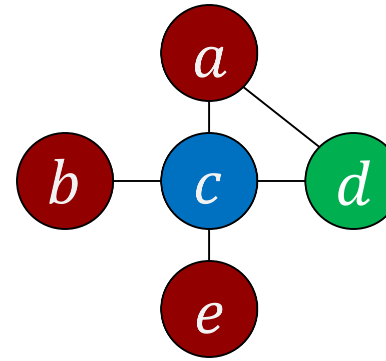








Accept



$\gamma(G)$

ZKP for 3-coloring

1. Prover selects random permutation π of $\{R, G, B\}$, commits to $\pi(\gamma(v))$ for all $v \in V$ and sends
2. Verifier selects an edge $(u, v) \in E$ uniformly at random and sends
3. Prover reveals $a = \pi(\gamma(u))$ and $b = \pi(\gamma(v))$
4. Verifier accepts iff $a \neq b$

- Poll 3: If G has no 3-coloring, what is the worst-case prob. Prover can convince Verifier?

1. $1 - \frac{1}{2}$
2. $1 - \frac{1}{n!}$
3. $1 - \frac{1}{3!}$
4. $1 - \frac{1}{|E|}$

ZKP for 3-coloring

1. Prover selects random permutation π of $\{R, G, B\}$, commits to $\pi(\gamma(v))$ for all $v \in V$ and sends
2. Verifier selects an edge $(u, v) \in E$ uniformly at random and sends
3. Prover reveals $a = \pi(\gamma(u))$ and $b = \pi(\gamma(v))$
4. Verifier accepts iff $a \neq b$

- To get soundness, we must repeat the protocol
- Intuition for zero knowledge: Prover just reveals a pair of distinct random colors!

More generally

- This "proves" that every problem in NP can be proved in zero-knowledge
- Theorem (Ben-Or et al., 1990): Every problem in IP can be proved in zero-knowledge

What you need to know

Nothing