R2.07 Graphes

Corentin Dufourg, Régis Fleurquin et Thibault Godin
IUT de Vannes Informatique

Organisation de week-ends

Un groupe d'amis a comme loisirs

- Alaric : age of empire et théâtre
- Gaspard : théâtre, musées et festivals
- Lily : escalade et surf ;
- Pierre : escalade, festival et age of empire
- Timothée : festivals, rugby et surf
- Valentin : festivals, musées et rugby
- Yoann : festivals et musées

Organisation de week-ends

Un groupe d'amis a comme loisirs

- Alaric : age of empire et théâtre
- Gaspard : théâtre, musées et festivals
- Lily : escalade et surf ;
- Pierre : escalade, festival et age of empire
- Timothée : festivals, rugby et surf
- Valentin : festivals, musées et rugby
- Yoann : festivals et musées

Objectif : faire le plus d'activités ensemble sur le mininum de week-ends (pour pouvoir recommencer plus souvent.

Organisation de week-ends

Un groupe d'amis a comme loisirs

- Alaric : age of empire et théâtre
- Gaspard : théâtre, musées et festivals
- Lily : escalade et surf ;
- Pierre : escalade, festival et age of empire
- Timothée : festivals, rugby et surf
- Valentin : festivals, musées et rugby
- Yoann : festivals et musées

Objectif : faire le plus d'activités ensemble sur le mininum de week-ends (pour pouvoir recommencer plus souvent.

Solution naïve : un week-end par activité
$\rightsquigarrow$ **7** week-ends

Organisation de week-ends

Un groupe d'amis a comme loisirs

- Alaric : age of empire et théâtre
- Gaspard : théâtre, musées et festivals
- Lily : escalade et surf ;
- Pierre : escalade, festival et age of empire
- Timothée : festivals, rugby et surf
- Valentin : festivals, musées et rugby
- Yoann : festivals et musées

Objectif : faire le plus d'activités ensemble sur le mininum de week-ends (pour pouvoir recommencer plus souvent.

Solution naïve : un week-end par activité

$\rightsquigarrow$ **7** week-ends

Comment faire mieux ?

Organisation de week-ends

Un groupe d'amis a comme loisirs

- Alaric : age of empire et théâtre
- Gaspard : théâtre, musées et festivals
- Lily : escalade et surf ;
- Pierre : escalade, festival et age of empire
- Timothée : festivals, rugby et surf
- Valentin : festivals, musées et rugby
- Yoann : festivals et musées

Objectif : faire le plus d'activités ensemble sur le mininum de week-ends (pour pouvoir recommencer plus souvent.

Solution naïve : un week-end par activité

$\rightsquigarrow$ **7** week-ends

Comment faire mieux ?

Certaines activités peuvent avoir lieu en même temps puisque personne ne veut les faire en même temps

Organisation de week-ends

Un groupe d'amis a comme loisirs

- Alaric : age of empire et théâtre
- Gaspard : théâtre, musées et festivals
- Lily : escalade et surf ;
- Pierre : escalade, festival et age of empire
- Timothée : festivals, rugby et surf
- Valentin : festivals, musées et rugby
- Yoann : festivals et musées

Objectif : faire le plus d'activités ensemble sur le mininum de week-ends (pour pouvoir recommencer plus souvent.

Solution naïve : un week-end par activité

$\rightsquigarrow$ **7** week-ends

Comment faire mieux ?

Certaines activités peuvent avoir lieu en même temps puisque personne ne veut les faire en même temps

exemple : Théatre et surf en même temps

$\rightsquigarrow$ **6** week-ends

Première modélisation

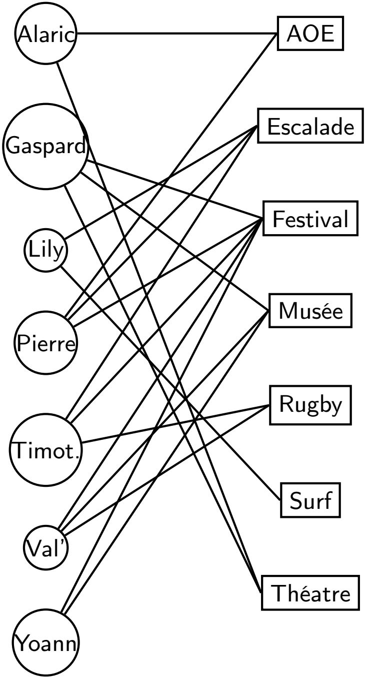
Un groupe d'amis a comme loisirs

- Alaric : age of empire et théâtre
- Gaspard : théâtre, musées et festivals
- Lily : escalade et surf ;
- Pierre : escalade, festival et age of empire
- Timothée : festivals, rugby et surf
- Valentin : festivals, musées et rugby
- Yoann : festivals et musées

Première modélisation

Un groupe d'amis a comme loisirs

- Alaric : age of empire et théâtre
- Gaspard : théâtre, musées et festivals
- Lily : escalade et surf ;
- Pierre : escalade, festival et age of empire
- Timothée : festivals, rugby et surf
- Valentin : festivals, musées et rugby
- Yoann : festivals et musées



Seconde modélisation

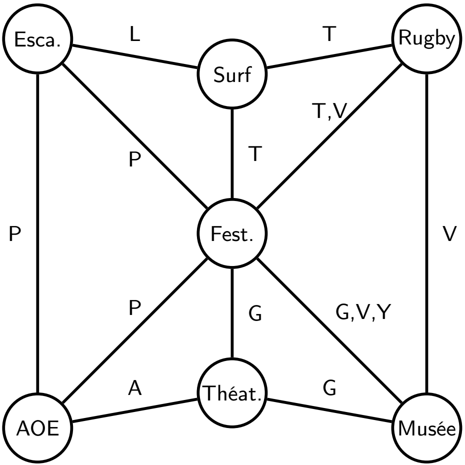
Un groupe d'amis a comme loisirs

- Alaric : age of empire et théâtre
- Gaspard : théâtre, musées et festivals
- Lily : escalade et surf ;
- Pierre : escalade, festival et age of empire
- Timothée : festivals, rugby et surf
- Valentin : festivals, musées et rugby
- Yoann : festivals et musées

Seconde modélisation

Un groupe d'amis a comme loisirs

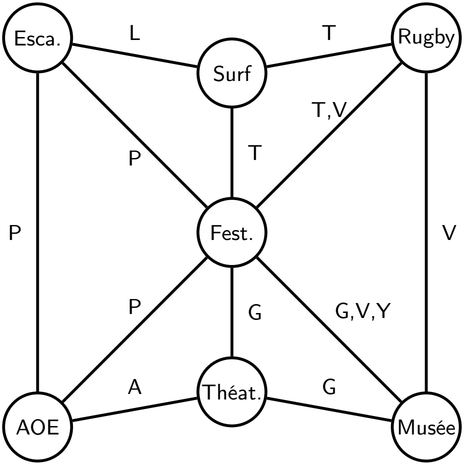
- Alaric : age of empire et théâtre
- Gaspard : théâtre, musées et festivals
- Lily : escalade et surf ;
- Pierre : escalade, festival et age of empire
- Timothée : festivals, rugby et surf
- Valentin : festivals, musées et rugby
- Yoann : festivals et musées



Seconde modélisation

Un groupe d'amis a comme loisirs

- Alaric : age of empire et théâtre
- Gaspard : théâtre, musées et festivals
- Lily : escalade et surf ;
- Pierre : escalade, festival et age of empire
- Timothée : festivals, rugby et surf
- Valentin : festivals, musées et rugby
- Yoann : festivals et musées

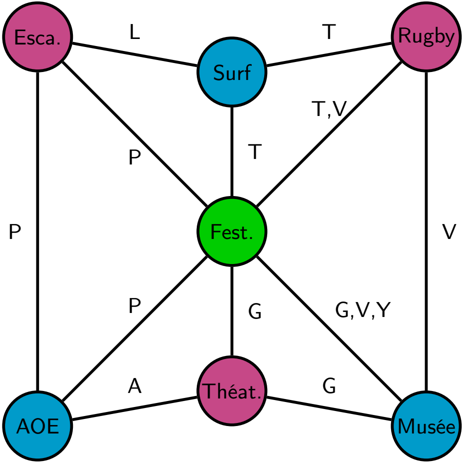


Deux activités qui intéressent la même personne ne peuvent pas avoir lieu le même week-end $\rightsquigarrow$ deux sommets voisins doivent avoir une couleur différentes

Seconde modélisation

Un groupe d'amis a comme loisirs

- Alaric : age of empire et théâtre
- Gaspard : théâtre, musées et festivals
- Lily : escalade et surf ;
- Pierre : escalade, festival et age of empire
- Timothée : festivals, rugby et surf
- Valentin : festivals, musées et rugby
- Yoann : festivals et musées

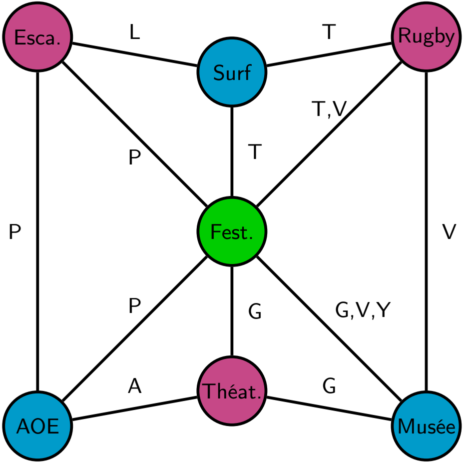


Deux activités qui intéressent la même personne ne peuvent pas avoir lieu le même week-end ⇝ deux sommets voisins doivent avoir une couleur différentes

Seconde modélisation

Un groupe d'amis a comme loisirs

- Alaric : age of empire et théâtre
- Gaspard : théâtre, musées et festivals
- Lily : escalade et surf ;
- Pierre : escalade, festival et age of empire
- Timothée : festivals, rugby et surf
- Valentin : festivals, musées et rugby
- Yoann : festivals et musées



Deux activités qui intéressent la même personne ne peuvent pas avoir lieu le même week-end $\rightsquigarrow$ deux sommets voisins doivent avoir une couleur différentes

$\rightsquigarrow$ **3** week-ends

Coloration de graphe

Dans cette partie du cours, tous les graphes seront simples et non-orientés

Coloration de graphe

Dans cette partie du cours, tous les graphes seront simples et non-orientés

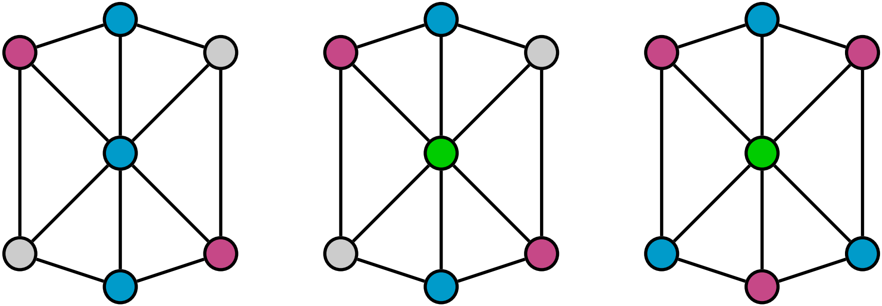
Dans cette partie du cours, tous les graphes seront simples et non-orientés

un *coloriage* est une fonction $c : S \to \mathbb{N}$

Dans cette partie du cours, tous les graphes seront simples et non-orientés

un *coloriage* est une fonction $c : S \to \mathbb{N}$

un *coloriage* est *valide* si deux sommets voisins n'ont pas la même couleur



Nombre Chromatique

Pour tout graphe G, il existe toujours un coloriage valide

Nombre Chromatique

Pour tout graphe G, il existe toujours un coloriage valide

Le prouver

Nombre Chromatique

Pour tout graphe G, il existe toujours un coloriage valide

✎ Le prouver

Nombre chromatique $\chi_G \rightsquigarrow$ nombre minimum de couleurs d'un coloriage valide du graphe G

Nombre Chromatique

Pour tout graphe G, il existe toujours un coloriage valide

📝 Le prouver

Nombre chromatique $\chi_G \rightsquigarrow$ nombre minimum de couleurs d'un coloriage valide du graphe G

Il existe des graphe ayant un nombre chromatique aussi grand que l'on veut.

📝 Le prouver

Exemples

Il est facile de déterminer le nombre chromatique de certains graphes classiques :

- graphe cyclique d'ordre n :χ_{C_n}
- graphe chemin d'ordre n :χ_{P_n}
- graphe étoile d'ordre n :χ_{S_n}
- graphe complet d'ordre n : χ_{K_n}
- G graphe biparti avec au moins une arête :χ_G
- G arbre avec au moins une arête : χ_G

Exemples

Il est facile de déterminer le nombre chromatique de certains graphes classiques :

- graphe cyclique d'ordre n :$\chi_{C_n} = 2$ si n pair et 3 si n impair ;
- graphe chemin d'ordre n :$\chi_{P_n} = 2$;
- graphe étoile d'ordre n :$\chi_{S_n} = 2$;
- graphe complet d'ordre n : $\chi_{K_n} = n$;
- G graphe biparti avec au moins une arête :$\chi_G = 2$; (en fait un graphe est 2- coloriable si et seulement s'il est biparti)
- G arbre avec au moins une arête : $\chi_G = 2$.

Soit G un graphe et G' un sous graphe, on a $\chi_{G'} \leq \chi_G$

Soit G un graphe et G' un sous graphe, on a $\chi_{G'} \leq \chi_G$

application : un graphe qui contient un $k-$clique nécessite au moins $k-1$ couleurs pour être colorié.

Soit G un graphe et G' un sous graphe, on a $\chi_{G'} \leq \chi_G$

application : un graphe qui contient un $k-$clique nécessite au moins $k-1$ couleurs pour être colorié.

remarque : la réciproque est fausse :

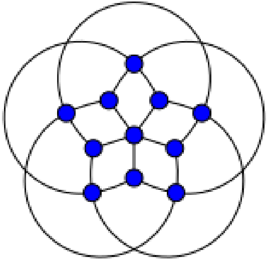


Figure – copyleft : David Epstein

Théorème de Brooks

Soit G un graphe et Δ son degré maximum. Alors le nombre chromatique χ_G vérifie $\chi(G) \leq \Delta$, sauf si G est un graphe complet ou un cycle de longueur impaire, auquel cas $\chi(G) = \Delta + 1$.

Plan

Coloration de graphes

- Algorithmes
- 4 couleurs

Jeux sur les graphes

Algorithmes

Algorithm 1: Coloration naïve

Data: graphe G=(S,A)

Result: coloriage c

for $s \in S$ **do**

  $c(s) \leftarrow$ plus petite couleur non utilisée par les voisins de s

Algorithmes

Algorithm 2: Coloration naïve

Data: graphe G=(S,A)

Result: coloriage c

for $s \in S$ **do**

  $c(s) \leftarrow$ plus petite couleur non utilisée par les voisins de s

Algorithmes

Algorithm 3: Coloration naïve

Data: graphe G=(S,A)

Result: coloriage c

for $s \in S$ **do**

  $c(s) \leftarrow$ plus petite couleur non utilisée par les voisins de s

---

Dépend fortement du choix de l'ordre

Algorithmes

Algorithm 4: Coloration naïve

Data: graphe G=(S,A)

Result: coloriage c

for $s \in S$ **do**

  $c(s) \leftarrow$ plus petite couleur non utilisée par les voisins de s

---

Dépend fortement du choix de l'ordre

Donne parfois (souvent) un coloriage sous-optimal

Algorithmes

Algorithm 5: Coloration naïve

Data: graphe G=(S,A)

Result: coloriage c

for $s \in S$ **do**

  $c(s) \leftarrow$ plus petite couleur non utilisée par les voisins de s

Dépend fortement du choix de l'ordre

Donne parfois (souvent) un coloriage sous-optimal

Au pire $\Delta(G) + 1$ couleurs ; très bonne complexité $O(|S|\Delta(G))$

Exemple

Algorithm 6: Algorithme de Welsh-Powell

Data: graphe G=(S,A)

Result: coloriage c

```
L ← liste des sommets ordonnés par degré
  décroissant ;
couleur-cour ← 0 ;
while L ≠ ∅ do
    couleur-cour ← couleur-cour + 1 ;
    s=L[0] ;
    c[s]=couleur-cour ;
    L ← L \ {s} ;
    V ← voisins de s ;
    for x ∈ L do
        if x ∉ V then
            c[x]=couleur-cour ;
            L ← L \ {x} ;
            V ← V∪ voisins de x ;
```

Algorithm 7: Algorithme de Welsh-Powell

Data: graphe G=(S,A)

Result: coloriage c

$L \leftarrow$ liste des sommets ordonnés par degré décroissant ;

couleur-cour $\leftarrow$ 0 ;

while $L \neq \emptyset$ **do**

    couleur-cour $\leftarrow$ couleur-cour + 1 ;

    s=L[0] ;

    c[s]=couleur-cour ;

    $L \leftarrow L \setminus \{s\}$;

    $V \leftarrow$ voisins de s ;

    for $x \in L$ **do**

        if $x \notin V$ **then**

            c[x]=couleur-cour ;

            $L \leftarrow L \setminus \{x\}$;

            $V \leftarrow V \cup$ voisins de x ;

Algorithm 8: Algorithme de Welsh-Powell

Data: graphe G=(S,A)

Result: coloriage c

$L \leftarrow$ liste des sommets ordonnés par degré décroissant ;

couleur-cour $\leftarrow 0$;

while $L \neq \emptyset$ **do**

  couleur-cour $\leftarrow$ couleur-cour $+ 1$;

  s=L[0] ;

  c[s]=couleur-cour ;

  $L \leftarrow L \setminus \{s\}$;

  $V \leftarrow$ voisins de s ;

  for $x \in L$ **do**

    if $x \notin V$ **then**

      c[x]=couleur-cour ;

      $L \leftarrow L \setminus \{x\}$;

      $V \leftarrow V \cup$ voisins de x ;

Indé du choix de l'ordre

Algorithm 9: Algorithme de Welsh-Powell

Data: graphe G=(S,A)

Result: coloriage c

$L \leftarrow$ liste des sommets ordonnés par degré décroissant ;

couleur-cour $\leftarrow$ 0 ;

while $L \neq \emptyset$ **do**

  couleur-cour $\leftarrow$ couleur-cour + 1 ;

  s=L[0] ;

  c[s]=couleur-cour ;

  $L \leftarrow L \setminus \{s\}$;

  $V \leftarrow$ voisins de s ;

  for $x \in L$ **do**

    if $x \notin V$ **then**

      c[x]=couleur-cour ;

      $L \leftarrow L \setminus \{x\}$;

      $V \leftarrow V \cup$ voisins de x ;

Indé du choix de l'ordre

Donne parfois ("rarement") un coloriage sous-optimal

Algorithm 10: Algorithme de Welsh-Powell

Data: graphe G=(S,A)

Result: coloriage c

$L \leftarrow$ liste des sommets ordonnés par degré décroissant ;

couleur-cour $\leftarrow$ 0 ;

while $L \neq \emptyset$ **do**

  couleur-cour $\leftarrow$ couleur-cour + 1 ;

  s=L[0] ;

  c[s]=couleur-cour ;

  $L \leftarrow L \setminus \{s\}$;

  $V \leftarrow$ voisins de s ;

  for $x \in L$ **do**

    if $x \notin V$ **then**

      c[x]=couleur-cour ;

      $L \leftarrow L \setminus \{x\}$;

      $V \leftarrow V\cup$ voisins de x ;

Indé du choix de l'ordre

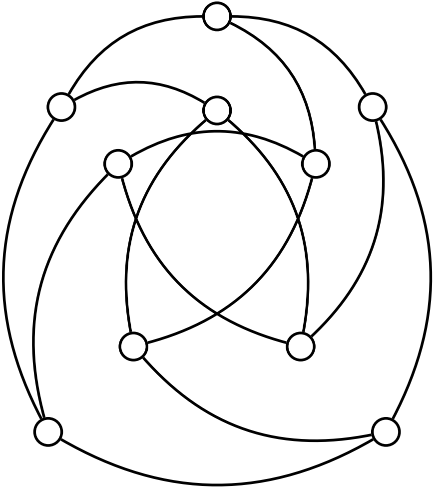
Donne parfois ("rarement") un coloriage sous-optimal

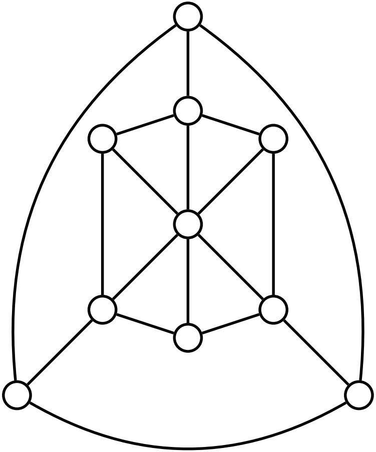
Au pire $\max_i \min\{\deg(s_i) + 1, i\}$, couleurs ; très bonne complexité $O(|S|\Delta(G))$

Exemple

Un coloriage à 1M d'euro (et même plus)

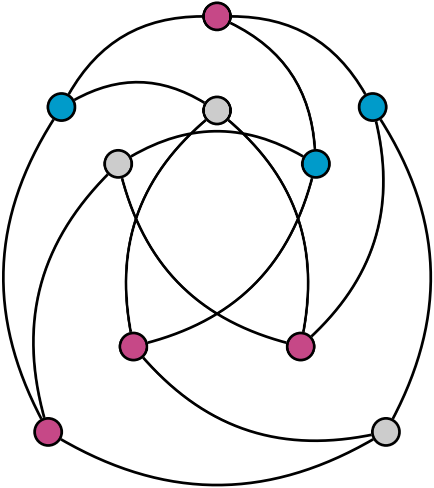
Donner un algorithme efficace (polynomial) le nombre chromatique d'un graphe.

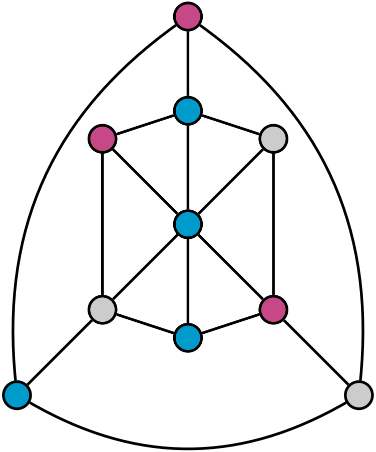




Un coloriage à 1M d'euro (et même plus)

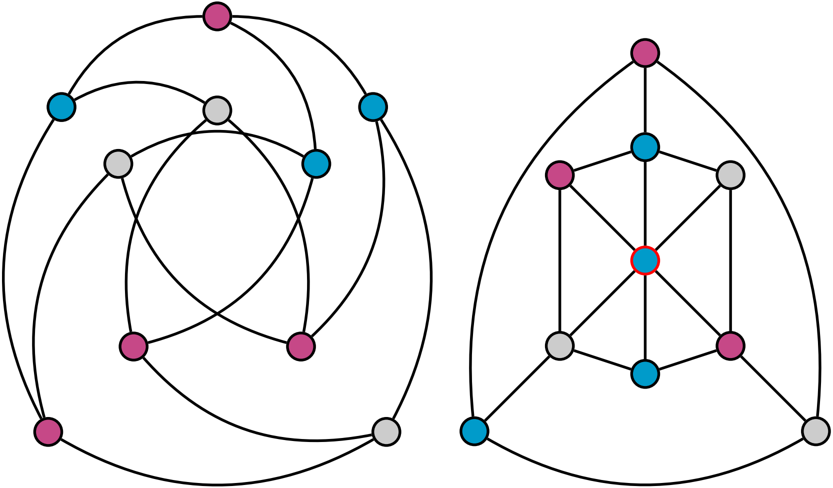
Donner un algorithme efficace (polynomial) le nombre chromatique d'un graphe.





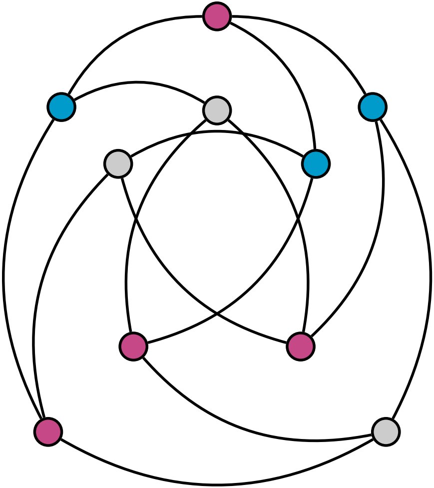
Un coloriage à 1M d'euro (et même plus)

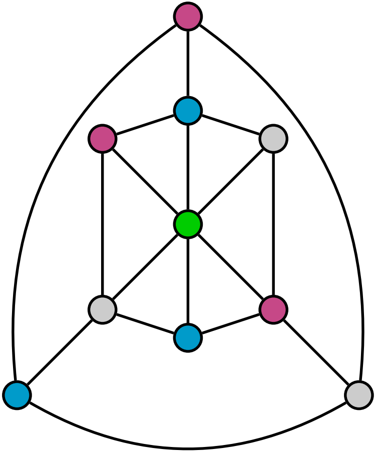
Donner un algorithme efficace (polynomial) le nombre chromatique d'un graphe.



Un coloriage à 1M d'euro (et même plus)

Donner un algorithme efficace (polynomial) le nombre chromatique d'un graphe.





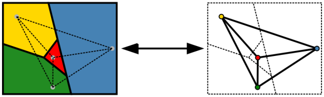
Plan

Théorème des 4 couleurs

Un graphe planaire peut-être colorié avec au plus 4 couleurs

Théorème des 4 couleurs

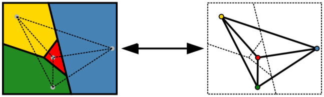
Un graphe planaire peut-être colorié avec au plus 4 couleurs

 source :

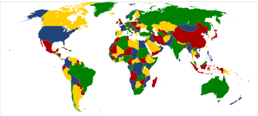
https://commons.wikimedia.org/wiki/File:Four_Colour_Planar_Graph.svg

Théorème des 4 couleurs

Un graphe planaire peut-être colorié avec au plus 4 couleurs

 source :

https://commons.wikimedia.org/wiki/File:Four_Colour_Planar_Graph.svg

 source :

https://commons.wikimedia.org/wiki/File:Four_Colour_Planar_Graph.svg

Théorème des 4 couleurs

Un graphe planaire peut-être colorié avec au plus 4 couleurs

Théorème des 4 couleurs

Un graphe planaire peut-être colorié avec au plus 4 couleurs

- **1852** Francis Guthrie $\rightsquigarrow$ *conjecture*

Théorème des 4 couleurs

Un graphe planaire peut-être colorié avec au plus 4 couleurs

- **1852** Francis Guthrie $\rightsquigarrow$ *conjecture*
- **1879** Alfred Kempe $\rightsquigarrow$ *preuve*

Théorème des 4 couleurs

Un graphe planaire peut-être colorié avec au plus 4 couleurs

- **1852** Francis Guthrie $\rightsquigarrow$ *conjecture*
- **1879** Alfred Kempe $\rightsquigarrow$ *preuve*
- **1880** Peter Guthrie Tait $\rightsquigarrow$ *preuve*

Théorème des 4 couleurs

Un graphe planaire peut-être colorié avec au plus 4 couleurs

- **1852** Francis Guthrie $\rightsquigarrow$ *conjecture*
- **1879** Alfred Kempe $\rightsquigarrow$ *preuve*
- **1880** Peter Guthrie Tait $\rightsquigarrow$ *preuve*
- **1890** Julius Petersen $\rightsquigarrow$ *invalidation*

Théorème des 4 couleurs

Un graphe planaire peut-être colorié avec au plus 4 couleurs

- **1852** Francis Guthrie $\rightsquigarrow$ *conjecture*
- **1879** Alfred Kempe $\rightsquigarrow$ *preuve*
- **1880** Peter Guthrie Tait $\rightsquigarrow$ *preuve*
- **1890** Julius Petersen $\rightsquigarrow$ *invalidation*
- **1891** Percy John Heawood $\rightsquigarrow$ *Théorème des 5 couleurs*

Théorème des 4 couleurs

Un graphe planaire peut-être colorié avec au plus 4 couleurs

- **1852** Francis Guthrie $\rightsquigarrow$ *conjecture*
- **1879** Alfred Kempe $\rightsquigarrow$ *preuve*
- **1880** Peter Guthrie Tait $\rightsquigarrow$ *preuve*
- **1890** Julius Petersen $\rightsquigarrow$ *invalidation*
- **1891** Percy John Heawood $\rightsquigarrow$ *Théorème des 5 couleurs*
- **1943** Hugo Hadwiger $\rightsquigarrow$ *conjecture plus générale*

Théorème des 4 couleurs

Un graphe planaire peut-être colorié avec au plus 4 couleurs

- **1852** Francis Guthrie $\rightsquigarrow$ *conjecture*
- **1879** Alfred Kempe $\rightsquigarrow$ *preuve*
- **1880** Peter Guthrie Tait $\rightsquigarrow$ *preuve*
- **1890** Julius Petersen $\rightsquigarrow$ *invalidation*
- **1891** Percy John Heawood $\rightsquigarrow$ *Théorème des 5 couleurs*
- **1943** Hugo Hadwiger $\rightsquigarrow$ *conjecture plus générale*
- **1976** Kenneth Appel et Wolfgang Haken $\rightsquigarrow$ *preuve*

Théorème des 4 couleurs

Un graphe planaire peut-être colorié avec au plus 4 couleurs

- **1852** Francis Guthrie $\rightsquigarrow$ *conjecture*
- **1879** Alfred Kempe $\rightsquigarrow$ *preuve*
- **1880** Peter Guthrie Tait $\rightsquigarrow$ *preuve*
- **1890** Julius Petersen $\rightsquigarrow$ *invalidation*
- **1891** Percy John Heawood $\rightsquigarrow$ *Théorème des 5 couleurs*
- **1943** Hugo Hadwiger $\rightsquigarrow$ *conjecture plus générale*
- **1976** Kenneth Appel et Wolfgang Haken $\rightsquigarrow$ *preuve*
- **1989** Kenneth Appel et Wolfgang Haken $\rightsquigarrow$ *preuve corrigée*

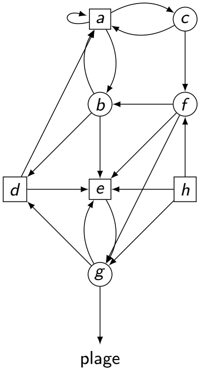
Théorème des 4 couleurs

Un graphe planaire peut-être colorié avec au plus 4 couleurs

- **1852** Francis Guthrie $\rightsquigarrow$ *conjecture*
- **1879** Alfred Kempe $\rightsquigarrow$ *preuve*
- **1880** Peter Guthrie Tait $\rightsquigarrow$ *preuve*
- **1890** Julius Petersen $\rightsquigarrow$ *invalidation*
- **1891** Percy John Heawood $\rightsquigarrow$ *Théorème des 5 couleurs*
- **1943** Hugo Hadwiger $\rightsquigarrow$ *conjecture plus générale*
- **1976** Kenneth Appel et Wolfgang Haken $\rightsquigarrow$ *preuve*
- **1989** Kenneth Appel et Wolfgang Haken $\rightsquigarrow$ *preuve corrigée*
- **1996** Neil Robertson, Daniel P. Sanders, Paul Seymour, et Robin Thomas $\rightsquigarrow$ *algo. quadratique*

Alice au pays des merveilles

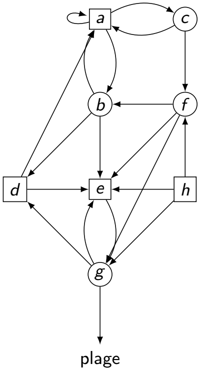
Alice souhaite aller à la plage, pour cela elle doit traverser la ville, mais Bob souhaite l'en empêcher, et contrôle certaines intersections :



Alice contrôle les sommets ronds, et Bob les sommets carrés.

Alice au pays des merveilles

Alice souhaite aller à la plage, pour cela elle doit traverser la ville, mais Bob souhaite l'en empêcher, et contrôle certaines intersections :



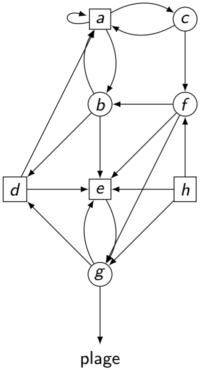
Alice contrôle les sommets ronds, et Bob les sommets carrés.

Alice peut-elle gagner si elle part de :

- c ?

Alice au pays des merveilles

Alice souhaite aller à la plage, pour cela elle doit traverser la ville, mais Bob souhaite l'en empêcher, et contrôle certaines intersections :



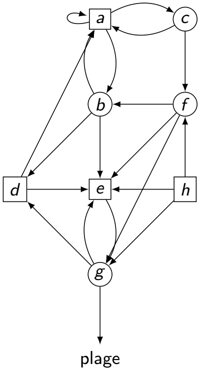
Alice contrôle les sommets ronds, et Bob les sommets carrés.

Alice peut-elle gagner si elle part de :

- c ?
- a ?

Alice au pays des merveilles

Alice souhaite aller à la plage, pour cela elle doit traverser la ville, mais Bob souhaite l'en empêcher, et contrôle certaines intersections :



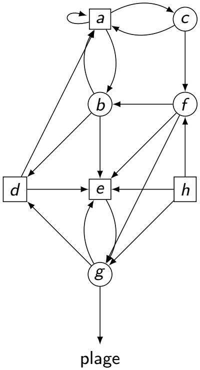
Alice contrôle les sommets ronds, et Bob les sommets carrés.

Alice peut-elle gagner si elle part de :

- c ?
- a ?
- g ?

Alice au pays des merveilles

Alice souhaite aller à la plage, pour cela elle doit traverser la ville, mais Bob souhaite l'en empêcher, et contrôle certaines intersections :



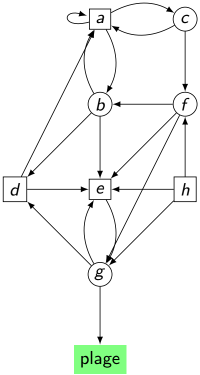
Alice contrôle les sommets ronds, et Bob les sommets carrés.

Alice peut-elle gagner si elle part de :

- c ?
- a ?
- g ?
- Plage ?

Alice au pays des merveilles

Alice souhaite aller à la plage, pour cela elle doit traverser la ville, mais Bob souhaite l'en empêcher, et contrôle certaines intersections :



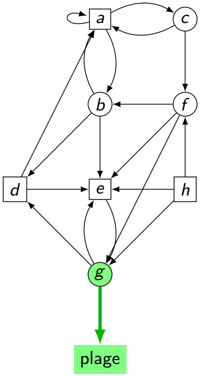
Alice contrôle les sommets ronds, et Bob les sommets carrés.

<u>Fact :</u> si Alice peut atteindre la plage depuis un sommet qu'elle contrôle, ce sommet est gagnant pour Alice

Si Bob est obligé de laisser Alice attendre la plage depuis un sommet qu'il contrôle, le sommet est gagnant pour Alice

Alice au pays des merveilles

Alice souhaite aller à la plage, pour cela elle doit traverser la ville, mais Bob souhaite l'en empêcher, et contrôle certaines intersections :



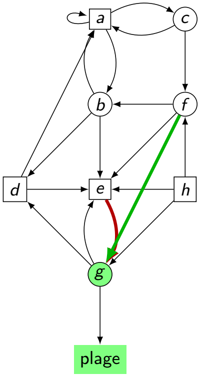
Alice contrôle les sommets ronds, et Bob les sommets carrés.

Fact : si Alice peut atteindre la plage depuis un sommet qu'elle contrôle, ce sommet est gagnant pour Alice

Si Bob est obligé de laisser Alice attendre la plage depuis un sommet qu'il contrôle, le sommet est gagnant pour Alice

Alice au pays des merveilles

Alice souhaite aller à la plage, pour cela elle doit traverser la ville, mais Bob souhaite l'en empêcher, et contrôle certaines intersections :



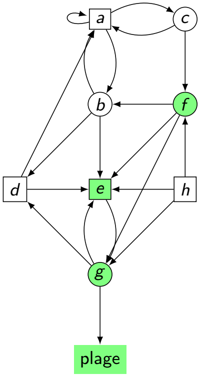
Alice contrôle les sommets ronds, et Bob les sommets carrés.

Fact : si Alice peut atteindre la plage depuis un sommet qu'elle contrôle, ce sommet est gagnant pour Alice

Si Bob est obligé de laisser Alice attendre la plage depuis un sommet qu'il contrôle, le sommet est gagnant pour Alice

Alice au pays des merveilles

Alice souhaite aller à la plage, pour cela elle doit traverser la ville, mais Bob souhaite l'en empêcher, et contrôle certaines intersections :



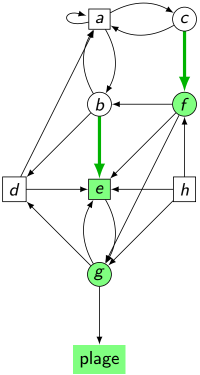
Alice contrôle les sommets ronds, et Bob les sommets carrés.

Fact : si Alice peut atteindre la plage depuis un sommet qu'elle contrôle, ce sommet est gagnant pour Alice

Si Bob est obligé de laisser Alice attendre la plage depuis un sommet qu'il contrôle, le sommet est gagnant pour Alice

Alice au pays des merveilles

Alice souhaite aller à la plage, pour cela elle doit traverser la ville, mais Bob souhaite l'en empêcher, et contrôle certaines intersections :



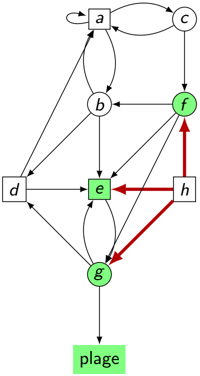
Alice contrôle les sommets ronds, et Bob les sommets carrés.

Fact : si Alice peut atteindre la plage depuis un sommet qu'elle contrôle, ce sommet est gagnant pour Alice

Si Bob est obligé de laisser Alice attendre la plage depuis un sommet qu'il contrôle, le sommet est gagnant pour Alice

Alice au pays des merveilles

Alice souhaite aller à la plage, pour cela elle doit traverser la ville, mais Bob souhaite l'en empêcher, et contrôle certaines intersections :



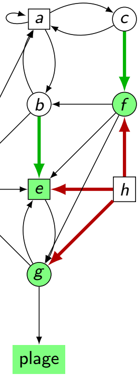
Alice contrôle les sommets ronds, et Bob les sommets carrés.

Fact : si Alice peut atteindre la plage depuis un sommet qu'elle contrôle, ce sommet est gagnant pour Alice

Si Bob est obligé de laisser Alice attendre la plage depuis un sommet qu'il contrôle, le sommet est gagnant pour Alice

Alice au pays des merveilles

Alice souhaite aller à la plage, pour cela elle doit traverser la ville, mais Bob souhaite l'en empêcher, et contrôle certaines intersections :



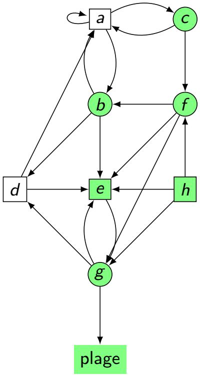
Alice contrôle les sommets ronds, et Bob les sommets carrés.

<u>Fact :</u> si Alice peut atteindre la plage depuis un sommet qu'elle contrôle, ce sommet est gagnant pour Alice

Si Bob est obligé de laisser Alice attendre la plage depuis un sommet qu'il contrôle, le sommet est gagnant pour Alice

Alice au pays des merveilles

Alice souhaite aller à la plage, pour cela elle doit traverser la ville, mais Bob souhaite l'en empêcher, et contrôle certaines intersections :



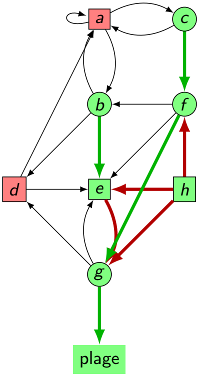
Alice contrôle les sommets ronds, et Bob les sommets carrés.

Fact : si Alice peut atteindre la plage depuis un sommet qu'elle contrôle, ce sommet est gagnant pour Alice

Si Bob est obligé de laisser Alice attendre la plage depuis un sommet qu'il contrôle, le sommet est gagnant pour Alice

Alice au pays des merveilles

Alice souhaite aller à la plage, pour cela elle doit traverser la ville, mais Bob souhaite l'en empêcher, et contrôle certaines intersections :



Alice contrôle les sommets ronds, et Bob les sommets carrés.

Fact : si Alice peut atteindre la plage depuis un sommet qu'elle contrôle, ce sommet est gagnant pour Alice

Si Bob est obligé de laisser Alice attendre la plage depuis un sommet qu'il contrôle, le sommet est gagnant pour Alice

Alice au pays de la formalisation

Soit $\mathcal{G} = (S, A)$ un graphe tel que les sommets soient partitionnés entre les sommets contrôlés par Alice et par Bob ; $S = S_A \sqcup S_B$. Alice souhaite aller dans les sommets $W \subset G$, depuis quels sommets a-t-elle une stratégie gagnante ?

Alice au pays de la formalisation

Soit $\mathcal{G} = (S, A)$ un graphe tel que les sommets soient partitionnés entre les sommets contrôlés par Alice et par Bob ; $S = S_A \sqcup S_B$. Alice souhaite aller dans les sommets $W \subset G$, depuis quels sommets a-t-elle une stratégie gagnante ?

$$\begin{aligned}
Attr_A^0 =& W \\
Attr_A^{n+1} =& Attr_0^n \cup \\
& \{v \in S_A | \exists w \in Attr_A^n, vAw\} \cup \\
& \{v \in S_B | \forall w \in S, vAw \Rightarrow w \in Attr_0^n\}
\end{aligned}$$

Alice au pays de la formalisation

Soit $\mathcal{G} = (S, A)$ un graphe tel que les sommets soient partitionnés entre les sommets contrôlés par Alice et par Bob ; $S = S_A \sqcup S_B$. Alice souhaite aller dans les sommets $W \subset G$, depuis quels sommets a-t-elle une stratégie gagnante ?

$$\begin{aligned} Attr_A^0 =& W \\ Attr_A^{n+1} =& Attr_0^n \cup \\ & \{v \in S_A | \exists w \in Attr_A^n, vAw\} \cup \\ & \{v \in S_B | \forall w \in S, vAw \Rightarrow w \in Attr_0^n\} \end{aligned}$$

$$Attr_A^0 \subseteq Attr_A^1 \subseteq Attr_A^2 \subseteq \cdots \subseteq Attr_A^{|S|}$$

Alice au pays de la formalisation

Soit $\mathcal{G} = (S, A)$ un graphe tel que les sommets soient partitionnés entre les sommets contrôlés par Alice et par Bob ; $S = S_A \sqcup S_B$. Alice souhaite aller dans les sommets $W \subset G$, depuis quels sommets a-t-elle une stratégie gagnante ?

$$\begin{aligned}
Attr_A^0 =& W \\
Attr_A^{n+1} =& Attr_0^n \cup \\
& \{v \in S_A | \exists w \in Attr_A^n, vAw\} \cup \\
& \{v \in S_B | \forall w \in S, vAw \Rightarrow w \in Attr_0^n\}
\end{aligned}$$

$Attr_A^0 \subseteq Attr_A^1 \subseteq Attr_A^2 \subseteq \cdots \subseteq Attr_A^{|S|}$

$Attr_A^i$ est l'ensemble des sommets tels que Alice peut gagner en moins de i mouvements.

Alice au pays de la formalisation

Soit $\mathcal{G} = (S, A)$ un graphe tel que les sommets soient partitionnés entre les sommets contrôlés par Alice et par Bob ; $S = S_A \sqcup S_B$. Alice souhaite aller dans les sommets $W \subset G$, depuis quels sommets a-t-elle une stratégie gagnante ?

$$\begin{aligned} Attr_A^0 &= W \\ Attr_A^{n+1} &= Attr_0^n \cup \\ &\quad \{v \in S_A | \exists w \in Attr_A^n, vAw\} \cup \\ &\quad \{v \in S_B | \forall w \in S, vAw \Rightarrow w \in Attr_0^n\} \end{aligned}$$

$Attr_A^0 \subseteq Attr_A^1 \subseteq Attr_A^2 \subseteq \cdots \subseteq Attr_A^{|S|}$

$Attr_A^i$ est l'ensemble des sommets tels que Alice peut gagner en moins de i mouvements.

$W_A = Attr_A^{|S|}$: la région gagnante pour Alice (plus petit point fixe).

Alice au pays de la formalisation

Soit $\mathcal{G} = (S, A)$ un graphe tel que les sommets soient partitionnés entre les sommets contrôlés par Alice et par Bob ; $S = S_A \sqcup S_B$. Alice souhaite aller dans les sommets $W \subset G$, depuis quels sommets a-t-elle une stratégie gagnante ?

$$\begin{aligned}
Attr_A^0 =& W \\
Attr_A^{n+1} =& Attr_0^n \cup \\
& \{v \in S_A | \exists w \in Attr_A^n, vAw\} \cup \\
& \{v \in S_B | \forall w \in S, vAw \Rightarrow w \in Attr_0^n\}
\end{aligned}$$

$Attr_A^0 \subseteq Attr_A^1 \subseteq Attr_A^2 \subseteq \cdots \subseteq Attr_A^{|S|}$

$Attr_A^i$ est l'ensemble des sommets tels que Alice peut gagner en moins de i mouvements.

$W_A = Attr_A^{|S|}$: la région gagnante pour Alice (plus petit point fixe).

$W_B = S \setminus Attr_A^{|S|}$: la région gagnante pour Bob.

Alice au pays de la formalisation

Soit $\mathcal{G} = (S, A)$ un graphe tel que les sommets soient partitionnés entre les sommets contrôlés par Alice et par Bob ; $S = S_A \sqcup S_B$. Alice souhaite aller dans les sommets $W \subset G$, depuis quels sommets a-t-elle une stratégie gagnante ?

$$Attr_A^0 = W$$

$$Attr_A^{n+1} = Attr_0^n \cup \{v \in S_A | \exists w \in Attr_A^n, vAw\} \cup \{v \in S_B | \forall w \in S, vAw \Rightarrow w \in Attr_0^n\}$$

$Attr_A^0 \subseteq Attr_A^1 \subseteq Attr_A^2 \subseteq \cdots \subseteq Attr_A^{|S|}$

$Attr_A^i$ est l'ensemble des sommets tels que Alice peut gagner en moins de i mouvements.

$W_A = Attr_A^{|S|}$: la région gagnante pour Alice (plus petit point fixe).

$W_B = S \setminus Attr_A^{|S|}$: la région gagnante pour Bob.

Algorithm 17: Reachability

Data: graphe G=(S,A), partition S_A, S_B, région gagnante W

Result: Région gagnante pour Alice W_A

```
Attr= W
V=S \ W
for i in 1,...,|S| do
    Temp=∅ ;
    for s in V do
        if s in S_A then
            if CanReach(G,Attr,s) then
                Temp=Temp∪{s}
                V=V\{s}
        else
            if MustReach(G,Attr,s)
             then
                Temp=Temp∪{s}
                V=V\{s}
    Attr=Attr∪ Temp
return Attr
```

Reachability - Safety

Il existe une notion duale de jeu : les jeux de sécurité $\rightsquigarrow$ Alice veut éviter la région L.

Reachability - Safety

Il existe une notion duale de jeu : les jeux de sécurité $\rightsquigarrow$ Alice veut éviter la région L.
Résolution : Les positions gagnantes pour Alice correspondent aux positions perdantes pour Bob s'il veut accéder à L, on résout donc ce jeu d'accessibilité pour Bob.

Reachability - Safety

Il existe une notion duale de jeu : les jeux de sécurité $\rightsquigarrow$ Alice veut éviter la région L.

Résolution : Les positions gagnantes pour Alice correspondent aux positions perdantes pour Bob s'il veut accéder à L, on résout donc ce jeu d'accessibilité pour Bob.

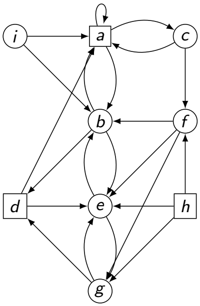
Alice souhaite éviter a, b

Reachability - Safety

Il existe une notion duale de jeu : les jeux de sécurité $\rightsquigarrow$ Alice veut éviter la région L.

Résolution : Les positions gagnantes pour Alice correspondent aux positions perdantes pour Bob s'il veut accéder à L, on résout donc ce jeu d'accessibilité pour Bob.

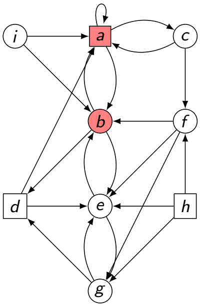
Alice souhaite éviter $a, b \rightsquigarrow$ positions perdantes pour Bob s'il veut accéder à $\{a, b\}$



Reachability - Safety

Il existe une notion duale de jeu : les jeux de sécurité $\rightsquigarrow$ Alice veut éviter la région L.
Résolution : Les positions gagnantes pour Alice correspondent aux positions perdantes pour Bob s'il veut accéder à L, on résout donc ce jeu d'accessibilité pour Bob.

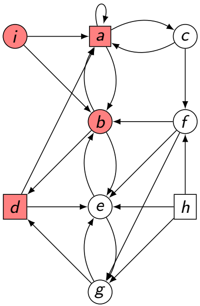
Alice souhaite éviter $a, b \rightsquigarrow$ positions perdantes pour Bob s'il veut accéder à $\{a, b\}$



Reachability - Safety

Il existe une notion duale de jeu : les jeux de sécurité $\rightsquigarrow$ Alice veut éviter la région L.
Résolution : Les positions gagnantes pour Alice correspondent aux positions perdantes pour Bob s'il veut accéder à L, on résout donc ce jeu d'accessibilité pour Bob.

Alice souhaite éviter $a, b \rightsquigarrow$ positions perdantes pour Bob s'il veut accéder à $\{a, b\}$

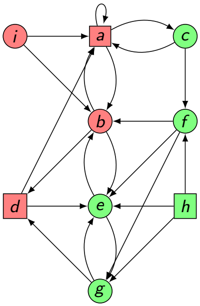


Reachability - Safety

Il existe une notion duale de jeu : les jeux de sécurité $\rightsquigarrow$ Alice veut éviter la région L.

Résolution : Les positions gagnantes pour Alice correspondent aux positions perdantes pour Bob s'il veut accéder à L, on résout donc ce jeu d'accessibilité pour Bob.

Alice souhaite éviter $a, b \rightsquigarrow$ positions perdantes pour Bob s'il veut accéder à $\{a, b\}$



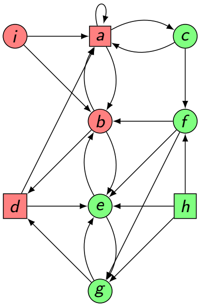
Reachability - Safety

Il existe une notion duale de jeu : les jeux de sécurité $\rightsquigarrow$ Alice veut éviter la région L.

Résolution : Les positions gagnantes pour Alice correspondent aux positions perdantes pour Bob s'il veut accéder à L, on résout donc ce jeu d'accessibilité pour Bob.

Alice souhaite éviter $a, b \rightsquigarrow$ positions perdantes pour Bob s'il veut accéder à $\{a, b\}$

$Attr_B(\{a, b\}) = \{a, b, d, i\}$

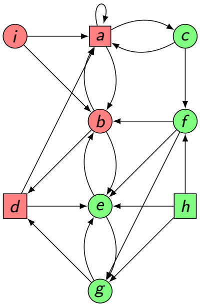


Reachability - Safety

Il existe une notion duale de jeu : les jeux de sécurité $\rightsquigarrow$ Alice veut éviter la région L.

Résolution : Les positions gagnantes pour Alice correspondent aux positions perdantes pour Bob s'il veut accéder à L, on résout donc ce jeu d'accessibilité pour Bob.

Alice souhaite éviter $a, b \rightsquigarrow$ positions perdantes pour Bob s'il veut accéder à $\{a, b\}$



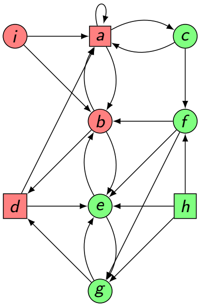
$Attr_B(\{a, b\}) = \{a, b, d, i\}$

Positions gagnantes pour Alice dans le jeu de sécurité : $\{c, e, f, g, h\}$

Reachability - Safety

Il existe une notion duale de jeu : les jeux de sécurité $\rightsquigarrow$ Alice veut éviter la région L.
Résolution : Les positions gagnantes pour Alice correspondent aux positions perdantes pour Bob s'il veut accéder à L, on résout donc ce jeu d'accessibilité pour Bob.

Alice souhaite éviter $a, b \rightsquigarrow$ positions perdantes pour Bob s'il veut accéder à $\{a, b\}$



$Attr_B(\{a, b\}) = \{a, b, d, i\}$
Positions gagnantes pour Alice dans le jeu de sécurité : $\{c, e, f, g, h\}$

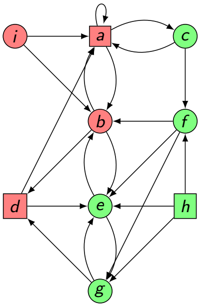
Applications au model-cheking (vérification formelle)

Reachability - Safety

Il existe une notion duale de jeu : les jeux de sécurité $\rightsquigarrow$ Alice veut éviter la région L.

Résolution : Les positions gagnantes pour Alice correspondent aux positions perdantes pour Bob s'il veut accéder à L, on résout donc ce jeu d'accessibilité pour Bob.

Alice souhaite éviter $a, b \rightsquigarrow$ positions perdantes pour Bob s'il veut accéder à $\{a, b\}$



$Attr_B(\{a, b\}) = \{a, b, d, i\}$

Positions gagnantes pour Alice dans le jeu de sécurité : $\{c, e, f, g, h\}$

Applications au model-cheking (vérification formelle)

e.g. G graphes des valeurs possible d'un paramètre dans un programme ; Alice = algorithme ; Bob = utilisateurs ; valeurs qui plantent le programme.