Constraint Satisfaction and Pebble Games

Víctor Dalmau

Universitat Pompeu Fabra

Goals:

- Show that existential pebble games (and some other variants of the game) arise naturally in the study of CSP and have many applications.
- Present a (rather personal) overview of the use of games in CSP with special emphasis on the computational complexity aspects.

Talk outline:

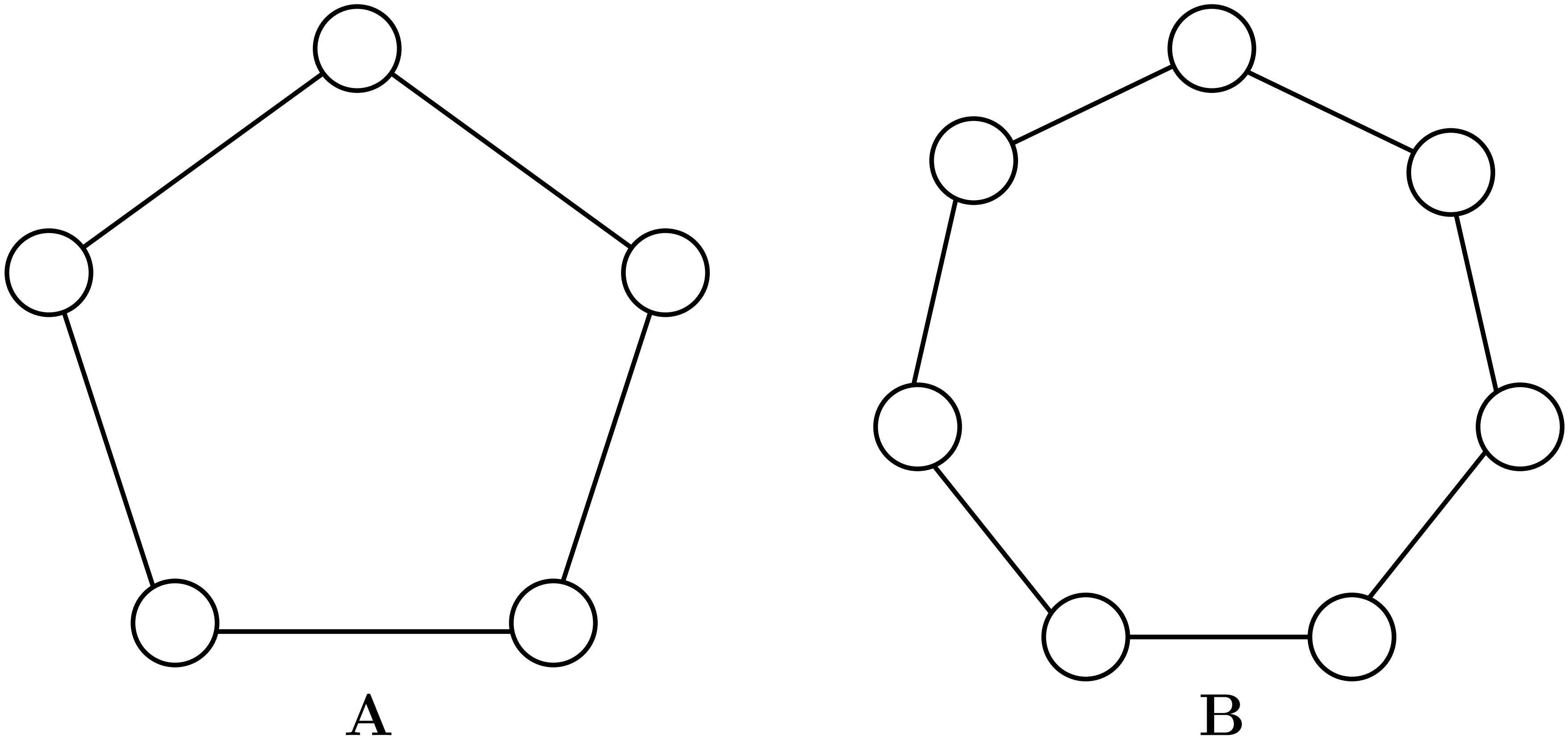
- Existential pebble games
  - Structural restrictions
  - Language restrictions
- Pebble Relation games
- Cover games

Existential k-pebble game

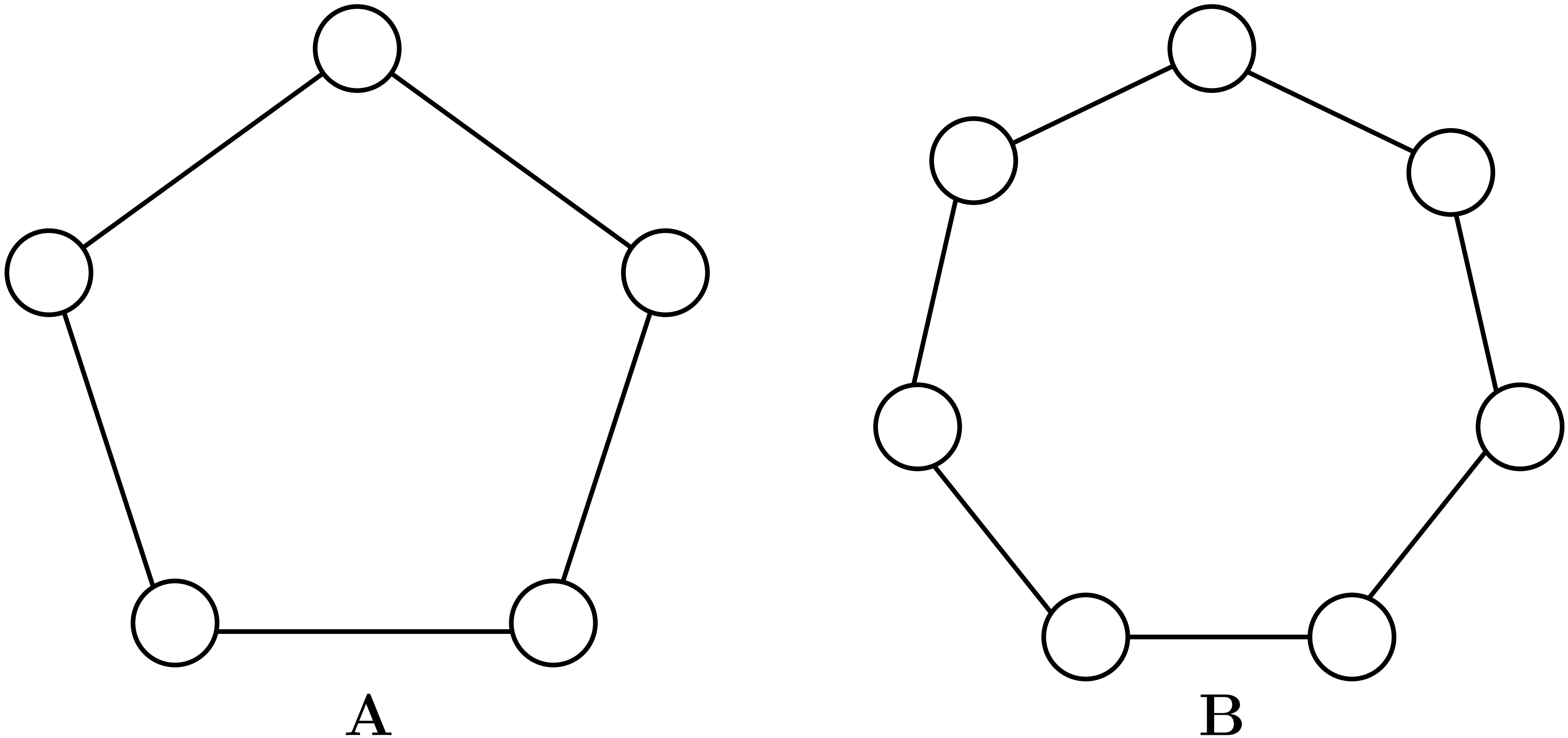
[Kolaitis, Vardi 95]

- Spoiler and Duplicator play on structures $\mathbf{A}$ and $\mathbf{B}$. Each player has k pebbles. In each move,
  - Spoiler places pebble on an element a_i of $\mathbf{A}$ or removes one of its pebbles.
  - Duplicator duplicates the move on $\mathbf{B}$.
- Spoiler wins if the mapping h taking $a_i \rightarrow b_i$ is not a *partial homomorphism*
- Duplicator wins if he has an strategy that allows him to play forever.

Example

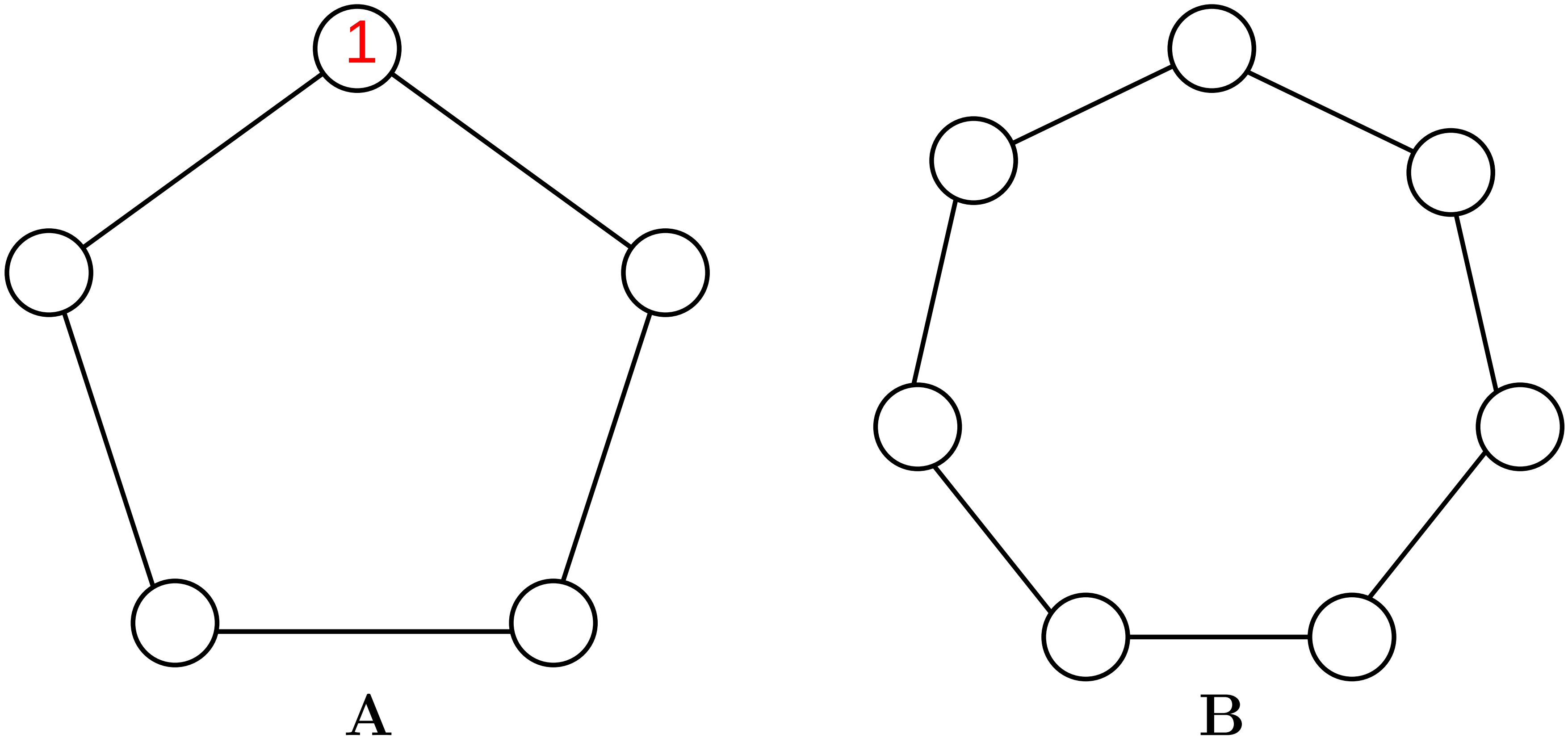


Example



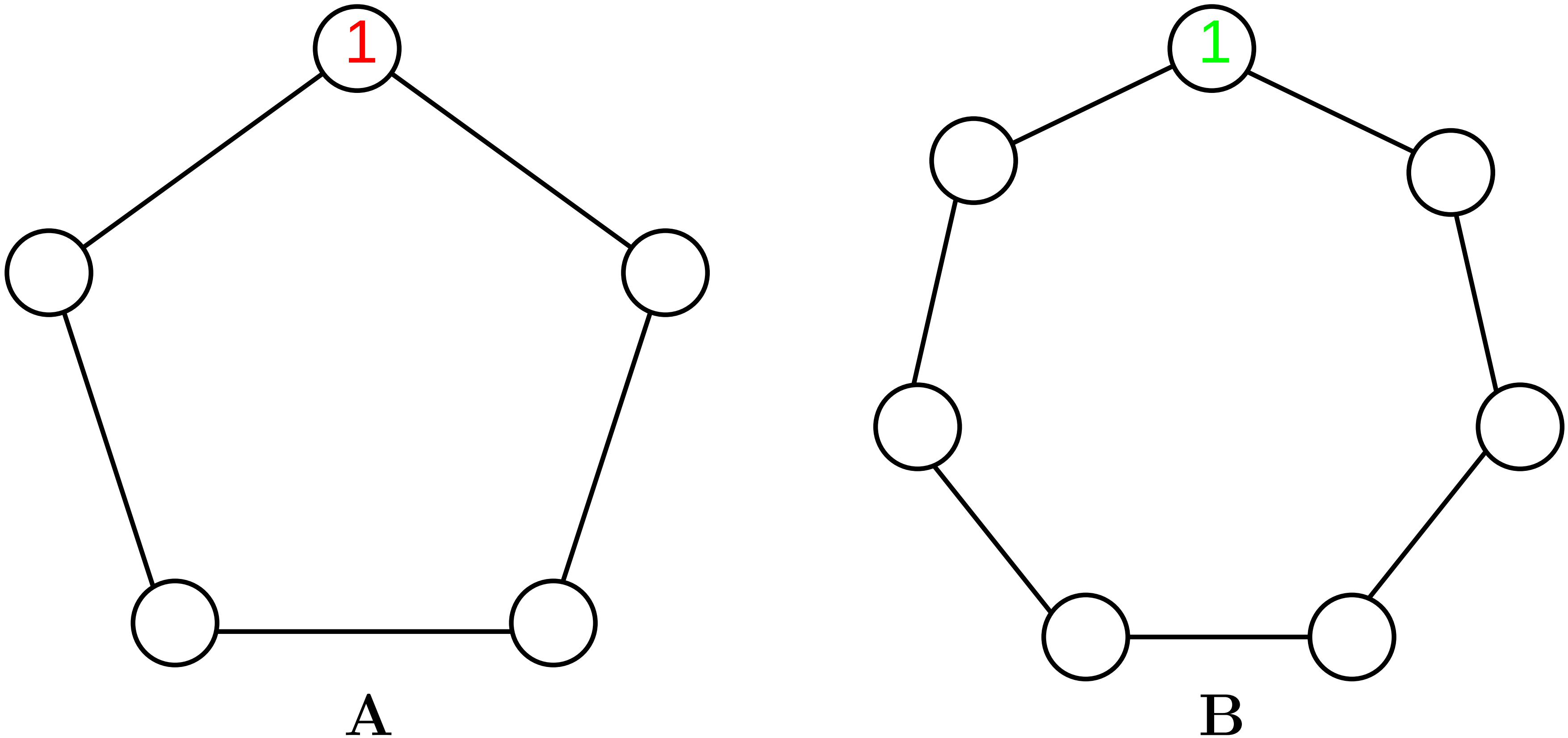
Duplicator wins the 2-pebble game

Example



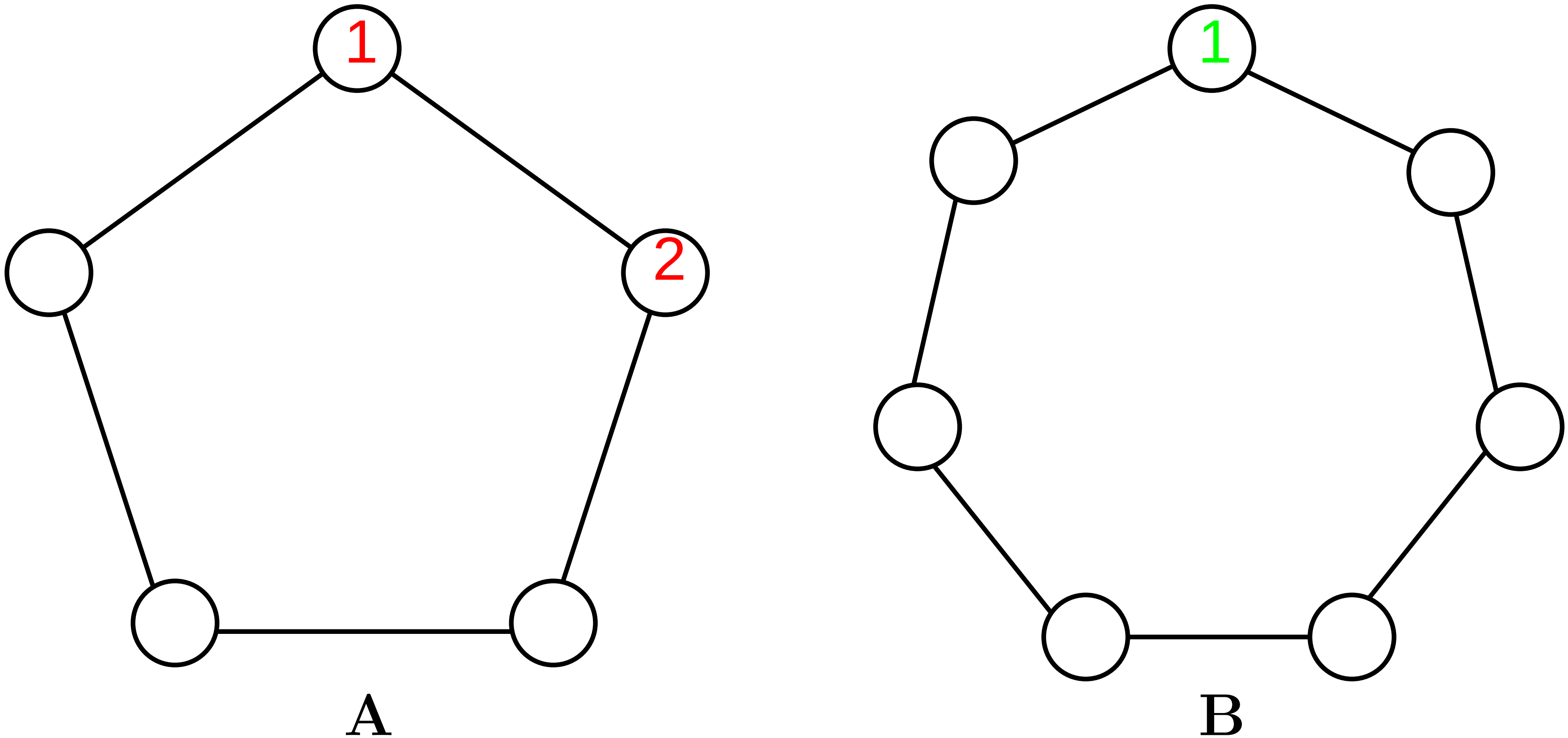
Duplicator wins the 2-pebble game

Example



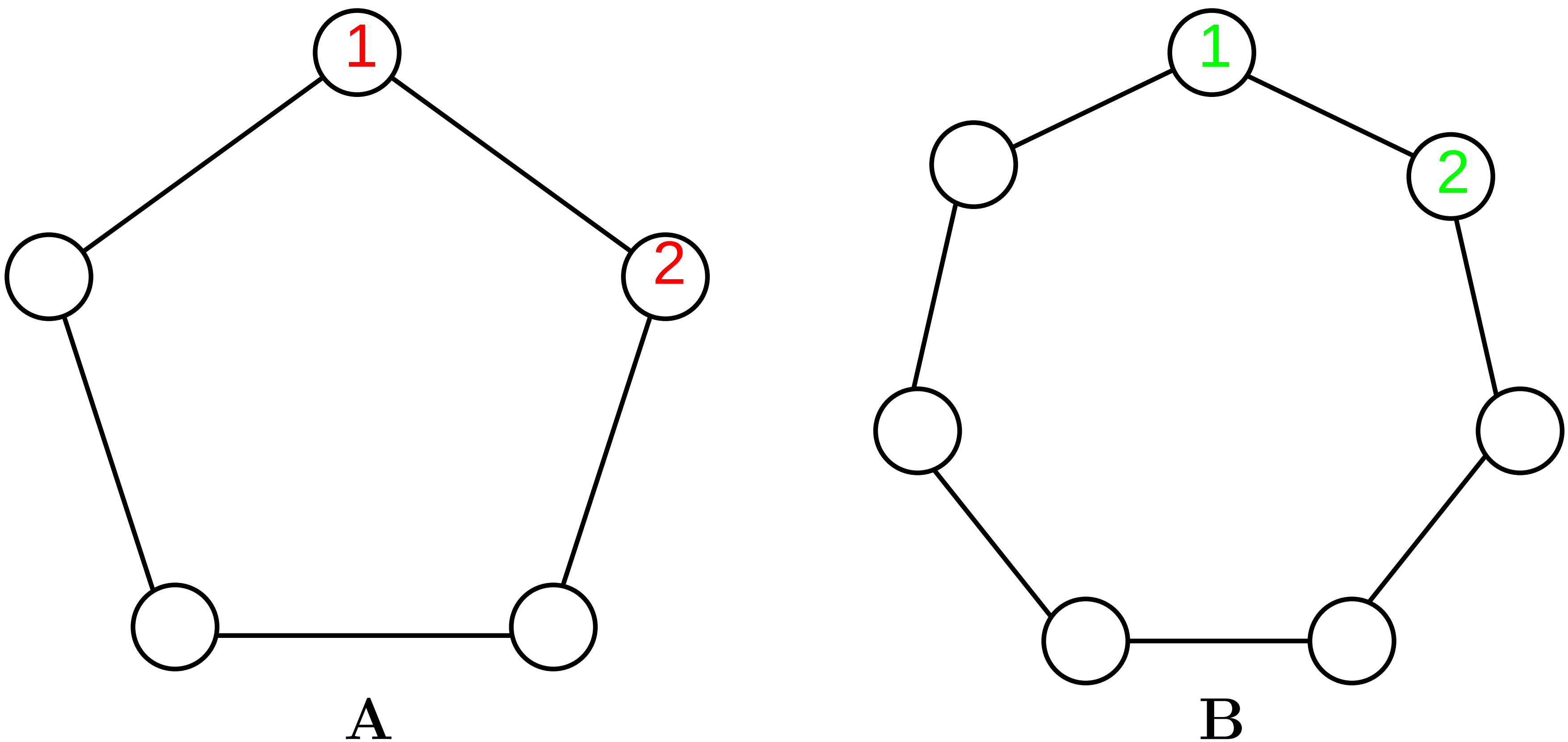
Duplicator wins the 2-pebble game

Example



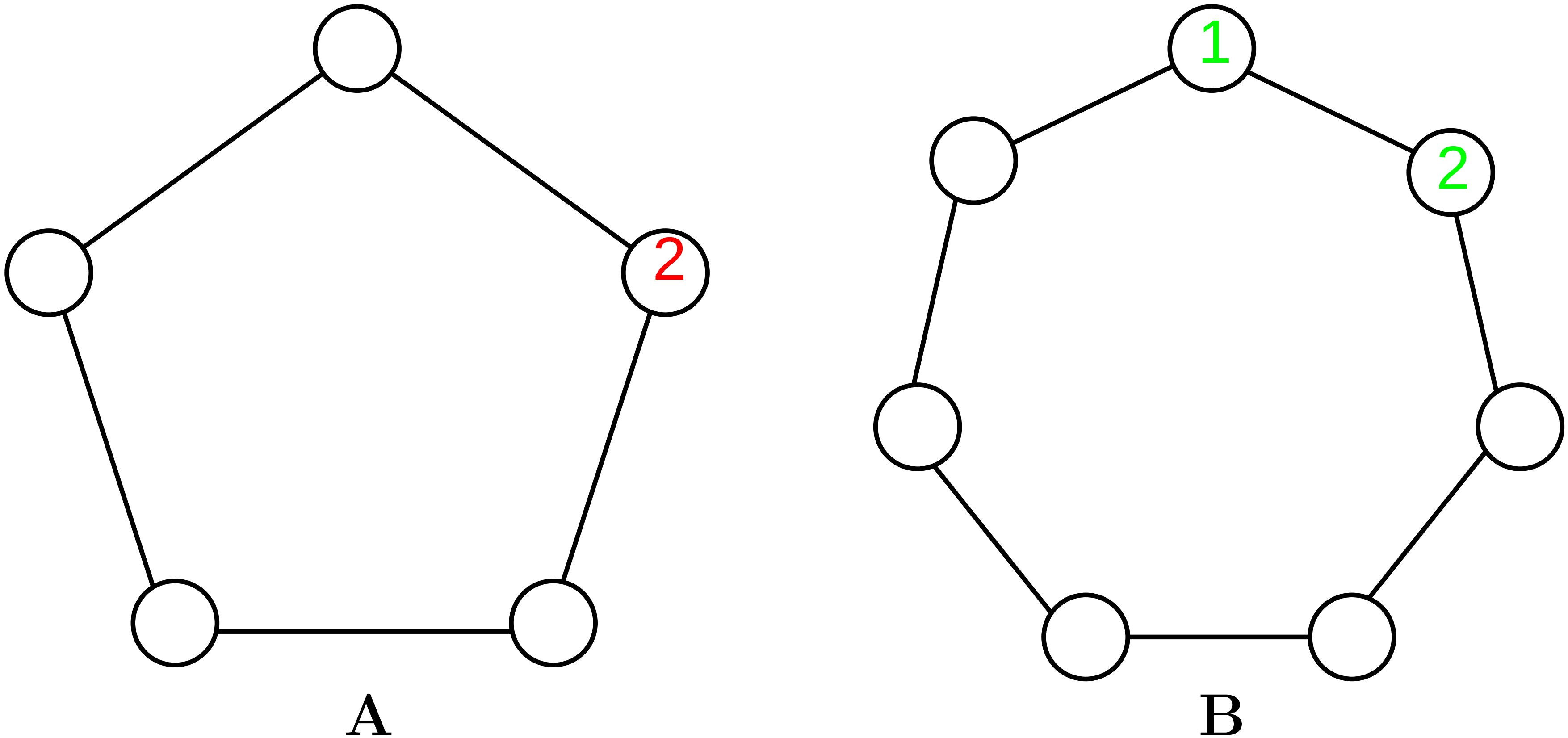
Duplicator wins the 2-pebble game

Example



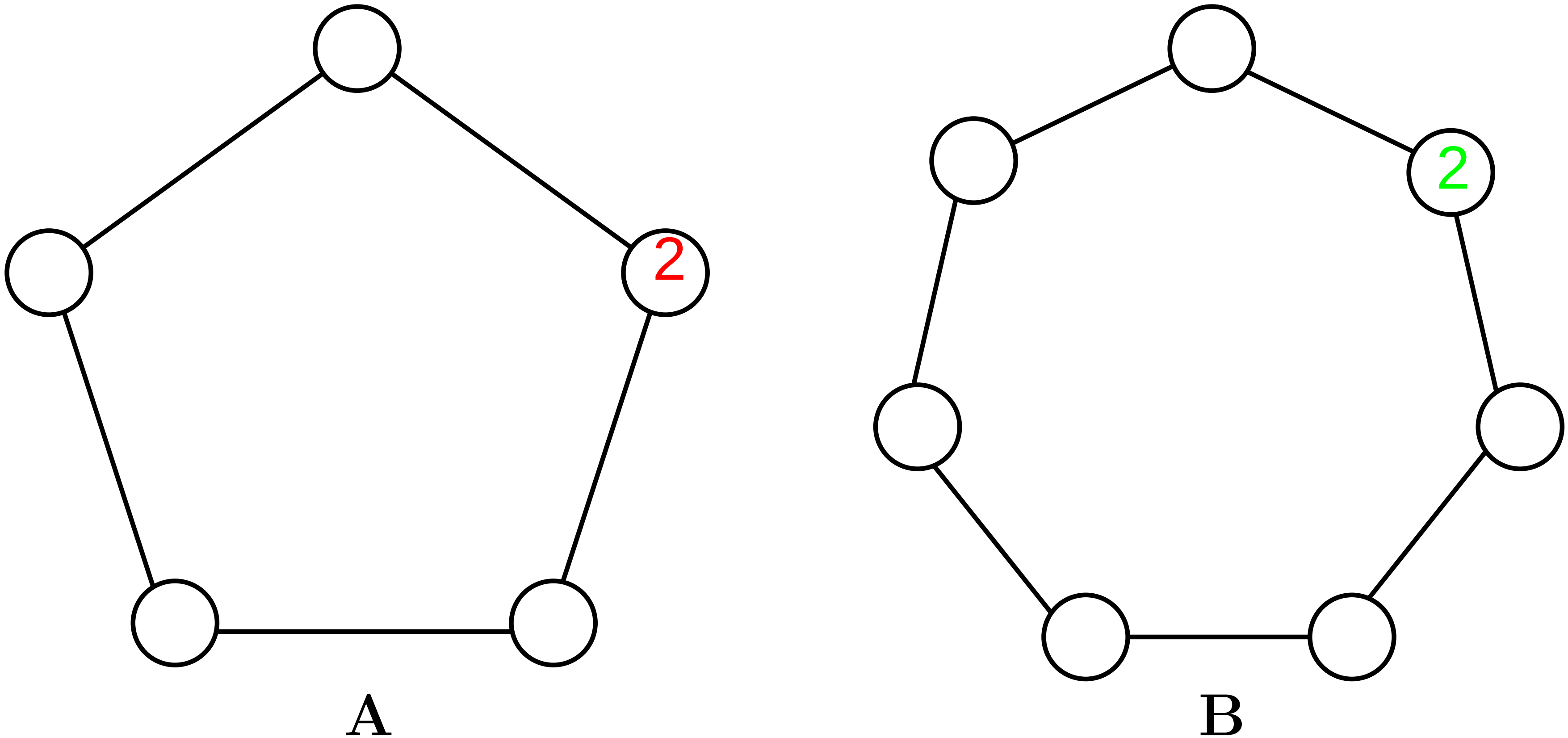
Duplicator wins the 2-pebble game

Example



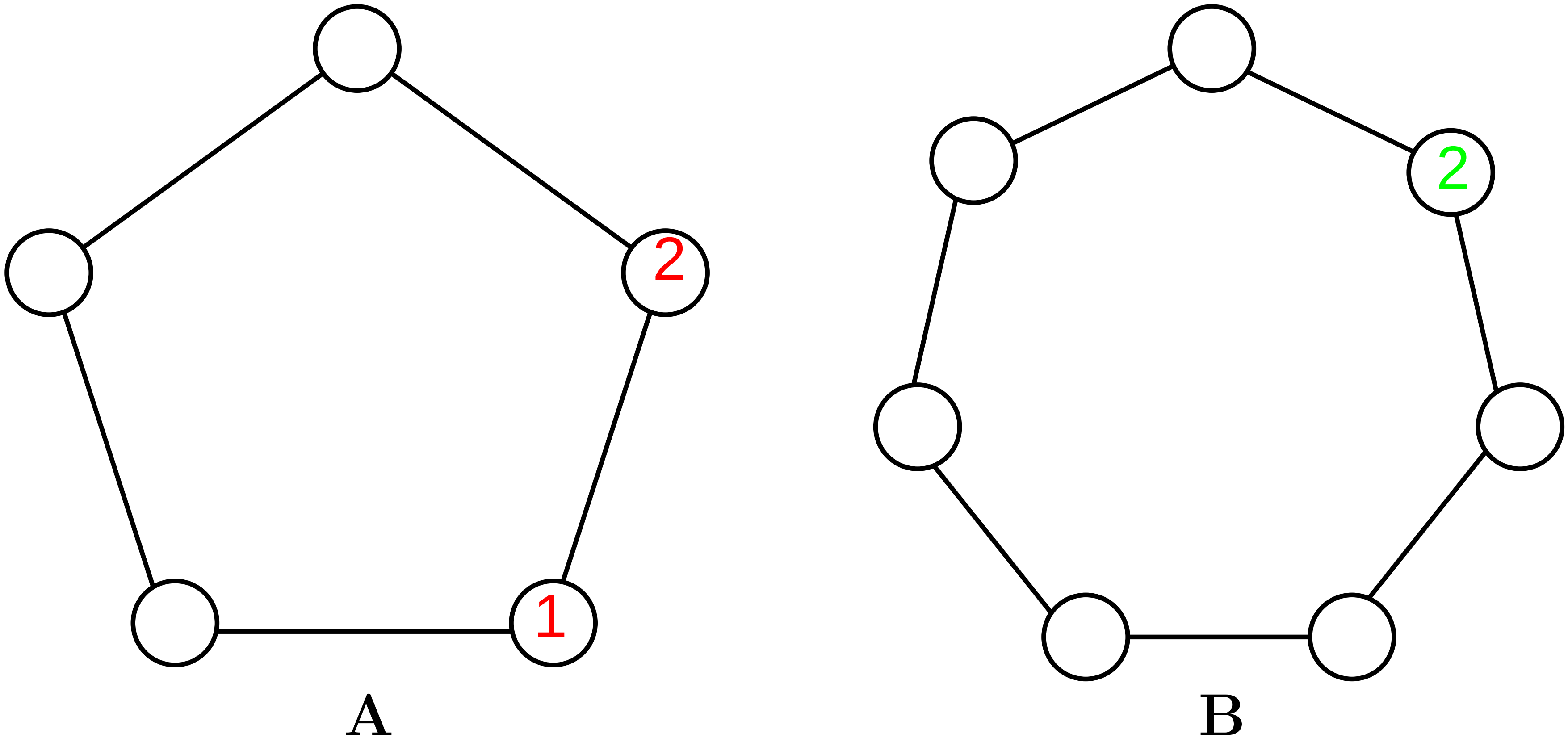
Duplicator wins the 2-pebble game

Example



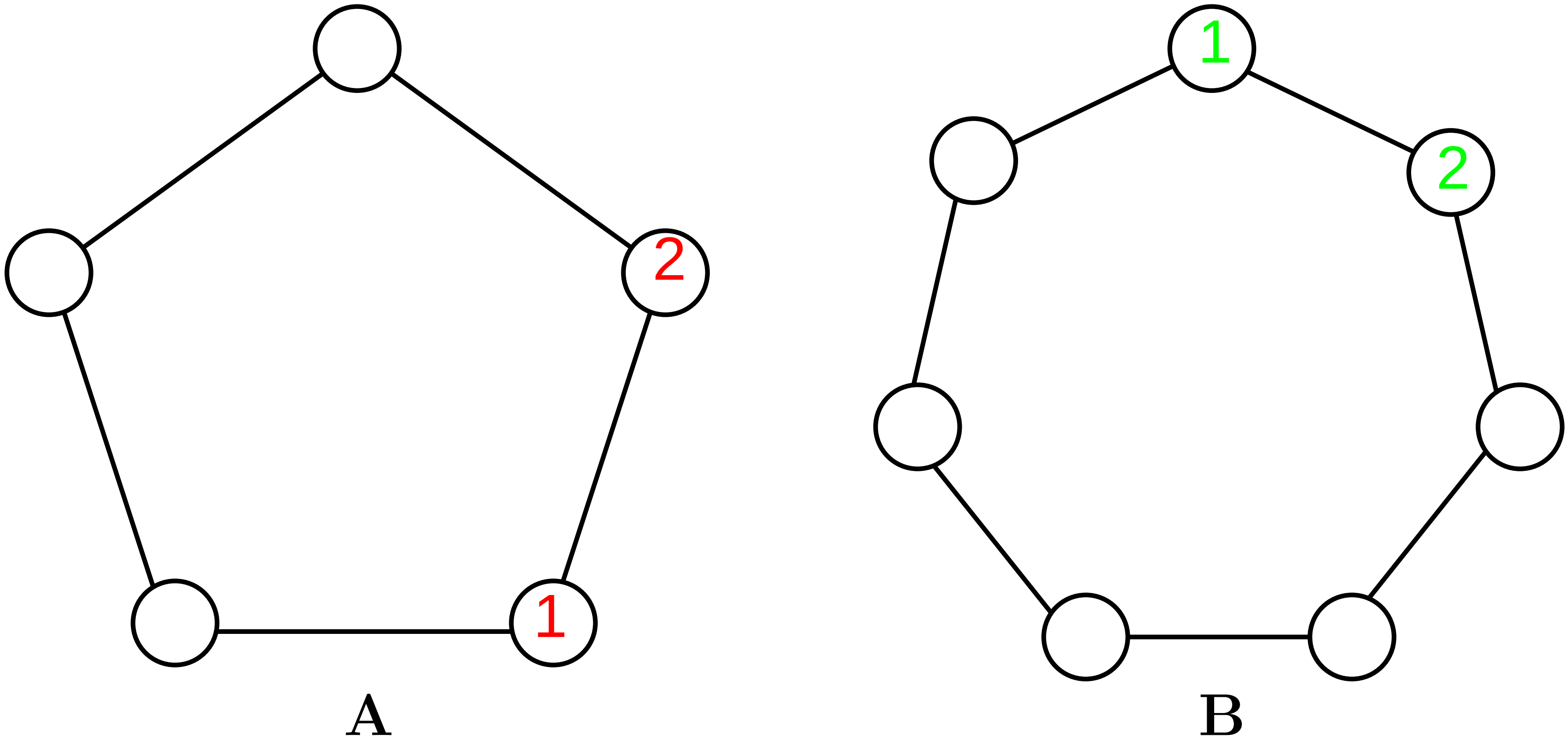
Duplicator wins the 2-pebble game

Example



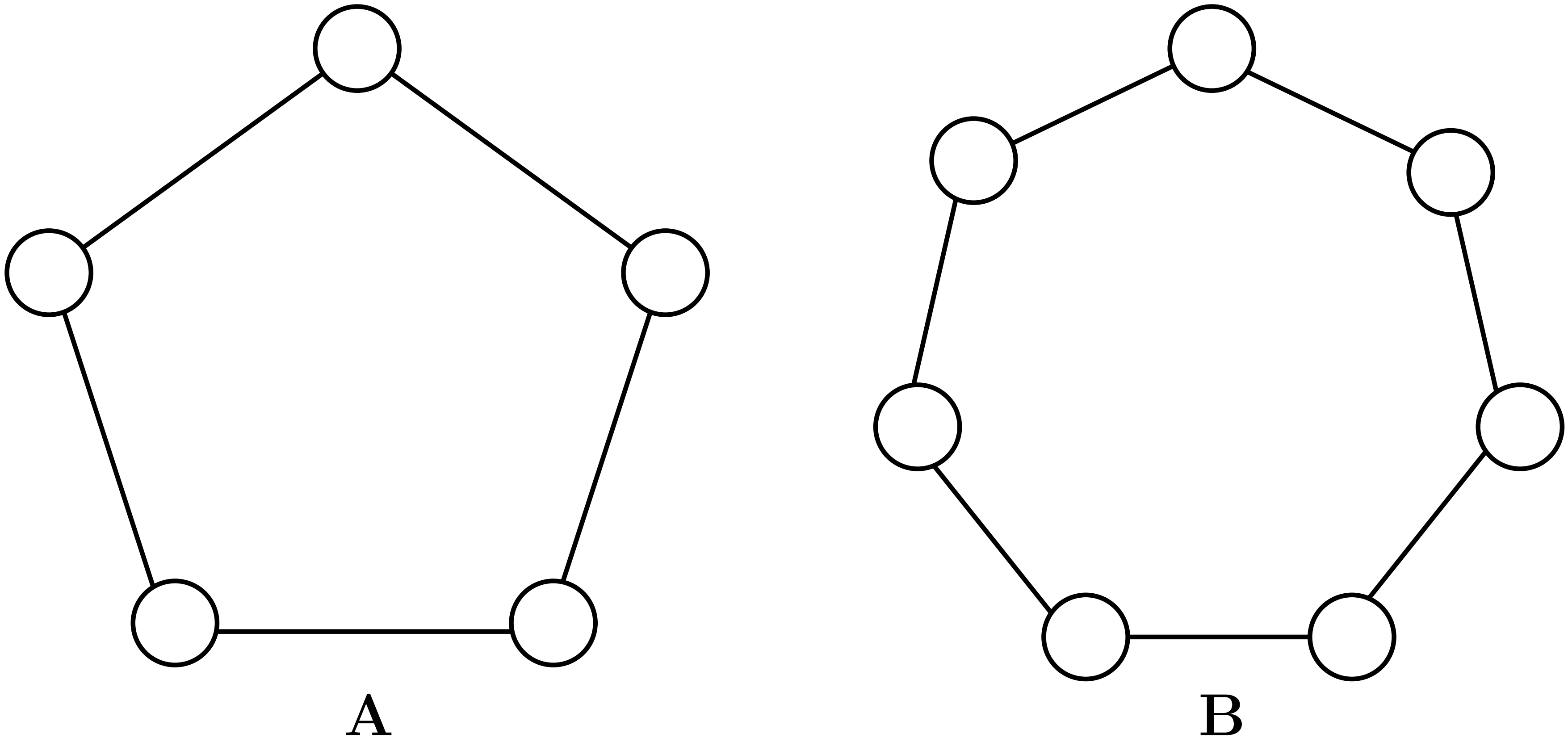
Duplicator wins the 2-pebble game

Example



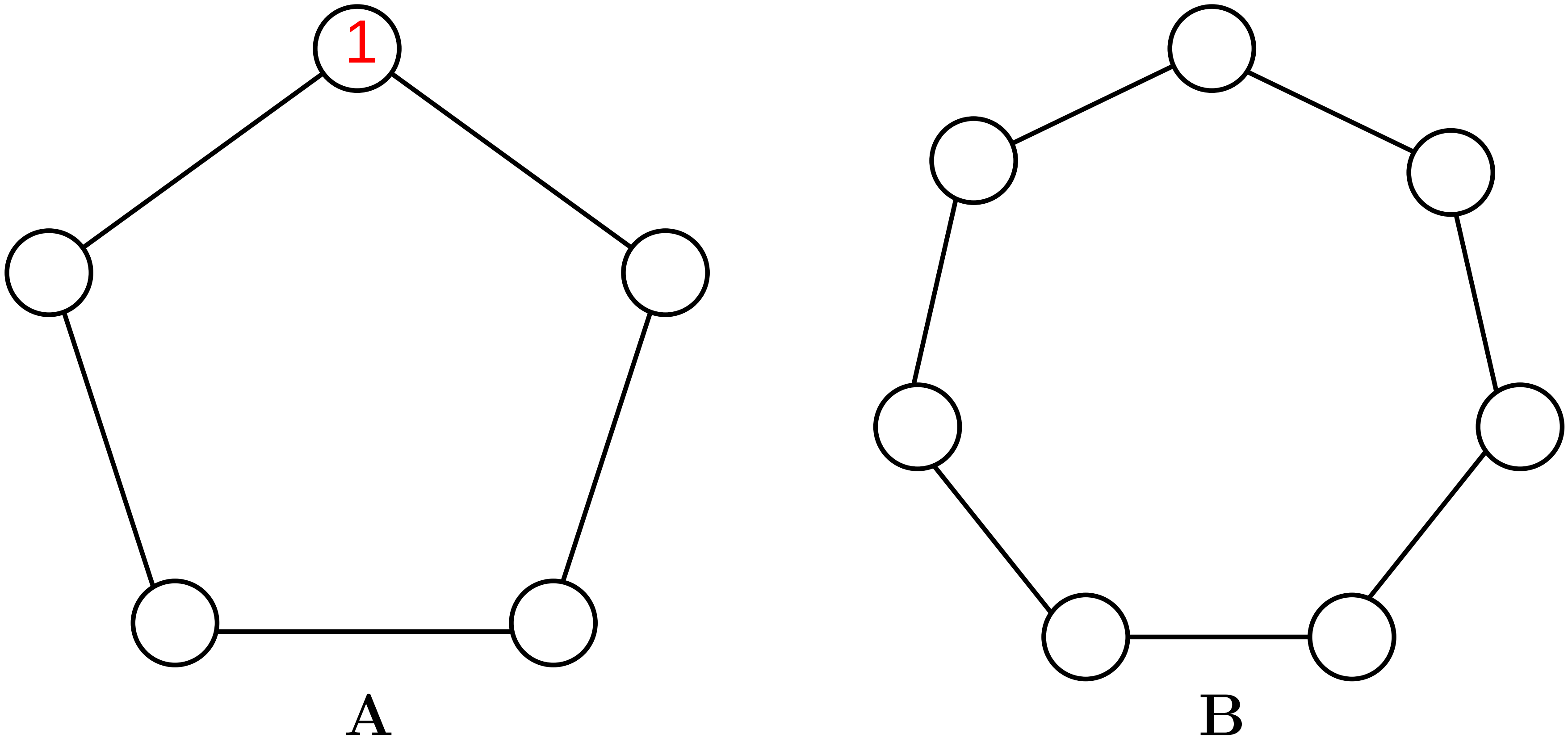
Duplicator wins the 2-pebble game

Example



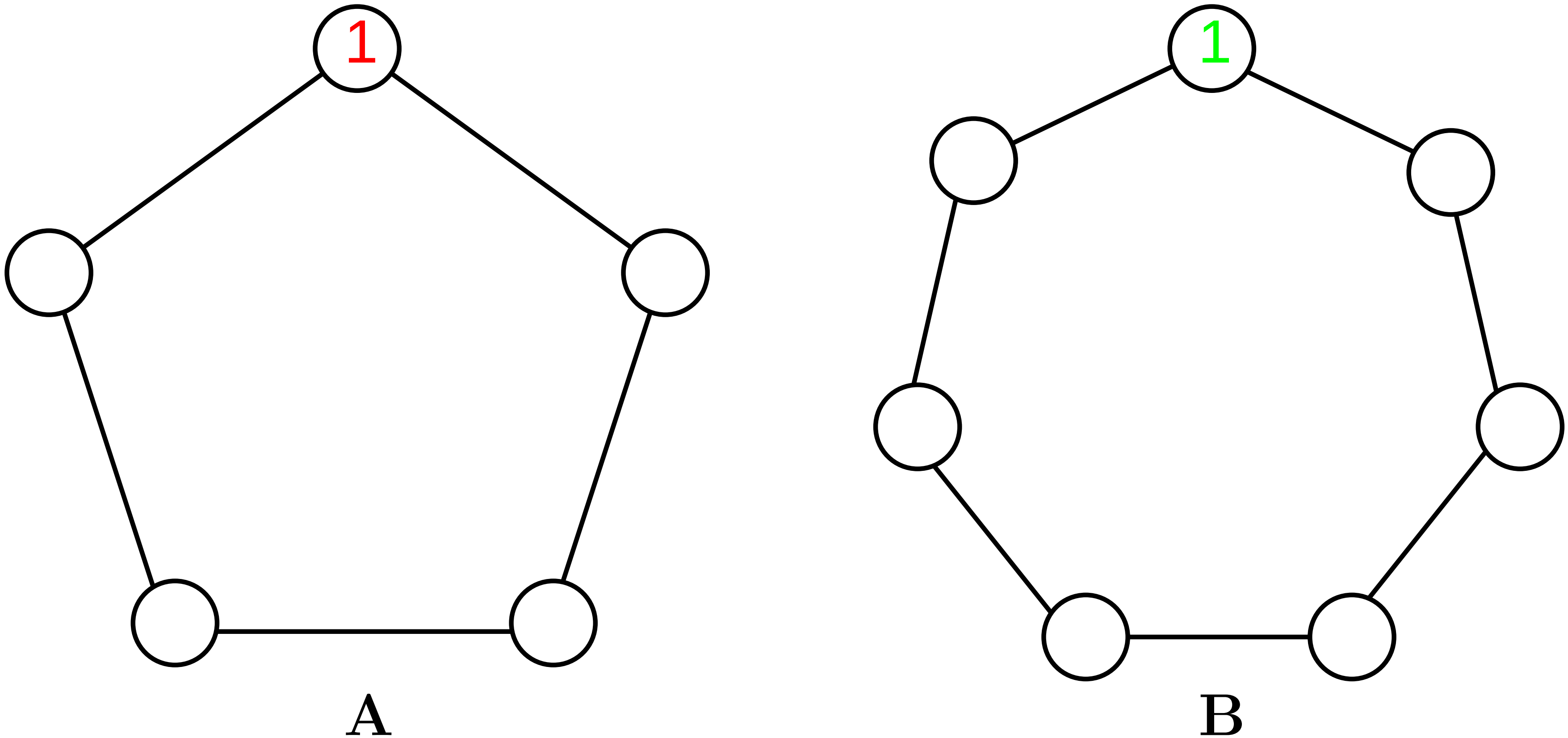
Duplicator wins the 2-pebble game
Spoiler wins the 3-pebble game

Example



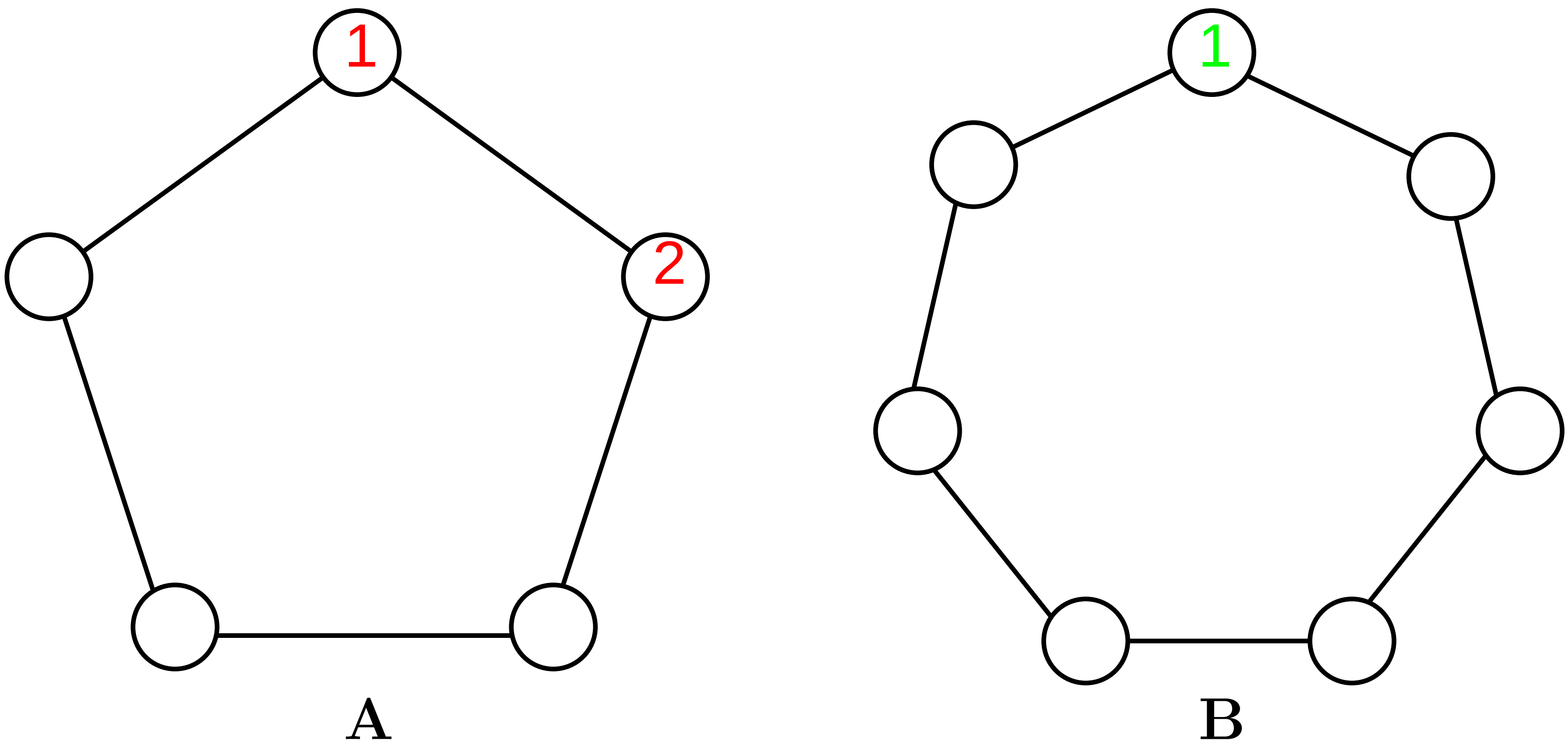
Duplicator wins the 2-pebble game
Spoiler wins the 3-pebble game

Example



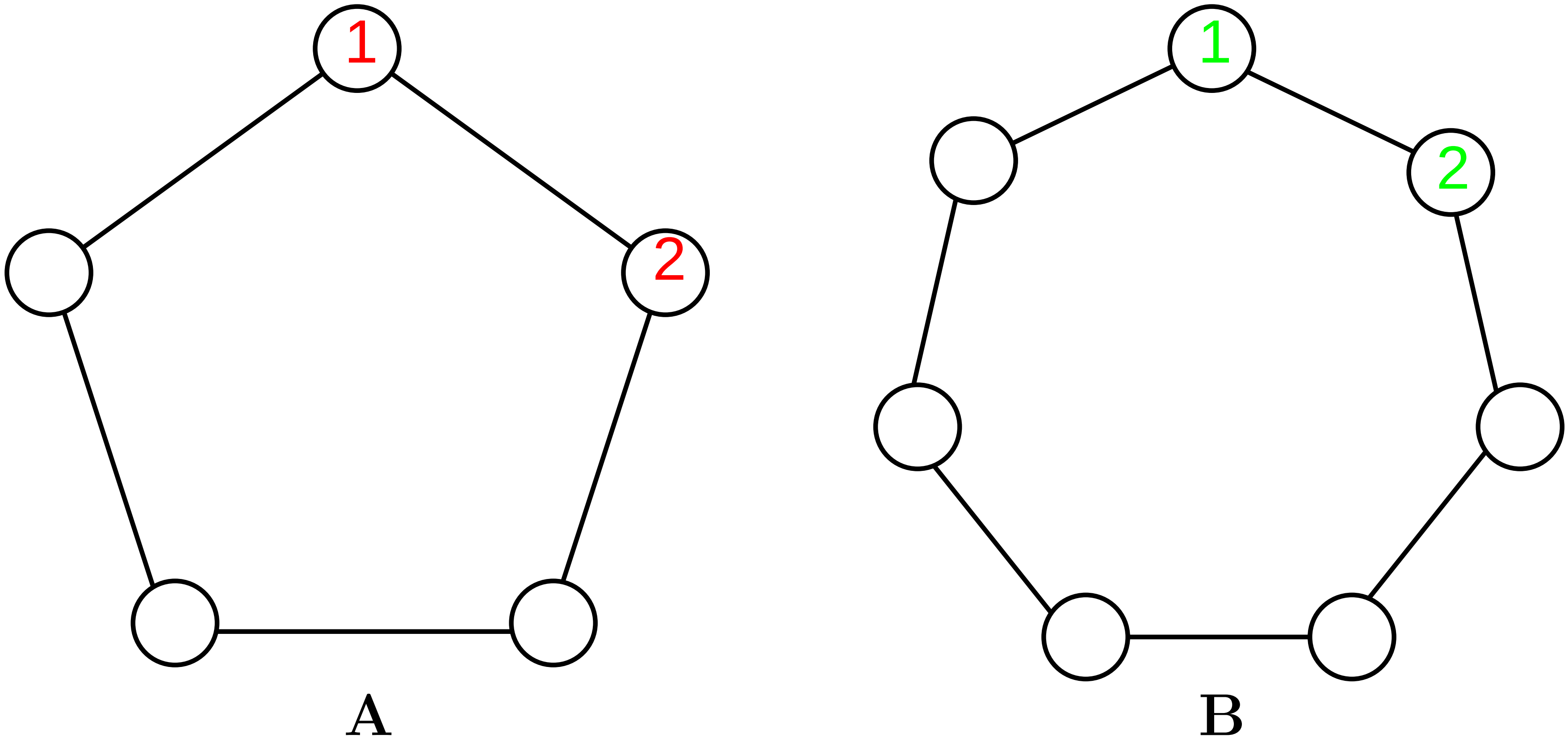
Duplicator wins the 2-pebble game
Spoiler wins the 3-pebble game

Example



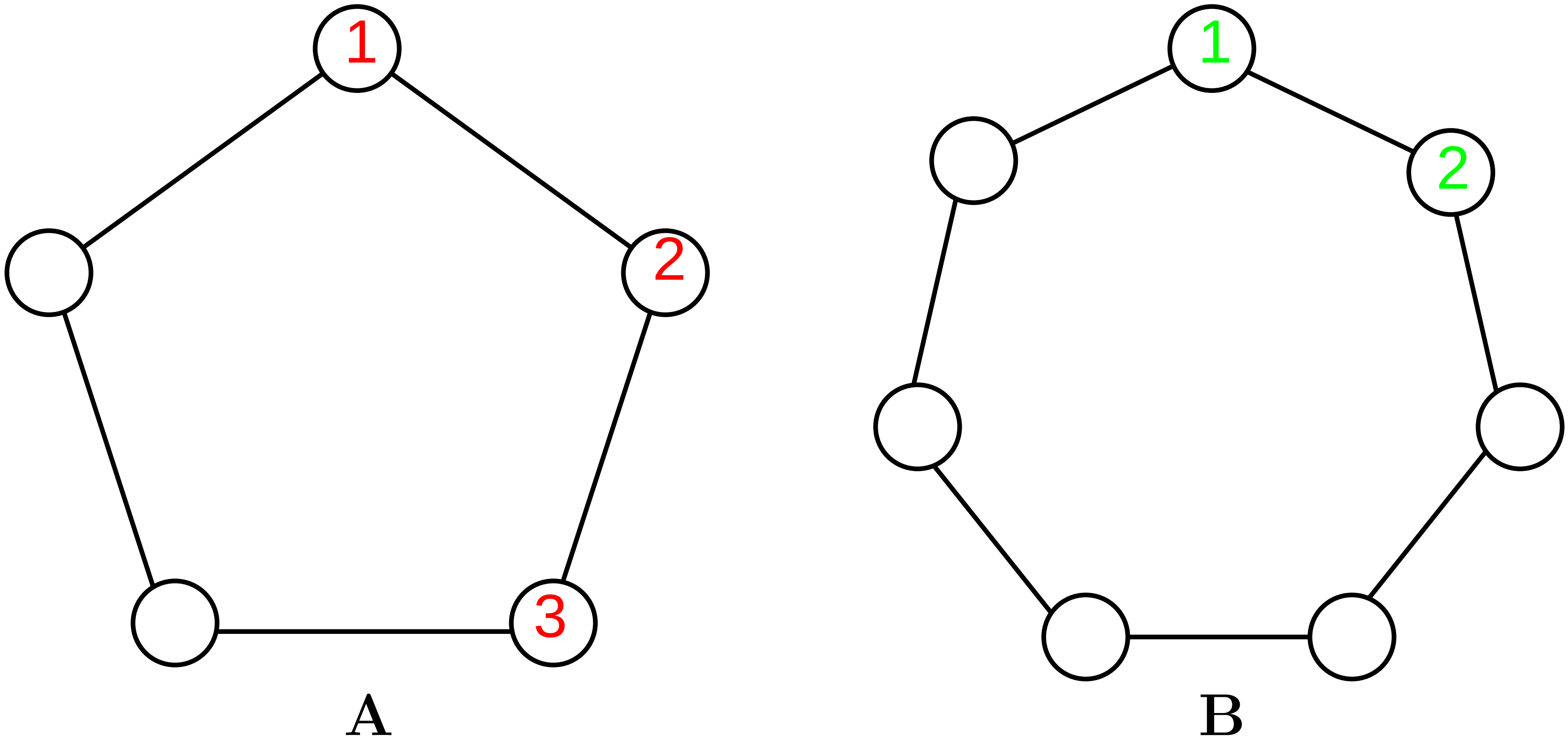
Duplicator wins the 2-pebble game
Spoiler wins the 3-pebble game

Example



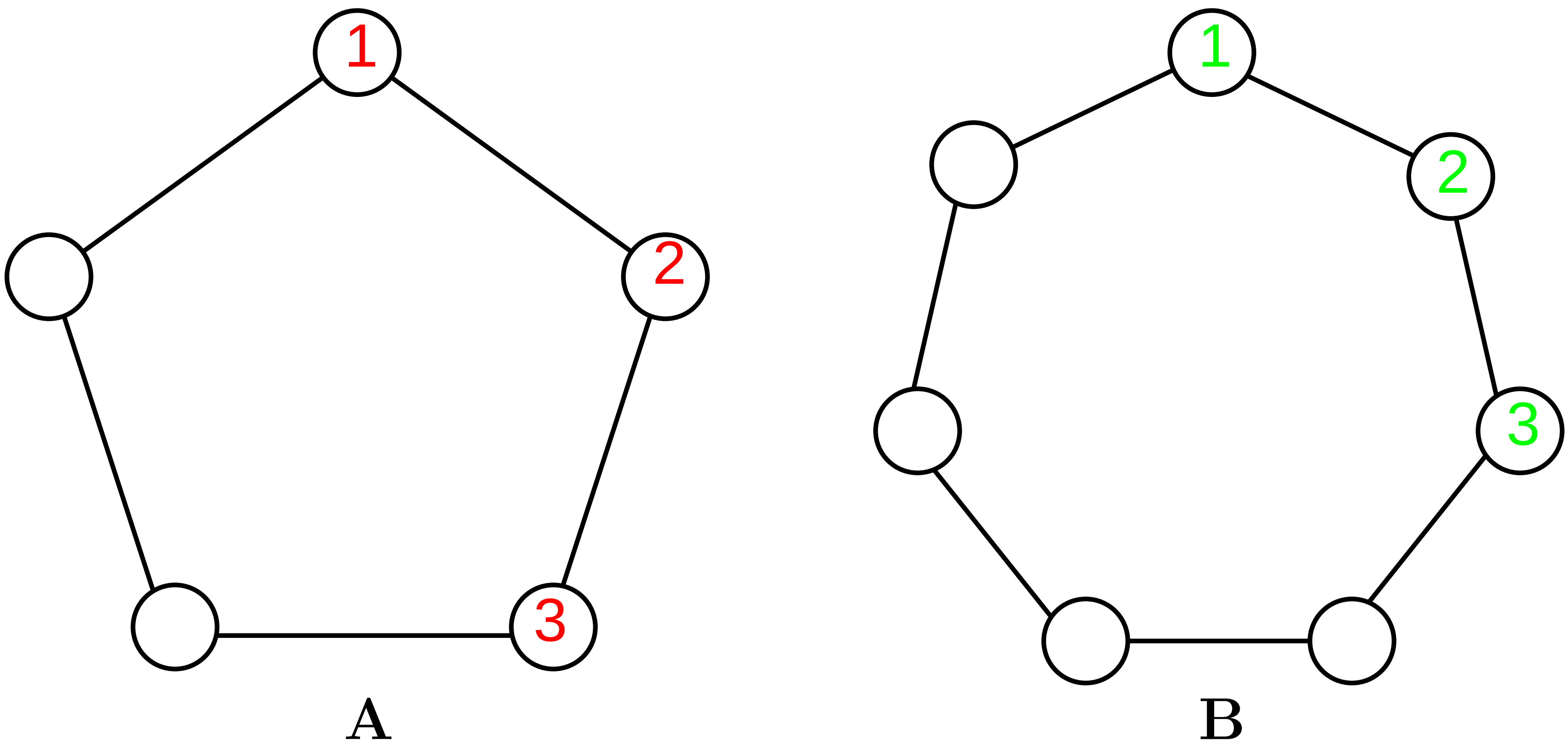
Duplicator wins the 2-pebble game
Spoiler wins the 3-pebble game

Example



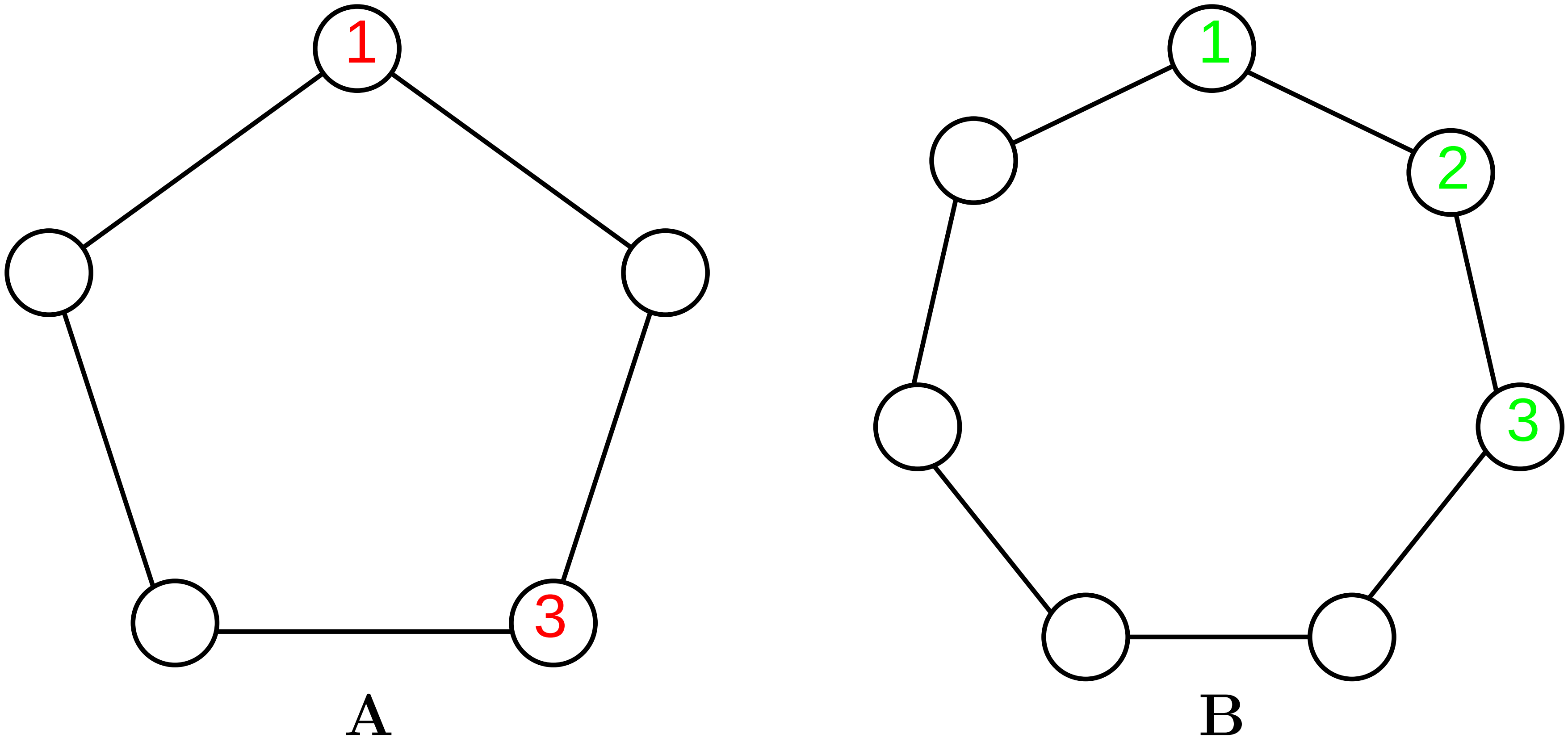
Duplicator wins the 2-pebble game
Spoiler wins the 3-pebble game

Example



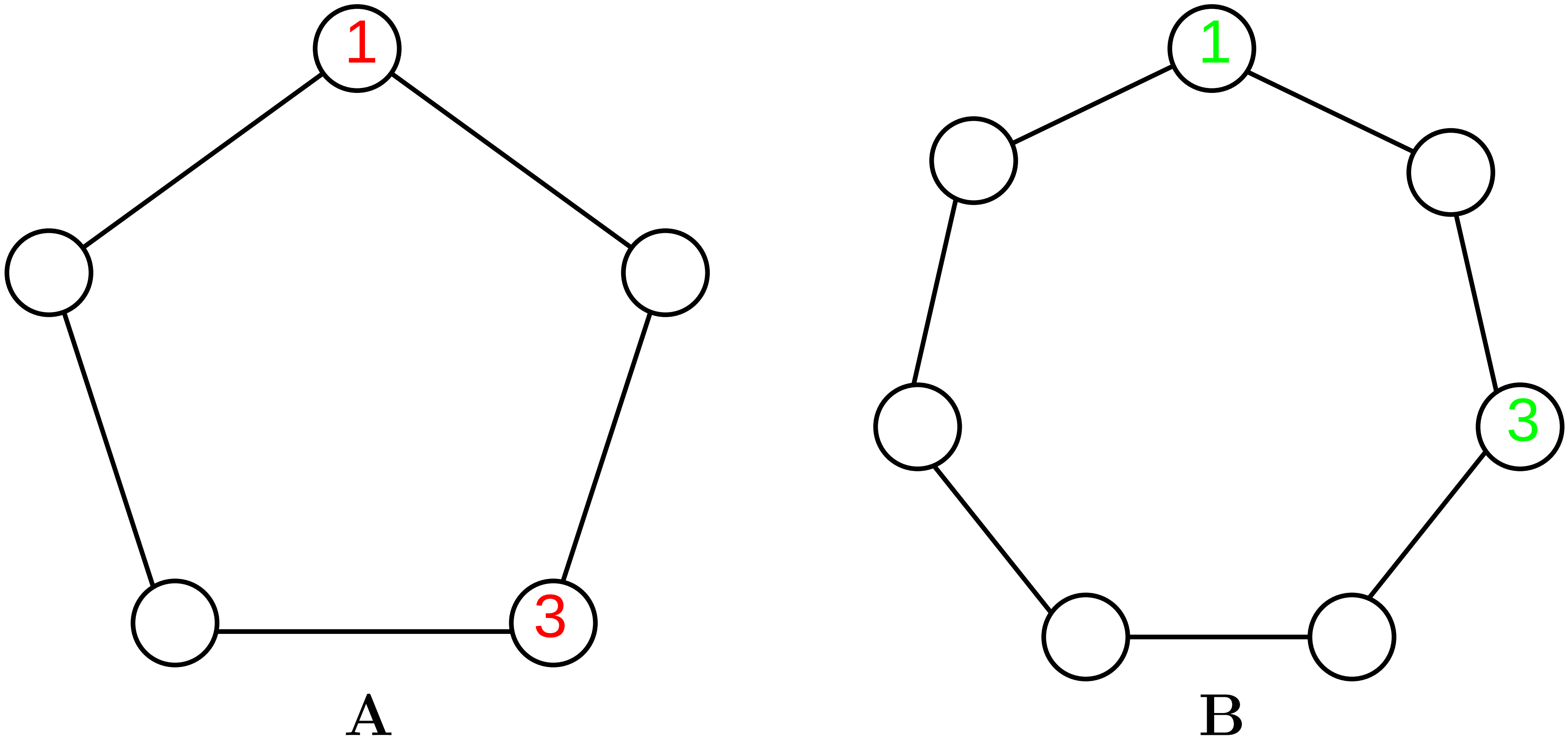
Duplicator wins the 2-pebble game
Spoiler wins the 3-pebble game

Example



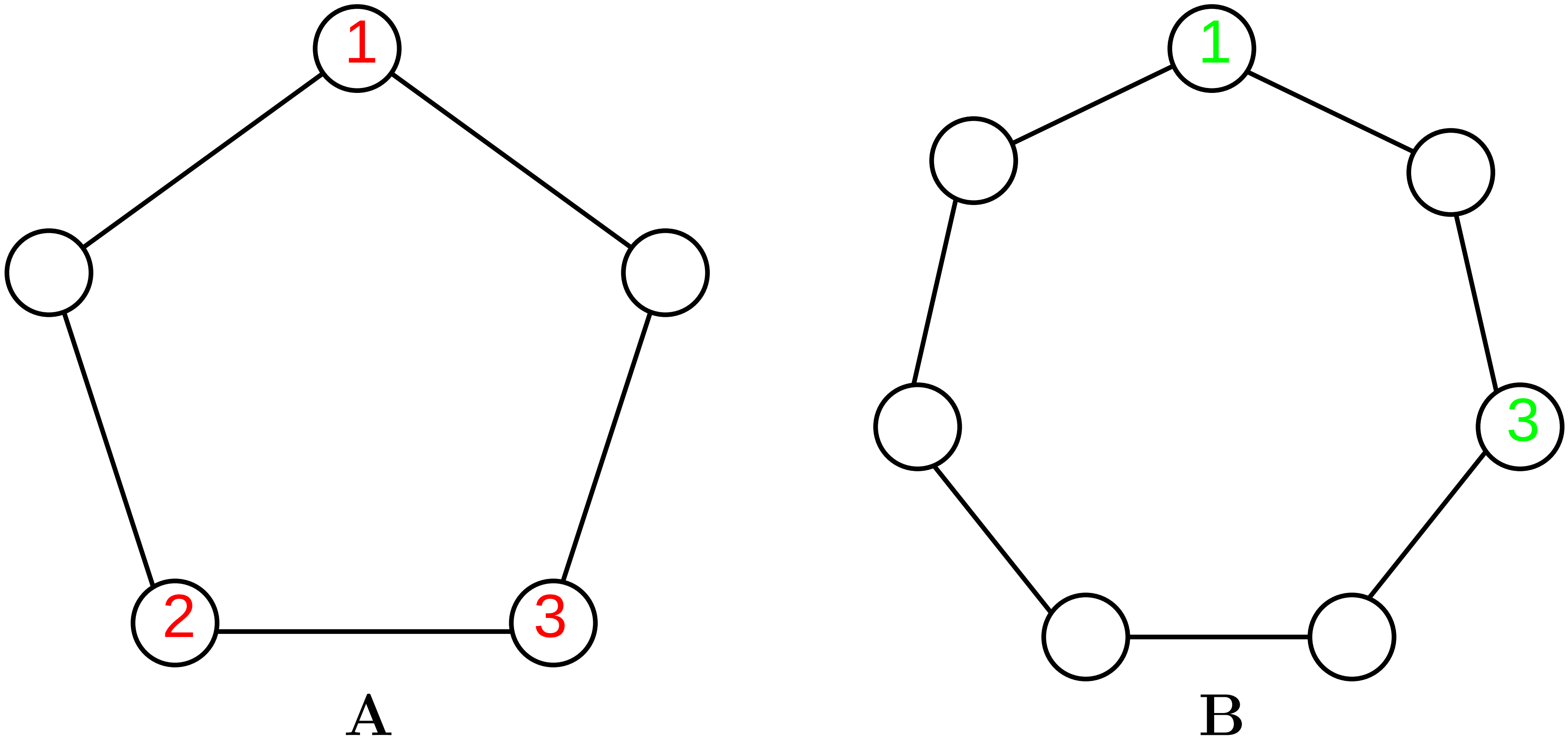
Duplicator wins the 2-pebble game
Spoiler wins the 3-pebble game

Example



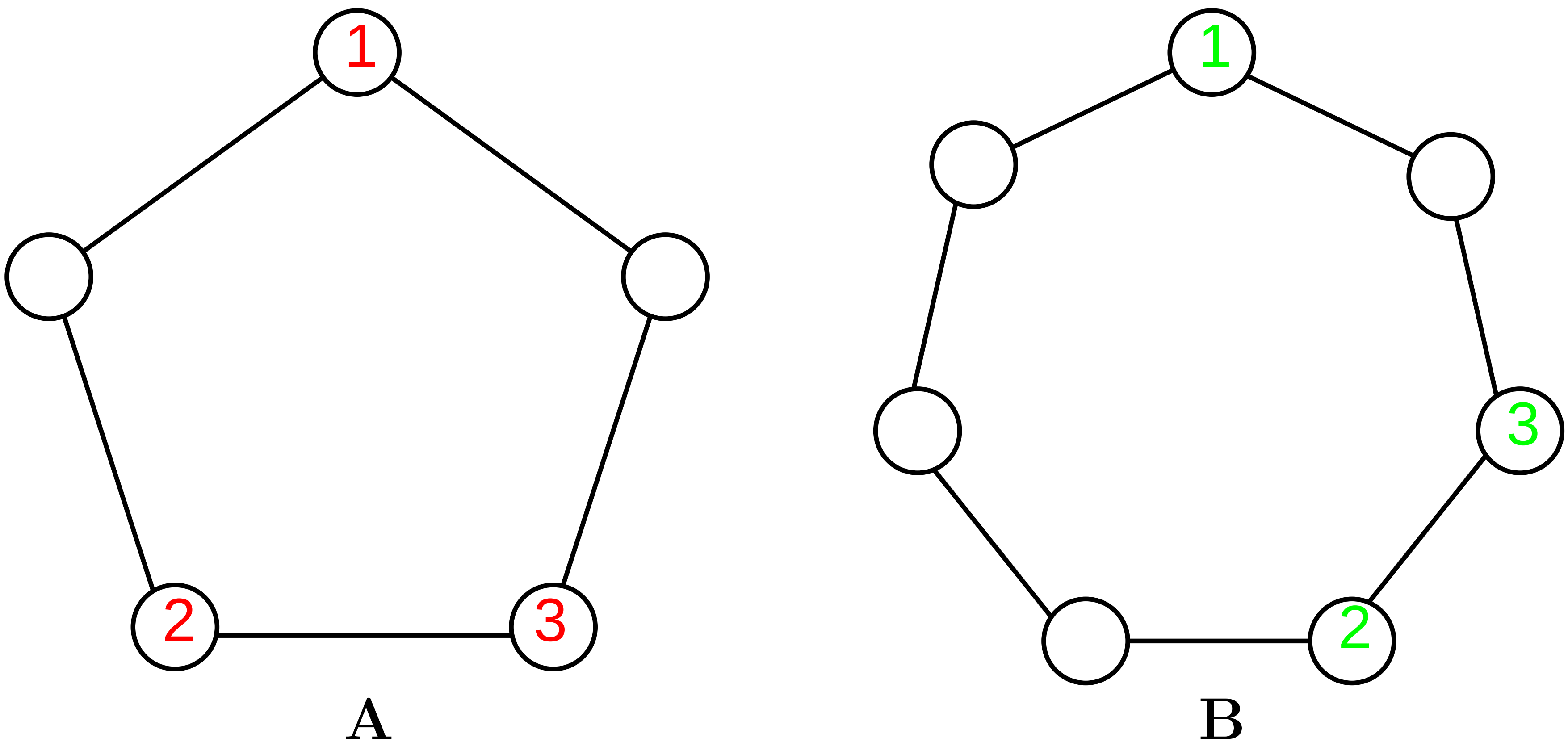
Duplicator wins the 2-pebble game
Spoiler wins the 3-pebble game

Example



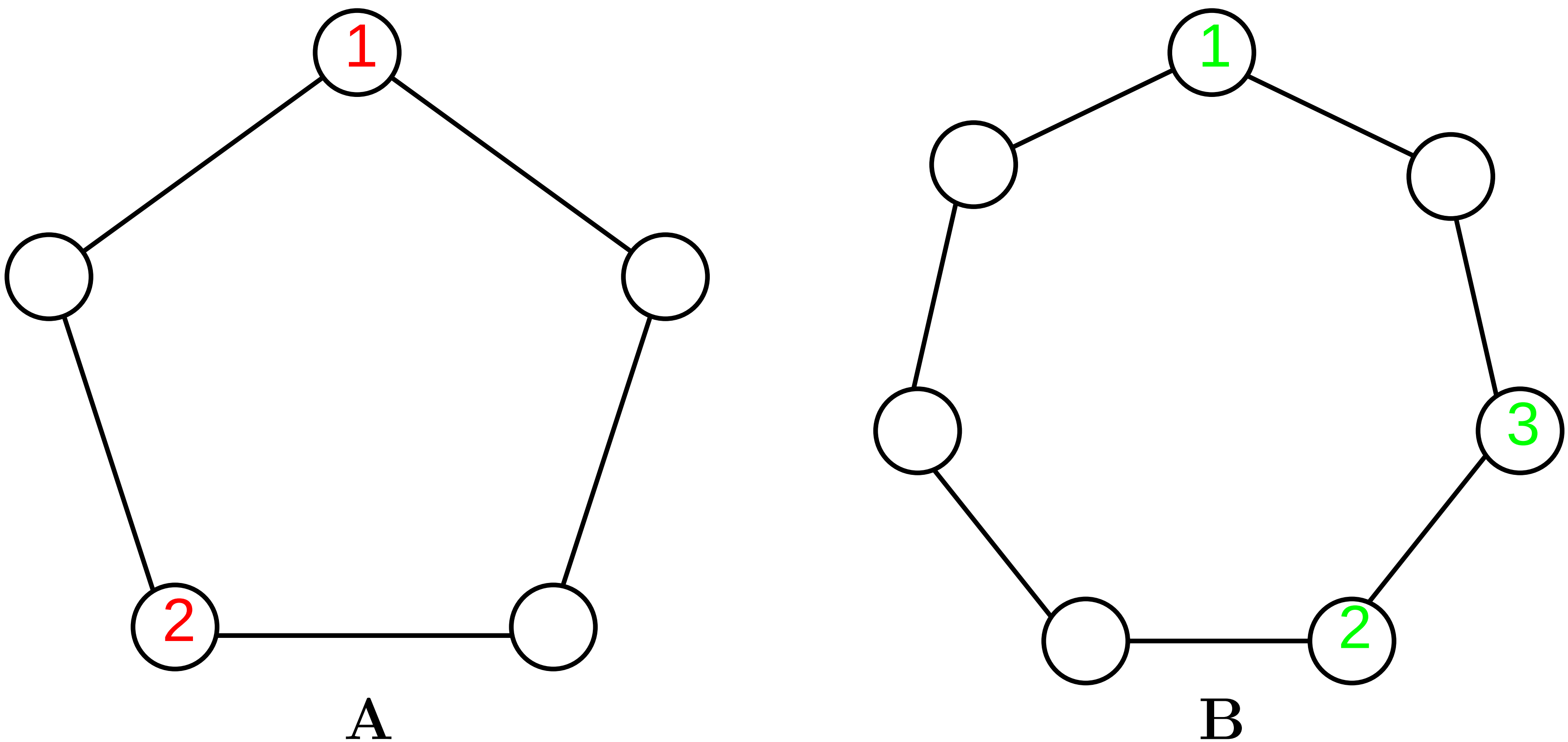
Duplicator wins the 2-pebble game
Spoiler wins the 3-pebble game

Example



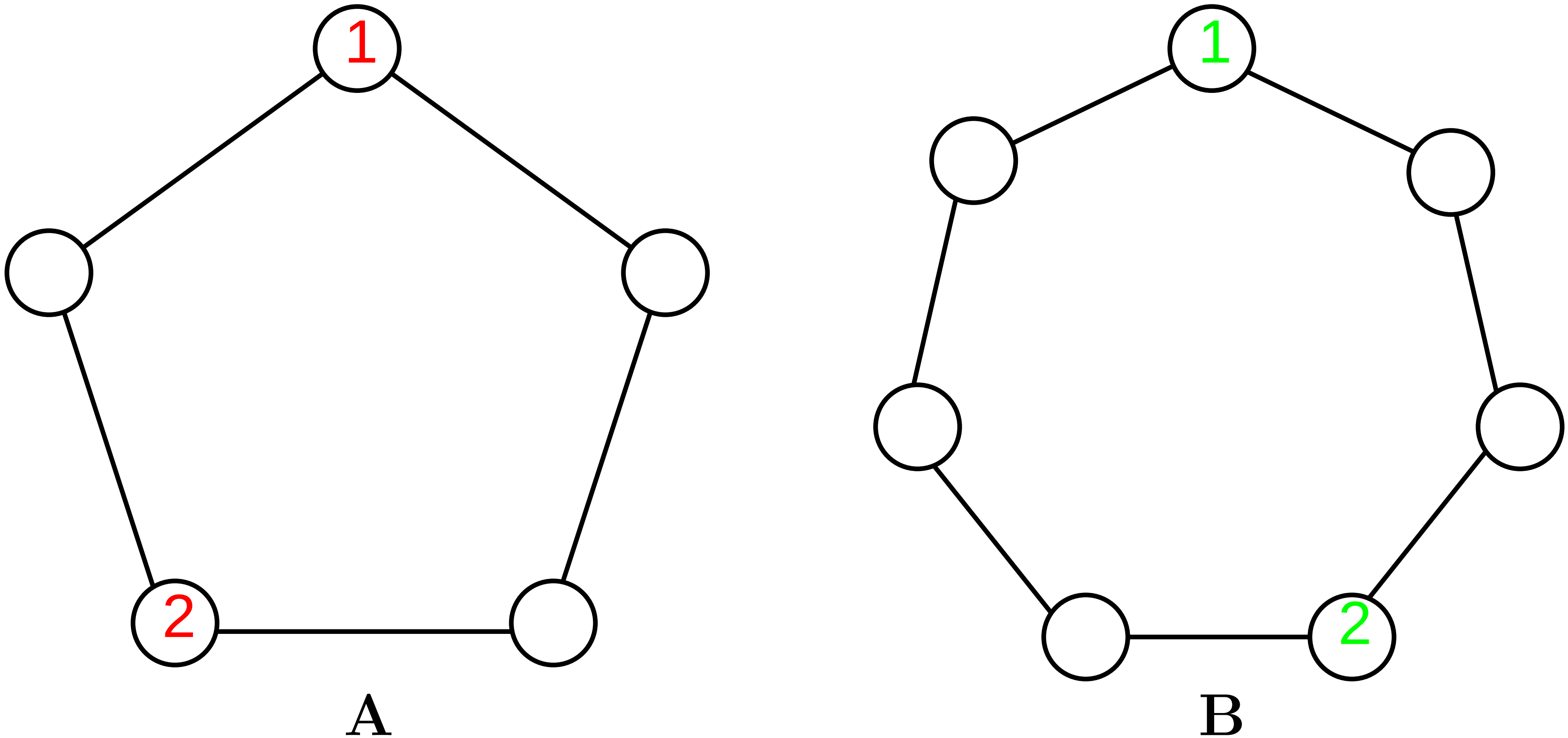
Duplicator wins the 2-pebble game
Spoiler wins the 3-pebble game

Example



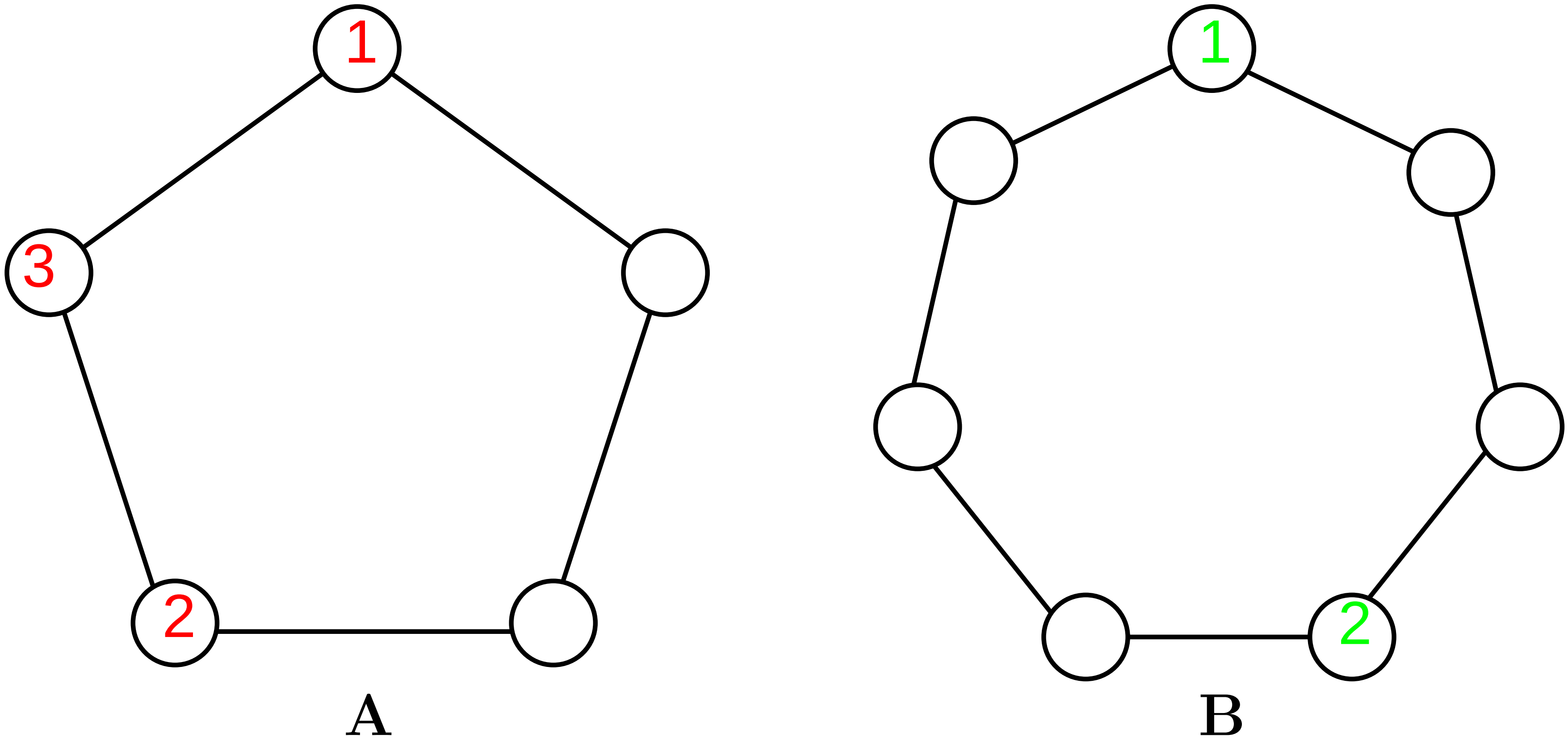
Duplicator wins the 2-pebble game
Spoiler wins the 3-pebble game

Example



Duplicator wins the 2-pebble game
Spoiler wins the 3-pebble game

Example



Duplicator wins the 2-pebble game
Spoiler wins the 3-pebble game

[Feder,Vardi 93,98, Kolaitis,Vardi 95,00]

Fact: Let $\mathbf{A}$ and $\mathbf{B}$ be structures. The following are equivalent:

- Duplicator wins the $(\exists, k)$-pebble game on $\mathbf{A}$ and $\mathbf{B}$

[Feder,Vardi 93,98, Kolaitis,Vardi 95,00]

Fact: Let $\mathbf{A}$ and $\mathbf{B}$ be structures. The following are equivalent:

- Duplicator wins the $(\exists, k)$-pebble game on $\mathbf{A}$ and $\mathbf{B}$
- For every formula φ in $\exists L^k_{\infty\omega}$

$$\mathbf{A} \models \varphi \;\;\Rightarrow\;\; \mathbf{B} \models \varphi$$

[Feder,Vardi 93,98, Kolaitis,Vardi 95,00]

Fact: Let $\mathbf{A}$ and $\mathbf{B}$ be structures. The following are equivalent:

- Duplicator wins the $(\exists, k)$-pebble game on $\mathbf{A}$ and $\mathbf{B}$
- For every formula φ in $\exists L^k_{\infty\omega}$

$$\mathbf{A} \models \varphi \quad \Rightarrow \quad \mathbf{B} \models \varphi$$

- For every struct. $\mathbf{T}$ with treewidth$(\mathbf{T}) < k$,

$$\mathbf{T} \rightarrow \mathbf{A} \quad \Rightarrow \quad \mathbf{T} \rightarrow \mathbf{B}$$

[Feder,Vardi 93,98, Kolaitis,Vardi 95,00]

Fact: Let $\mathbf{A}$ and $\mathbf{B}$ be structures. The following are equivalent:

- Duplicator wins the $(\exists, k)$-pebble game on $\mathbf{A}$ and $\mathbf{B}$
- For every formula φ in $\exists L^k_{\infty\omega}$

$$\mathbf{A} \models \varphi \quad \Rightarrow \quad \mathbf{B} \models \varphi$$

- For every struct. $\mathbf{T}$ with treewidth$(\mathbf{T}) < k$,

$$\mathbf{T} \to \mathbf{A} \quad \Rightarrow \quad \mathbf{T} \to \mathbf{B}$$

- Strong k-consistency can be established on $\mathbf{A}$ and $\mathbf{B}$

Algebraic definition:

A winning strategy for the Duplicator in the $(\exists, k)$-pebble game is a (non-empty) set $\mathcal{H}$ of partial homomorphisms such that

- If $f \in \mathcal{H}$ and $h \subseteq f$ then $h \in \mathcal{H}$ ($\mathcal{H}$ is *closed under subfunctions*)
- If $f \in \mathcal{H}$ then for every $a \in A$ such that $|\operatorname{dom}(f) \cup \{a\}| \leq k$ there is g with $f \subseteq g$ and $a \in \operatorname{dom}(g)$ ($\mathcal{H}$ has the *forth property up to k*)

Algebraic definition:

A winning strategy for the Duplicator in the $(\exists, k)$-pebble game is a (non-empty) set $\mathcal{H}$ of partial homomorphisms such that

- If $f \in \mathcal{H}$ and $h \subseteq f$ then $h \in \mathcal{H}$
  ($\mathcal{H}$ is *closed under subfunctions*)
- If $f \in \mathcal{H}$ then for every $a \in A$ such that $|\operatorname{dom}(f) \cup \{a\}| \leq k$ there is g with $f \subseteq g$ and $a \in \operatorname{dom}(g)$
  ($\mathcal{H}$ has the *forth property up to k*)

Intuitively: elements of $\mathcal{H}$ are winning positions for the Duplicator

For *fixed* k, there is a polynomial-time algorithm that computes a winning strategy (or determine that none exists)

For *fixed* k, there is a polynomial-time algorithm that computes a winning strategy (or determine that none exists)

- Initially place in $\mathcal{H}$ all partial homomorphisms with domain size at most k
  (there are $O(|A|^k|B|^k)$ of them)

For *fixed* k, there is a polynomial-time algorithm that computes a winning strategy (or determine that none exists)

- Initially place in $\mathcal{H}$ all partial homomorphisms with domain size at most k
(there are $O(|A|^k|B|^k)$ of them)
- Remove all $f \in \mathcal{H}$ not satisfying condition (2) of winning strategy until stabilizes

For *fixed* k, there is a polynomial-time algorithm that computes a winning strategy (or determine that none exists)

- Initially place in $\mathcal{H}$ all partial homomorphisms with domain size at most k
  (there are $O(|A|^k|B|^k)$ of them)
- Remove all $f \in \mathcal{H}$ not satisfying condition (2) of winning strategy until stabilizes

If k is also part of the input then deciding the existence of a winning strategy for the duplicator is EXPTIME-complete [Kolaitis, Panttaja 03]

Let $k > 0$ be *fixed*.

Observation:
For every structures $\mathbf{A}$ and $\mathbf{B}$

Spoiler wins the $(\exists, k)$-pebble game $\Rightarrow \mathbf{A} \not\to \mathbf{B}$

Let $k > 0$ be *fixed*.

Observation:

For every structures $\mathbf{A}$ and $\mathbf{B}$

Spoiler wins the $(\exists, k)$-pebble game $\Rightarrow \mathbf{A} \not\to \mathbf{B}$

The converse is not necessarily true

Let $k > 0$ be *fixed*.

Observation:
For every structures $\mathbf{A}$ and $\mathbf{B}$

$$\text{Spoiler wins the } (\exists, k)\text{-pebble game} \Rightarrow \mathbf{A} \not\rightarrow \mathbf{B}$$

The converse is not necessarily true

Question:
When is the converse true? That is,

under which circumstances deciding who wins the $(\exists, k)$*-pebble game is a sound and complete algorithm for the homomorphism problem?*

Looking at the left side

(Structural restrictions)

Let $\mathbf{A}$ be a structure.

Fact: [D., Kolaitis, Vardi, 02]
If treewidth(core($\mathbf{A}$))$< k$ then for every structure $\mathbf{B}$

Duplicator wins the $(\exists, k)$-pebble game $\Rightarrow \mathbf{A} \rightarrow \mathbf{B}$

Fact: [Atserias, Bulatov, D., 06]
If treewidth(core($\mathbf{A}$))$\geq k$ then there exists a structure $\mathbf{B}$ such that:

Duplicator wins the $(\exists, k)$-pebble game and $\mathbf{A} \not\rightarrow \mathbf{B}$

Complexity of CSP($\mathcal{C}$,All)

Let $\mathcal{C}$ be a set of structures.

Def: CSP($\mathcal{C}$,All) is the family of instances $\mathbf{A}$, $\mathbf{B}$ such that:
-$\mathbf{A} \in \mathcal{C}$ and
-$\mathbf{B}$ is arbitrary

Complexity of CSP($\mathcal{C}$,All)

Let $\mathcal{C}$ be a set of structures.

Def: CSP($\mathcal{C}$,All) is the family of instances $\mathbf{A}$, $\mathbf{B}$ such that:
-$\mathbf{A} \in \mathcal{C}$ and
-$\mathbf{B}$ is arbitrary

Fact:
core($\mathcal{C}$) has bounded treewidth $\Rightarrow$ CSP($\mathcal{C}$,All)$\in$PTIME

Note: core$(\mathcal{C}) = \{$core$(\mathbf{A}) : \mathbf{A} \in \mathcal{C}\}$

Complexity of CSP($\mathcal{C}$,All)

Let $\mathcal{C}$ be a set of structures.

Def: CSP($\mathcal{C}$,All) is the family of instances $\mathbf{A}$, $\mathbf{B}$ such that:
-$\mathbf{A} \in \mathcal{C}$ and
-$\mathbf{B}$ is arbitrary

Fact:
core($\mathcal{C}$) has bounded treewidth $\Rightarrow$ CSP($\mathcal{C}$,All)$\in$PTIME

Note: core$(\mathcal{C}) = \{$core$(\mathbf{A}) : \mathbf{A} \in \mathcal{C}\}$

Fact: [Grohe 03]
CSP($\mathcal{C}$,All)$\in$PTIME $\Rightarrow$ core($\mathcal{C}$) has bounded treewidth

(... under some assumptions:

FPT$\neq$W$[1]$, $\mathcal{C}$ is RE and of *bounded arity*)

Looking at the right side

(Language, template restrictions)

Def: $\mathbf{B}$ has width k if for every $\mathbf{A}$
Duplicator wins the k-pebble game on $\mathbf{A}$ and $\mathbf{B} \Rightarrow \mathbf{A} \rightarrow \mathbf{B}$

Looking at the right side

(Language, template restrictions)

Def: $\mathbf{B}$ has width k if for every $\mathbf{A}$
Duplicator wins the k-pebble game on $\mathbf{A}$ and $\mathbf{B} \Rightarrow \mathbf{A} \rightarrow \mathbf{B}$

Fact: [Feder,Vardi 93,98, Kolatis, Vardi 00]
The following are equivalent:

- $\mathbf{B}$ has width k
- $\neg$CSP($\mathbf{B}$) is definable in k-datalog
- $\mathbf{B}$ has an obstruction set of treewidth $< k$

Example: 2-COLORABILITY = CSP($\mathbf{K_2}$)

Example: 2-COLORABILITY = CSP($\mathbf{K_2}$)

- Datalog Program for non-2-COLORABILITY

$$
\begin{aligned}
\text{OddPath}(X,Y) &\ :-\ E(X,Y) \\
\text{OddPath}(X,Y) &\ :-\ \text{OddPath}(X,Z), E(Z,W), E(W,Y) \\
\text{Non2Colorable} &\ :-\ \text{OddPath}(X,X)
\end{aligned}
$$

Example: 2-COLORABILITY = CSP($\mathbf{K_2}$)

- Datalog Program for non-2-COLORABILITY

$$\begin{array}{lcl} \text{OddPath}(X,Y) & :- & E(X,Y) \\ \text{OddPath}(X,Y) & :- & \text{OddPath}(X,Z), E(Z,W), E(W,Y) \\ \text{Non2Colorable} & :- & \text{OddPath}(X,X) \end{array}$$

- An obstruction set for $\mathbf{K_2}$ is $\mathcal{O} = \{\mathbf{C_3}, \mathbf{C_5}, \ldots\}$.

That is, for every $\mathbf{A}$,

$$\mathbf{A} \rightarrow \mathbf{K_2} \Leftrightarrow \forall \mathbf{C} \in \mathcal{O} \;\; \mathbf{C} \not\rightarrow \mathbf{A}$$

Question:

Determine, for every k, which structures have width k

Question:
Determine, for every k, which structures have width k

- Not complete answer.

Question:

Determine, for every k, which structures have width k

- Not complete answer.
- Indeed, not even fully agreement on whether width is the *right* notion

Question:

Determine, for every k, which structures have width k

- Not complete answer.
- Indeed, not even fully agreement on whether width is the *right* notion
- Several alternative parameterizations have been proposed:

  relational width, (j, k)-width,...

Question:

Determine, for every k, which structures have width k

- Not complete answer.
- Indeed, not even fully agreement on whether width is the *right* notion
- Several alternative parameterizations have been proposed:

  relational width, (j, k)-width,...

Def:

$\mathbf{B}$ has bounded width if it has width k for some k

Sufficient conditions

Fact: $\mathbf{B}$ has bounded width if:

- $\mathbf{B}$ has a set function [Feder,Vardi 93,98]
- $\mathbf{B}$ has an extended set function [Chen, D., 04]
- $\mathbf{B}$ is invariant under a near-unanimity operation [Feder,Vardi 93,98][Jeavons, Cohen, Cooper 97]
- $\mathbf{B}$ is invariant under a 2-semilattice [Bulatov 02]
- $\mathbf{B}$ belong to certain classes of the known partial classification results [Bulatov 02,03,04]
- $\mathbf{B}$ has bounded treewidth duality [Hell, Zhu 94][Hell,Zhu 95][Hell, Nešetřil, Zhu 96]...
- $\vdots$

Necessary conditions

Fact: [Bulatov 04][Larose, Zádori 06]

If $\mathbf{B}$ has bounded width then $\mathcal{V}(\mathcal{A}(\mathbf{B}))$ omits types 1 and 2

Conjecture:

The converse is true

Inside $(\exists, k)$-peble games

- Observation: For some structures $\mathbf{B}$ with bounded width, CSP($\mathbf{B}$) is solvable in NLOGSPACE

Examples: 2-COLORABILITY, 2-SAT, 0/1/all constraints,...

- What do these examples have in common?

All of them have an obstruction set of bounded pathwidth

Example: 2-COLORABILITY = CSP($\mathbf{K_2}$)
For every graph $\mathbf{G}$

$$\mathbf{G} \rightarrow \mathbf{K_2} \text{ iff for every odd cicle } \mathbf{C},\ \mathbf{C} \not\rightarrow \mathbf{G}$$

Let $\mathbf{A}$ and $\mathbf{B}$ be structures.

Recall that the following are equivalent

- Duplicator wins the $(\exists, k)$-pebble game on $\mathbf{A}$ and $\mathbf{B}$
- For every $\mathbf{T}$ with treewidth($\mathbf{T}$)$< k$

$$\mathbf{T} \rightarrow \mathbf{A} \Rightarrow \mathbf{T} \rightarrow \mathbf{B}$$

Let $\mathbf{A}$ and $\mathbf{B}$ be structures.

Recall that the following are equivalent

- Duplicator wins the $(\exists, k)$-pebble game on $\mathbf{A}$ and $\mathbf{B}$
- For every $\mathbf{T}$ with treewidth($\mathbf{T}$)$< k$

$$\mathbf{T} \rightarrow \mathbf{A} \Rightarrow \mathbf{T} \rightarrow \mathbf{B}$$

We want to define the right game so that the following are equivalent

- Duplicator wins the ? game on $\mathbf{A}$ and $\mathbf{B}$
- For every $\mathbf{C}$ with pathwidth($\mathbf{C}$)$< k$

$$\mathbf{C} \rightarrow \mathbf{A} \Rightarrow \mathbf{C} \rightarrow \mathbf{B}$$

k-Pebble-Relation Game

Intuition: At each round of the game Duplicator does not need to commit.

- In the $(\exists, k)$-pebble game a configuration defines a partial homomorphism f.
  - Spoiler decides the domain of f
  - Duplicator decides the actual mapping f

k-Pebble-Relation Game

Intuition: At each round of the game Duplicator does not need to commit.

- In the $(\exists, k)$-pebble game a configuration defines a partial homomorphism f.
  - Spoiler decides the domain of f
  - Duplicator decides the actual mapping f
- In the k-pebble relation game, a configuration defines a *set* of partial homomorphisms with identical domain
  - Spoiler decides the domain
  - Duplicator defines the mappings
- Restriction: Duplicator can only extend *existing* mappings.

Fact:
The following are equivalent:

- Duplicator wins the k-pebble-relation game on $\mathbf{A}$ and $\mathbf{B}$
- For every $\mathbf{C}$ with pathwidth($\mathbf{C}$)$< k$,

$$\mathbf{C} \rightarrow \mathbf{A} \Rightarrow \mathbf{C} \rightarrow \mathbf{B}$$

Definition:

An structure $\mathbf{B}$ has k-pathwidth duality if for every $\mathbf{A}$

Duplicator wins the k-PR game on $\mathbf{A}$ and $\mathbf{B} \Rightarrow \mathbf{A} \rightarrow \mathbf{B}$

Fact: [D. 05]

Let $\mathbf{B}$ be an structure. The following are equivalent

- $\mathbf{B}$ has k-pathwidth duality
- $\mathbf{B}$ has an obstruction set of patwidth $< k$
- $\neg CSP(\mathbf{B})$ is definable in *linear* k-datalog.

Definition: A datalog program is linear if it has at most one IDB in the body each rule

Example: Datalog Program for non-2-COLORABILITY

$$\begin{aligned}
\text{OddPath}(X,Y) &\ :- \ E(X,Y) \\
\text{OddPath}(X,Y) &\ :- \ \text{OddPath}(X,Z), E(Z,W), E(W,Y) \\
\text{Non2Colorable} &\ :- \ \text{OddPath}(X,X)
\end{aligned}$$

Definition:
An structure $\mathbf{B}$ has k-pathwidth duality if for every $\mathbf{A}$
Duplicator wins the k-PR game on $\mathbf{A}$ and $\mathbf{B} \Rightarrow \mathbf{A} \to \mathbf{B}$

Fact: [D. 05]
Let $\mathbf{B}$ be an structure. The following are equivalent

- $\mathbf{B}$ has k-pathwidth duality
- $\mathbf{B}$ has an obstruction set of patwidth $< k$
- $\neg CSP(\mathbf{B})$ is definable in *linear* k-datalog.

Consequence:

$\mathbf{B}$ has bounded pathwidth $\Rightarrow$ CSP($\mathbf{B}$)$\in$ NL

Definition:

An structure $\mathbf{B}$ has k-pathwidth duality if for every $\mathbf{A}$

Duplicator wins the k-PR game on $\mathbf{A}$ and $\mathbf{B} \Rightarrow \mathbf{A} \rightarrow \mathbf{B}$

Fact: [D. 05]

Let $\mathbf{B}$ be an structure. The following are equivalent

- $\mathbf{B}$ has k-pathwidth duality
- $\mathbf{B}$ has an obstruction set of patwidth $< k$
- $\neg CSP(\mathbf{B})$ is definable in *linear* k-datalog.

Consequence:

$\mathbf{B}$ has bounded pathwidth $\Rightarrow$ CSP($\mathbf{B}$)$\in$ NL

Open Question: Is the converse also true?

Question:
Which structures have bounded pathwidth duality?

Fact: An structure $\mathbf{B}$ has bounded pathwidth duality if

- $\mathbf{B}$ is a poset invariant under a near-unanimity operation [Krokhin, Larose 03]
- $\mathbf{B}$ is invariant under a majority operation [D., Krokhin, 06]

Question:
Which structures have bounded pathwidth duality?

Fact: An structure $\mathbb{B}$ has bounded pathwidth duality if

- $\mathbb{B}$ is a poset invariant under a near-unanimity operation [Krokhin, Larose 03]
- $\mathbb{B}$ is invariant under a majority operation [D., Krokhin, 06]

Open question: Is it true that every $\mathbb{B}$ invariant under a near-unanimity operation has bounded pathwidth duality.

Remark: There are structures with bounded pathwidth duality *not* invariant under a near-unanimity operation [Krokhin, Larose 03]

Beyond $(\exists, k)$-pebble games

Motivation: Structural restrictions with unbounded arity

- Many results on structural restrictions e.g. [Gyssens, Jeavons, Cohen '94], [Gottlob, Leone, Scarcello '00, '01, '03], [Cohen, Jeavons, Gyssens '05]
- Note: Unbounded arity $\Rightarrow$ unbounded treewidth
- *Bounded hypertree width* [Gottlob, Leone, Scarcello, journal paper '03] subsumes every other decompostion method.
- Recently [Grohe, Marx 06, next talk] have found a new structural restriction incomparable with hypertree width

Beyond $(\exists, k)$-pebble games

Motivation: Structural restrictions with unbounded arity

- Many results on structural restrictions e.g. [Gyssens, Jeavons, Cohen ’94], [Gottlob, Leone, Scarcello ’00, ’01, ’03], [Cohen, Jeavons, Gyssens ’05]
- Note: Unbounded arity $\Rightarrow$ unbounded treewidth
- *Bounded hypertree width* [Gottlob, Leone, Scarcello, journal paper ’03] subsumes every other decompostion method.
- Recently [Grohe, Marx 06, next talk] have found a new structural restriction incomparable with hypertree width

Can games see anything meaningful on this?

k-cover game

[Chen, D. 05]
Def: A k-cover of $\mathbf{A}$ is a union $\mathrm{var}(\overline{t_1}) \cup \cdots \cup \mathrm{var}(\overline{t_k})$ of the variables of k tuples $\overline{t_1}, \ldots, \overline{t_k}$ of $\mathbf{A}$

The k-cover game is defined as the $(\exists, k)$-pebble game with some differences:

- The players have an infinite supply of pebbles
- Spoiler can place a new pebble only if the elements pebbled (after placing it) are entirely contained in a k-union.

Note: Duplicator wins the k-cover game $\Rightarrow$ Duplicator wins the $(\exists, k)$-pebble game

Fact:
The following are equivalent:

- Duplicator wins the k-cover game on $\mathbf{A}$ and $\mathbf{B}$
- For every $\mathbf{T}$ with generalized hypertree width $\leq k$,

$$\mathbf{T} \to \mathbf{A} \Rightarrow \mathbf{T} \to \mathbf{B}$$

Consequence:
Let $\mathbf{A}$ with ghw(core($\mathbf{A}$))$\leq k$. Then the following are equivalent

- Duplicator wins the k-cover game on $\mathbf{A}$ and $\mathbf{B}$
- $\mathbf{A} \to \mathbf{B}$

Again on the left side

Fact:
For fixed k, there is a polynomial-time algorithm that computes a winning strategy for the Duplicator (or determine that none exists).

Consequence:
CSP($\mathcal{C}$,All) is solvable in polynomial time if core($\mathcal{C}$) has bounded generalized hypertree width.

Other applications of games on CSP and related problems

- Quantified CSP [Chen, D. 05]
- CSP with infinite templates [Bodirsky, D. 06]
- Resolution width [Atserias, D. 03]