Interactive Proof Systems

CS 480

Computational Theory

Benjamin Walker

Provers and Verifiers

Recap:
NP is the class of languages that have polynomial time verifiers.

Provers and Verifiers

Recap:
NP is the class of languages that have polynomial time verifiers.

Prover	Verifier
Convince the Verifier	Verify the answer

Provers and Verifiers

Recap:
NP is the class of languages that have polynomial time verifiers.

Prover	Verifier
Convince the Verifier	Verify the answer
No computational constraints	Polynomial time only

Provers and Verifiers

$$SAT = (\overline{a} \vee \overline{b} \vee c \vee k \vee \overline{u}) \wedge (a \vee \overline{g}) \wedge \ldots \wedge (r \vee \overline{y} \vee z)$$

Provers and Verifiers

$$SAT = (\overline{a} \vee \overline{b} \vee c \vee k \vee \overline{u}) \wedge (a \vee \overline{g}) \wedge \ldots \wedge (r \vee \overline{y} \vee z)$$

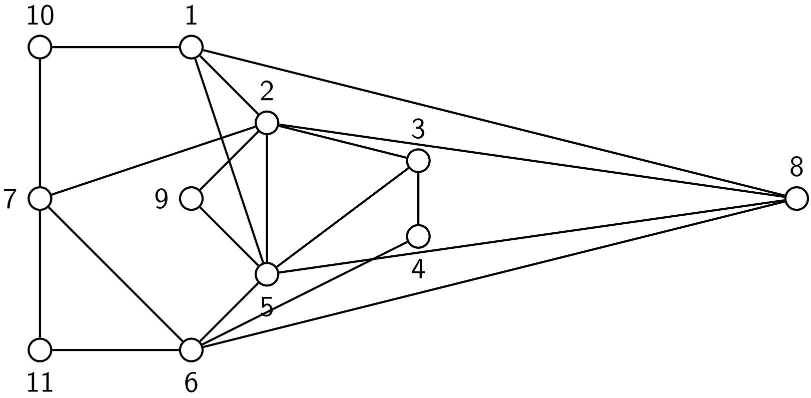
Prover

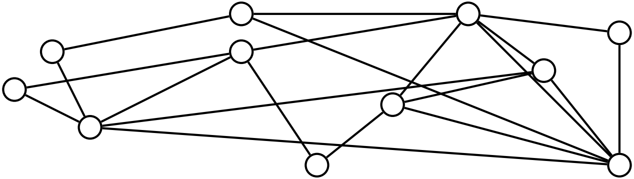
Provide Verifier with values

Verifier

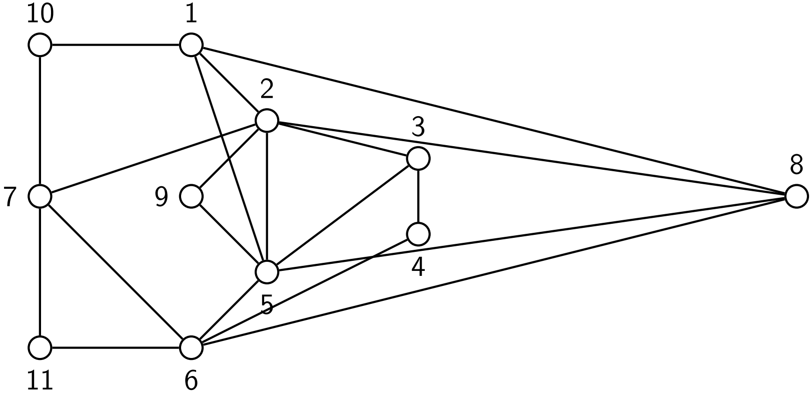
Plug values into SAT problem to verify

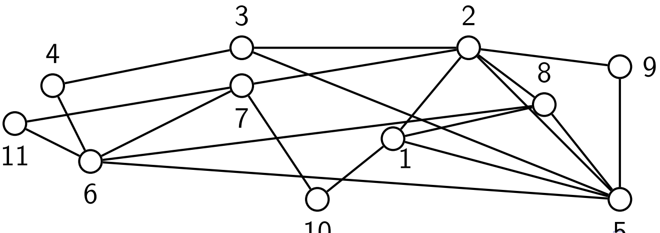
Isomorphic?





Isomorphic?





Provers and Verifiers

The prover can convince the Verifier of a correct answer in polynomial time,

Provers and Verifiers

The prover can convince the Verifier of a correct answer in polynomial time, but can the Prover prove to the Verifier that an incorrect answer is not correct in polynomial time?

Provers and Verifiers

The prover can convince the Verifier of a correct answer in polynomial time, but can the Prover prove to the Verifier that an incorrect answer is not correct in polynomial time?

Interestingly, YES!

Provers and Verifiers

The prover can convince the Verifier of a correct answer in polynomial time, but can the Prover prove to the Verifier that an incorrect answer is not correct in polynomial time?

Interestingly, YES!

...Provided we give some leeway to our Prover and Verifier definitions.

Provers and Verifiers

Prover

Convince the Verifier

No computational constraints

Verifier

Verify the answer

Polynomial time only

Provers and Verifiers

Prover

Convince the Verifier

No computational constraints

Can engage in a two-way dialog with the Verifier

Verifier

Verify the answer

Polynomial time only

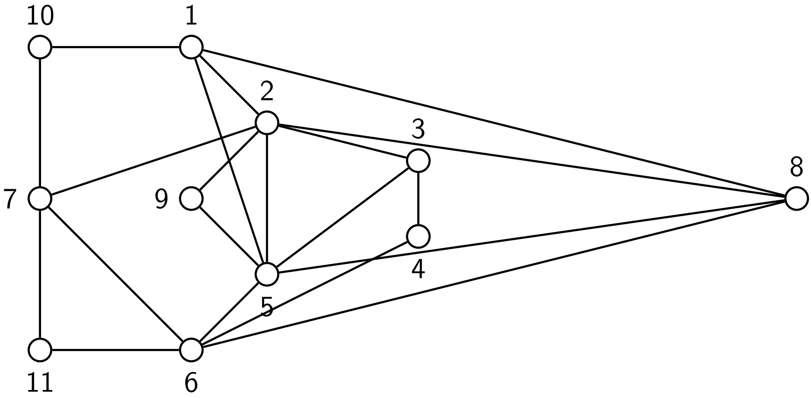
Allowed to be a Probabilistic Polynomial Turing machine

Provers and Verifiers

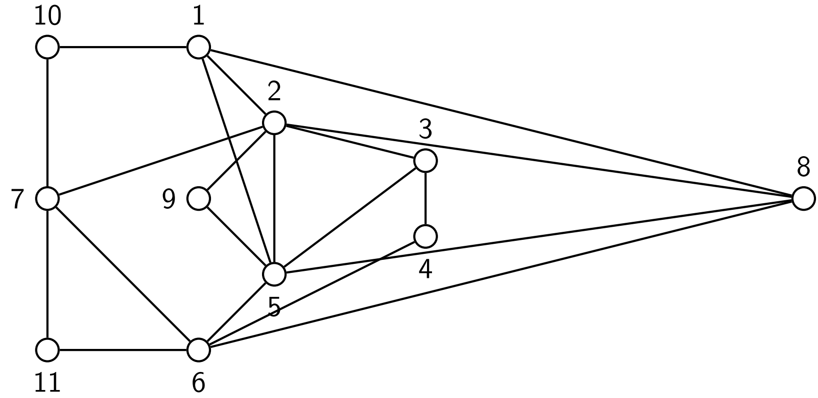
Prover	Verifier
Convince the Verifier	Verify the answer
No computational constraints	Polynomial time only
Can engage in a two-way dialog with the Verifier	Allowed to be a Probabilistic Polynomial Turing machine

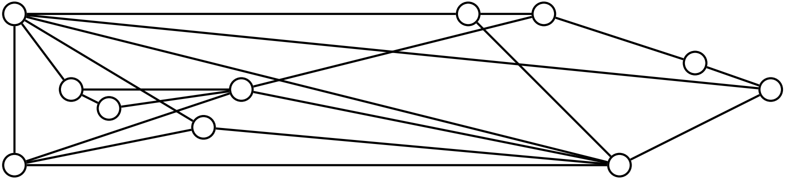
This is what makes an Interactive Proof System.

Isomorphic?

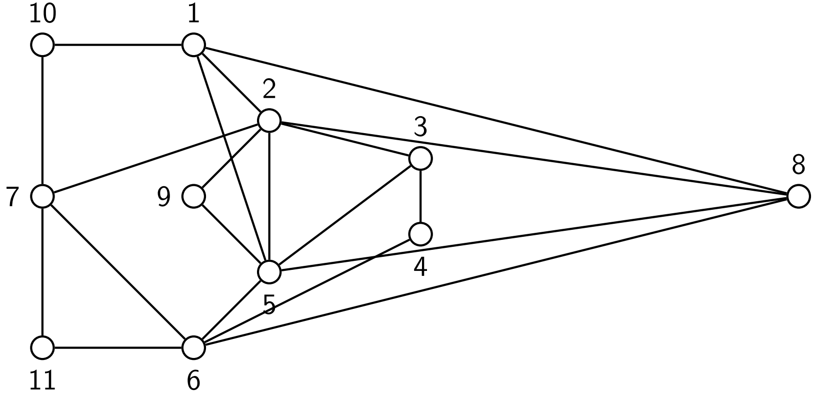


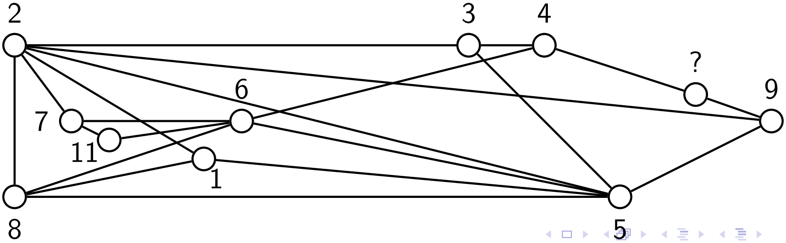
Isomorphic?





Isomorphic?





Uses of Interactive Proof Systems

Uses of Interactive Proof Systems

1. Profoundly affected complexity theory [Multiple Provers]

Uses of Interactive Proof Systems

1. Profoundly affected complexity theory [Multiple Provers]
2. Advances in cryptography [Zero Knowledge]

Uses of Interactive Proof Systems

1. Profoundly affected complexity theory [Multiple Provers]
2. Advances in cryptography [Zero Knowledge]
3. Advances in approximation algorithms

Things I didn't talk about

1. Approximate Shortest Lattice Vector is another one of the "elusive" problems
   1.1 "elusive" = NP Problems not known to be in P or to be NP-Complete
   1.2 Approximation algorithm techniques (Interactive Proof System techniques) are used to help find answers to this problem
2. The set of languages which have interactive proof systems is equivalent to PSPACE
3. MIP (Multiprover Interactive Proofs) is equivalent to NEXP