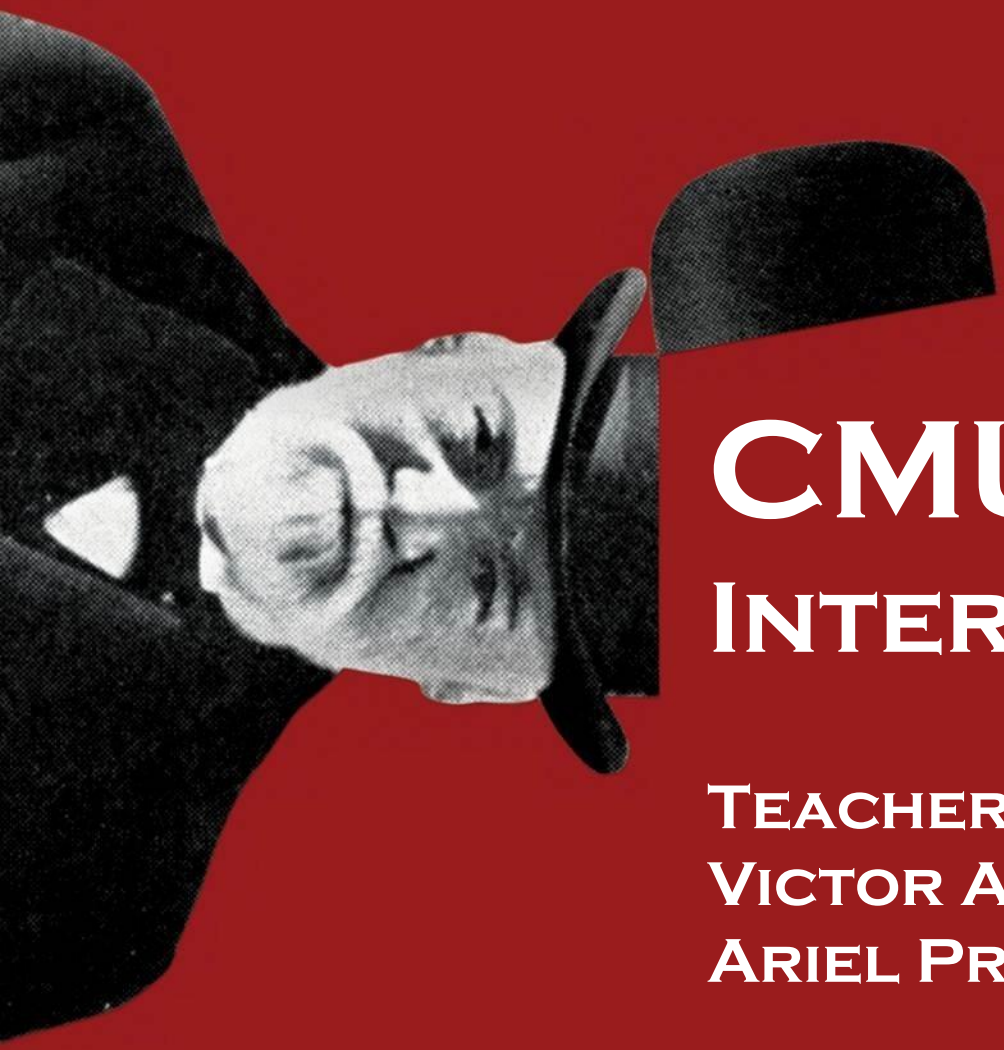


CMU 15-251

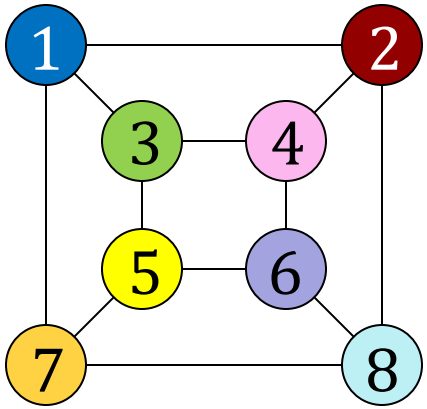
Interactive proofs

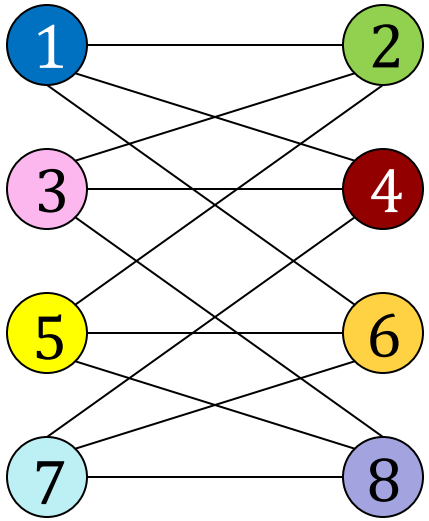
Teachers:
Victor Adamchik
Ariel Procaccia (this time)

Graph nonisomorphism

- Reminder: 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

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

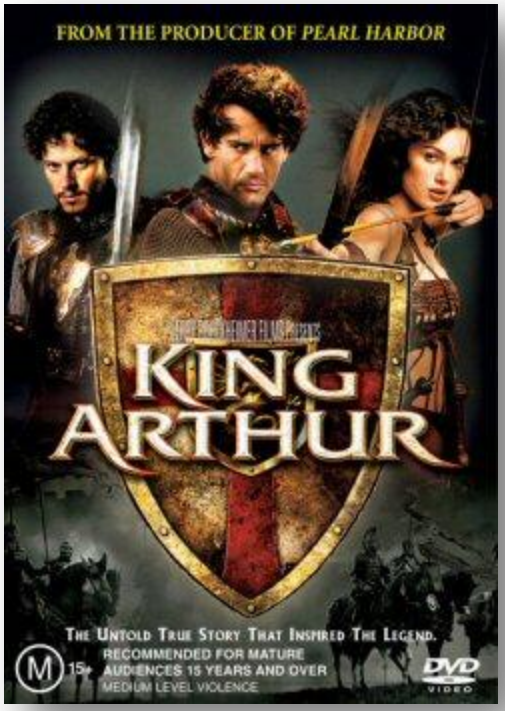




Graph nonisomorphism

- 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!

Our protagonists



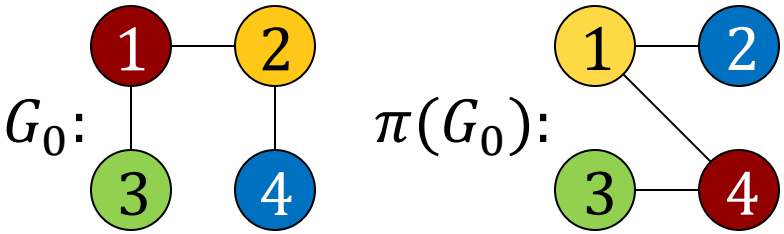
IP for Graph nonisomorphism

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

Prover sends a bit b'

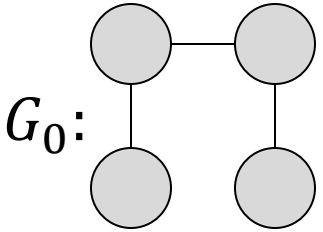
If $b = b'$ verifier accepts, otherwise verifier rejects

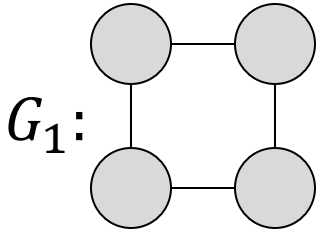
IP for Graph nonisomorphism



0

Accept





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

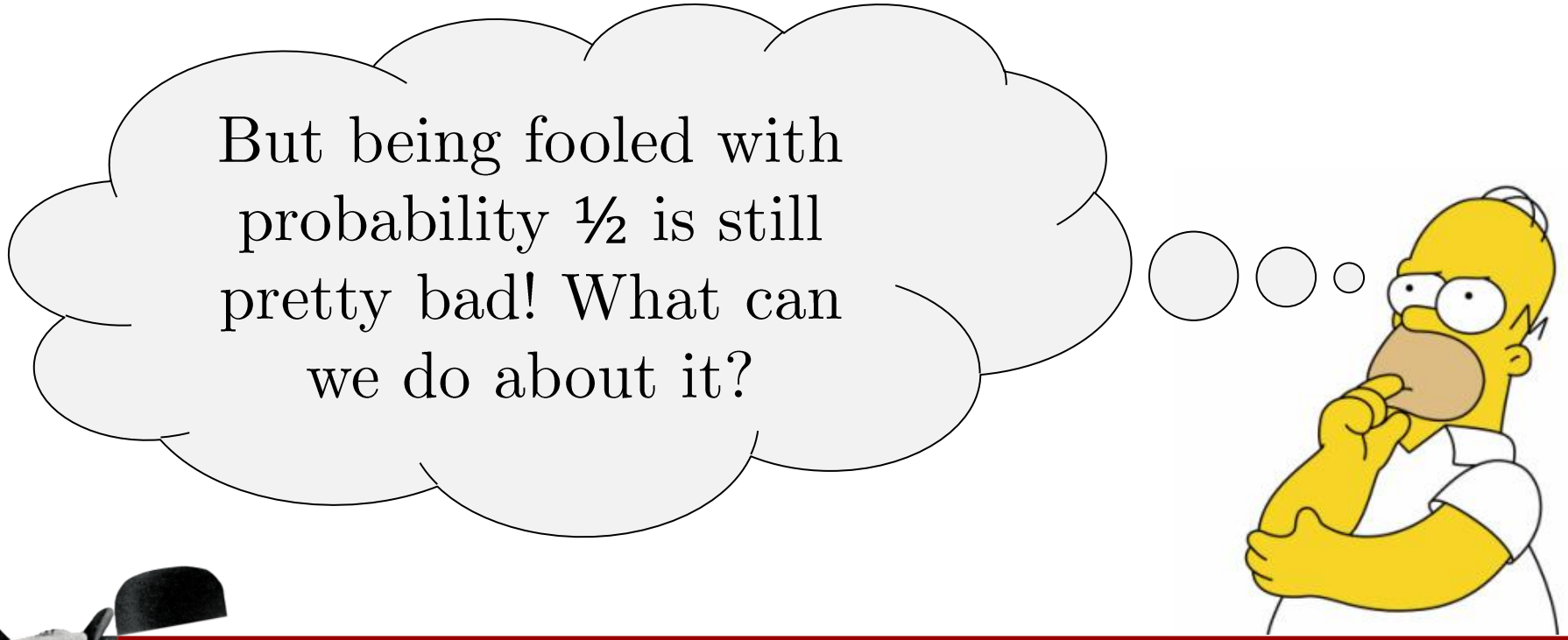
- Vote: Probability that prover will get the verifier to accept, when the graphs are nonisomorphic and isomorphic, respectively?
  1. 1 and $1/n!$
  2. 1 and $1/2$
  3. $1/2$ and $1/n!$
  4. $1/2$ and $1/2$

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 on input x:
  - 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



Interactive proofs

- Vote: What is the relation between **NP** and **IP**?
  1. **NP ⊂ IP**
  2. **IP ⊂ NP**
  3. **IP = 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

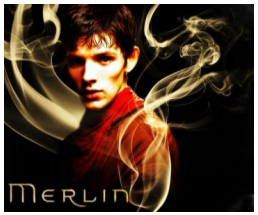
* Just for fun

Why do we need ZKPs?*

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

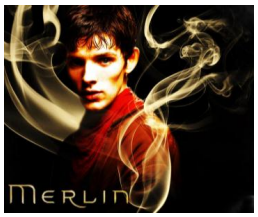
* Just for fun

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

* Just for fun

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!

* Just for fun

Zero knowledge*

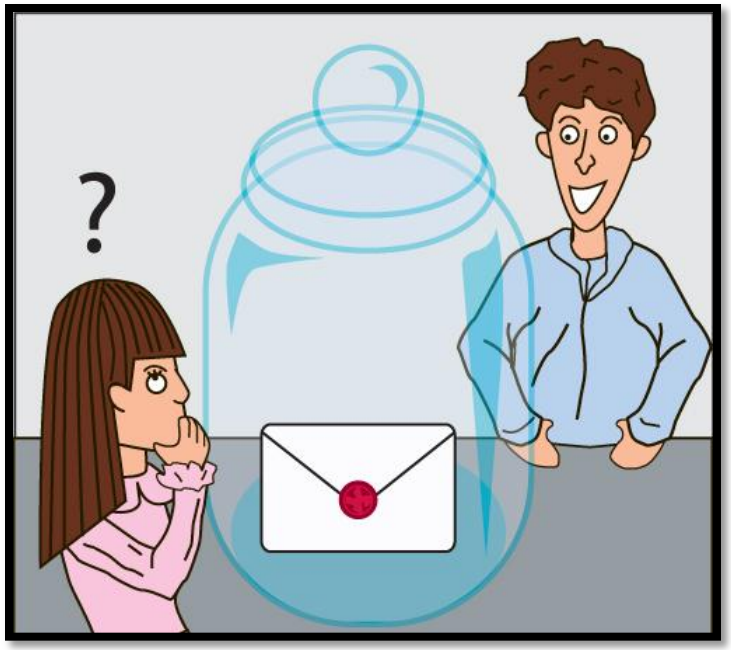
- Zero knowledge, informal definition: For any probabilistic polynomial time verifier V' and $x \in L$ there must exist a simulator $S_{V'}$ that produces the same distribution over interaction transcripts as talking with the actual prover!

A dishonest verifier can only gain info from interacting with the prover; but it can gain the same info from talking with itself!

* Just for fun

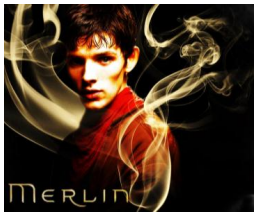
ZKP for 3-COLORING*

- 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



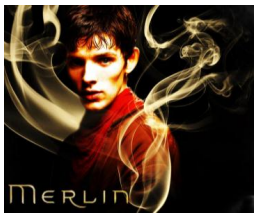
* Just for fun

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

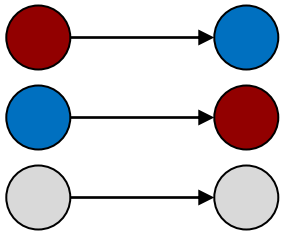


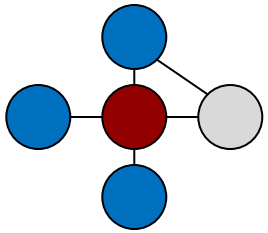
Prover reveals $a = \pi(\gamma(u))$ and $b = \pi(\gamma(v))$

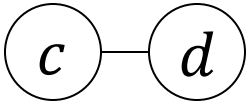
Verifier accepts iff $a \neq b$

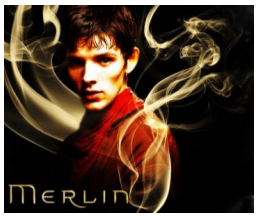
* Just for fun

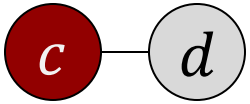
ZKP for 3-coloring*



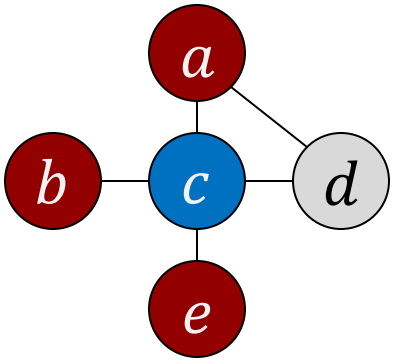








Accept



$\gamma(G)$

* Just for fun

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 $a \neq b$

- Vote: 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. $1 - \frac{1}{|E|}$

* Just for fun

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!

* Just for fun

What you need to know

- Definitions
  - Interactive proof system
  - The class **IP**
- Algorithms
  - Interactive proof system for Graph nonisomorphism