Calculabilité

Hubert Comon (ENS Paris-Saclay)

May 30, 2017

Un professeur paresseux

- Les élèves doivent écrire un programme qui calcule le pgcd de deux entiers strictement positifs.
- Pour faciliter la correction, le professeur envisage d'écrire un programme qui vérifie automatiquement les solutions proposées par les élèves.

Un professeur paresseux

- Les élèves doivent écrire un programme qui calcule le pgcd de deux entiers strictement positifs.
- Pour faciliter la correction, le professeur envisage d'écrire un programme qui vérifie automatiquement les solutions proposées par les élèves.

Un tel programme n'existe pas !

Calculabilité

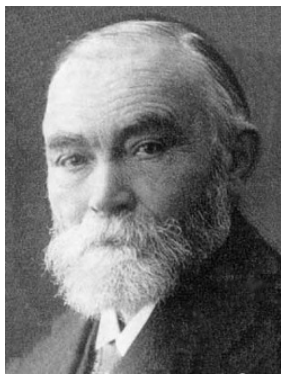
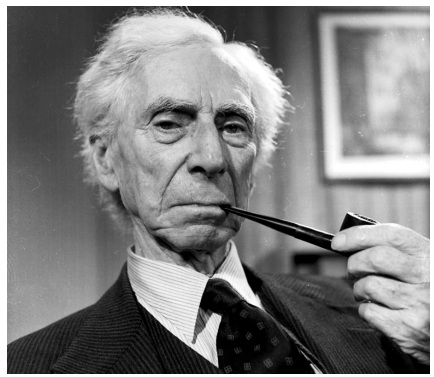
Que peut on espérer (ou qu'il est inutile d'espérer) faire avec des ordinateurs ?

Calculabilité

Que peut on espérer (ou qu'il est inutile d'espérer) faire avec des ordinateurs ?

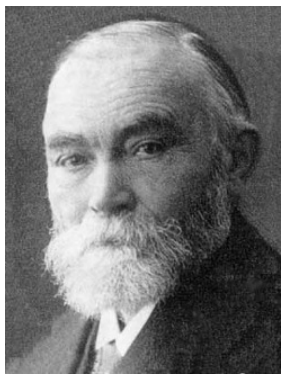
Il existe des problèmes (certains très simples) qu'on ne pourra jamais résoudre automatiquement, quels que soient la vitesse de calcul et la mémoire disponibles.

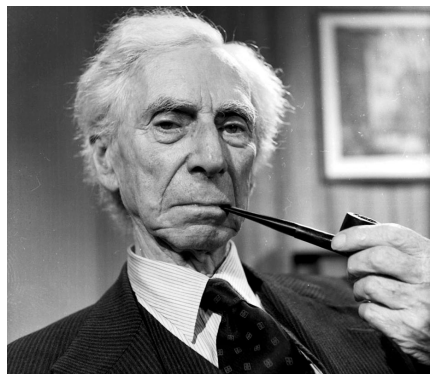
Les fondements des mathématiques


Les fondements des mathématiques

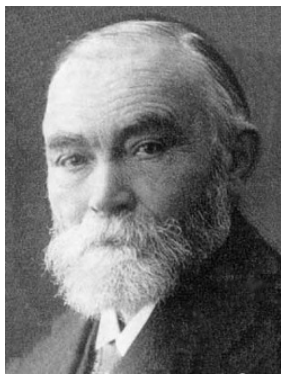
David Hilbert



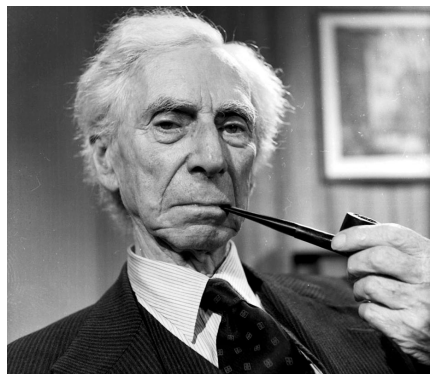


Les fondements des mathématiques

David Hilbert

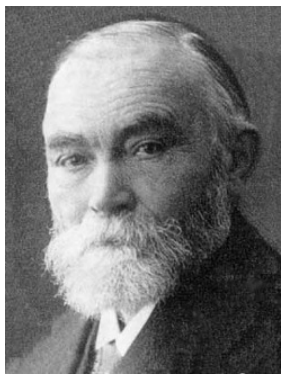


Gottlob Frege

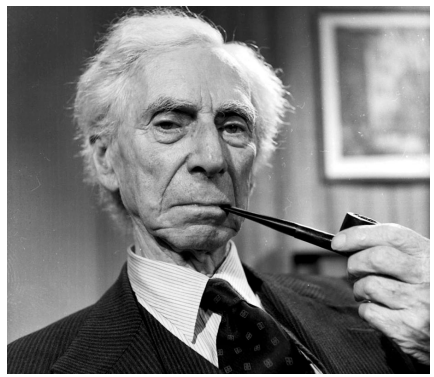


Les fondements des mathématiques

David Hilbert



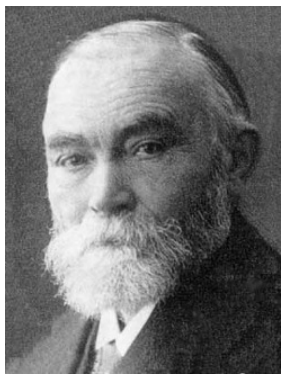
Gottlob Frege



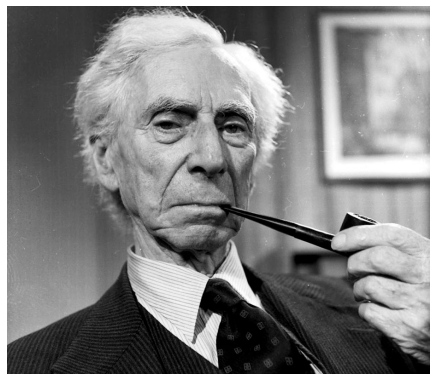
Bertrand Russell

Les fondements des mathématiques

David Hilbert



Gottlob Frege



Bertrand Russell

Kurt Gödel

Le paradoxe de Russell

Le paradoxe de Russell

Version grand public

Le barbier rase tous ceux qui ne rasent pas eux mêmes.

Le paradoxe de Russell

Version grand public

Le barbier rase tous ceux qui ne rasent pas eux mêmes.

Version Russell/Frege

$y = \{x \mid x \notin x\}$

Le paradoxe de Russell: Version informatique

$$\text{Eval}(''P'', D) = P(D)$$

Le paradoxe de Russell: Version informatique

$$\text{Eval}(''P'', D) = P(D)$$

$$\text{Diag}(''P'') = \begin{cases} 1 & \text{Si Eval}(''P'', ''P'') = 0 \\ 0 & \text{Sinon} \end{cases}$$

Le paradoxe de Russell: Version informatique

$$\text{Eval}(\text{"}P\text{"}, D) = P(D)$$

$$\text{Diag}(\text{"}P\text{"}) = \begin{cases} 1 & \text{Si Eval}(\text{"}P\text{"}, \text{"}P\text{"}) = 0 \\ 0 & \text{Sinon} \end{cases}$$

Diag(“Diag”) ?

Le paradoxe de Russell: Version informatique

$$\text{Eval}(\text{"}P\text{"}, D) = P(D)$$

$$\text{Diag}(\text{"}P\text{"}) = \begin{cases} 1 & \text{Si Eval}(\text{"}P\text{"}, \text{"}P\text{"}) = 0 \\ 0 & \text{Sinon} \end{cases}$$

Diag(“Diag”) ?

Il n’y a pas de programme Diag.

Le problème de l'arrêt

Il n'y a aucun programme A qui s'arrête sur toute donnée tel que:

Sur les données d'un programme P à un paramètre et une donnée D

$A(P, D) = 1$ ssi P s'arrête sur D

Le problème de l'arrêt

Il n'y a aucun programme A qui s'arrête sur toute donnée tel que:

Sur les données d'un programme P à un paramètre et une donnée D

$A(P, D) = 1$ ssi P s'arrête sur D

Preuve

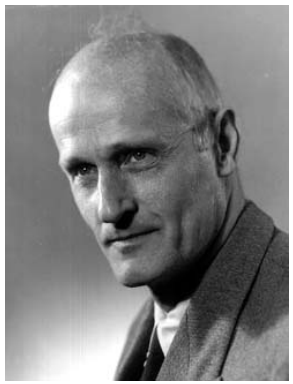
Par l'absurde: si A existe, on construit Q:

$$Q(P) \stackrel{\text{def}}{=} \text{if } A(P, P) = 1 \text{ then } \textbf{boucle} \text{ else } 1$$

Q s'arrête sur Q
ssi $A(Q, Q) = 1$ (par définition de A)
ssi Q ne s'arrête pas sur Q (par définition de Q)
Absurde.

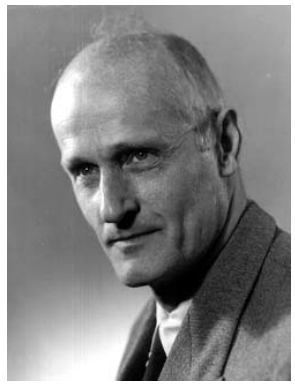
Qu'est ce qu'un programme ?





Qu'est ce qu'un programme ?

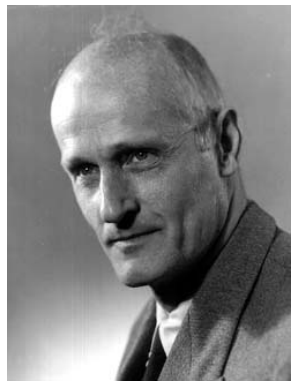
Alan Turing



Qu'est ce qu'un programme ?

Alan Turing

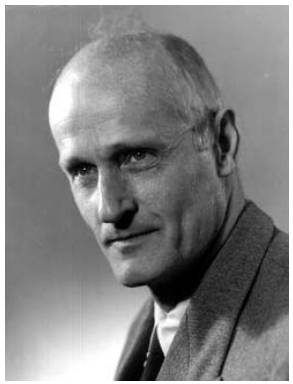
Alonzo Church



Qu'est ce qu'un programme ?

Alan Turing

Alonzo Church

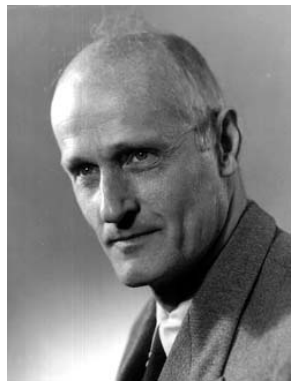


Stephen Kleene

Qu'est ce qu'un programme ?

Alan Turing

Alonzo Church

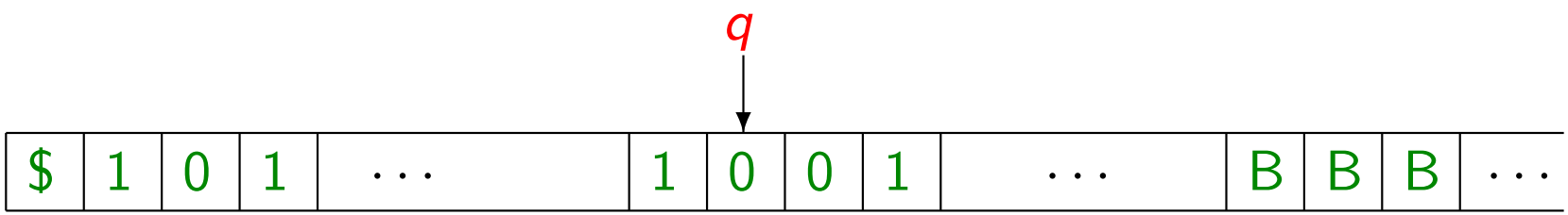


Stephen Kleene

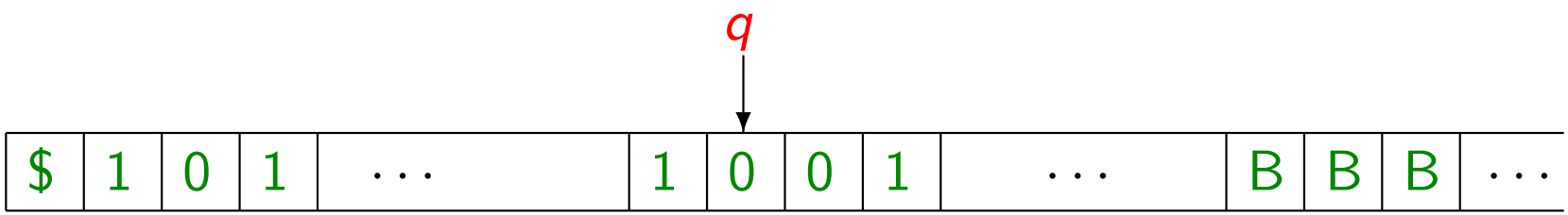
Thèse de Church-Turing: tous les modèles de calculs sont équivalents.

calculable = récursive totale = décidable (pour les prédicats).

Machines de Turing

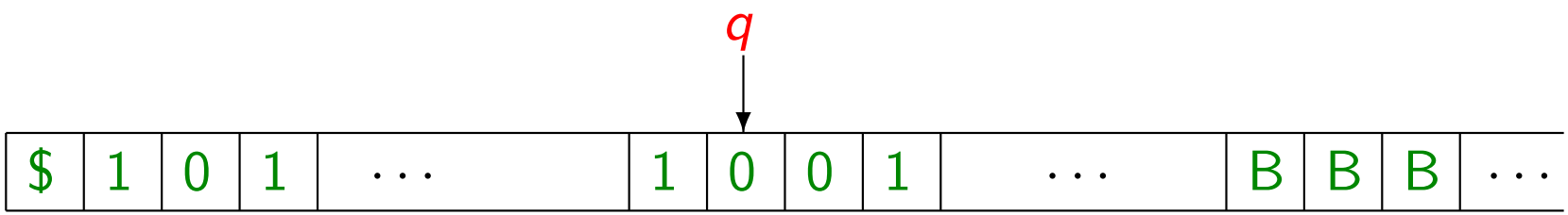


Machines de Turing

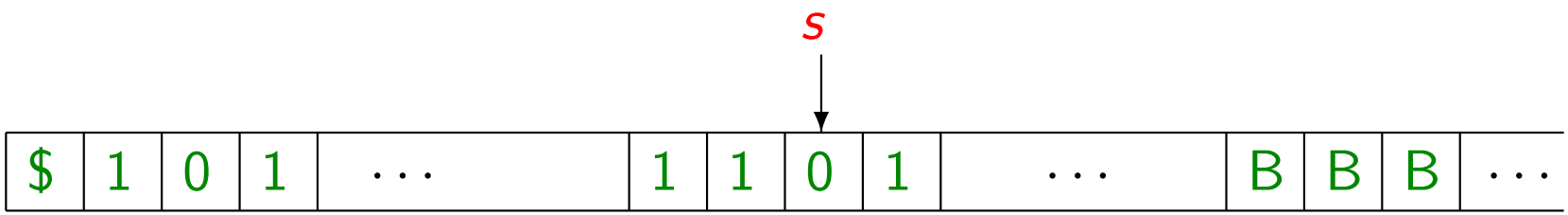


$\delta(q, 0) = (s, 1, \rightarrow)$

Machines de Turing



$\delta(q, 0) = (s, 1, \rightarrow)$



Calculabilité, récursivité, indécidablité

Indécidabilité

Un problème: $\begin{cases} \text{Donnée:} & D \\ \text{Question:} & Q \end{cases}$ où $D \in S$ et $Q \subseteq S$

est indécidable s'il n'existe aucune fonction calculable f de S dans $\{0, 1\}$ telle que $f(D) = 1$ ssi $D \in Q$.

Calculabilité, récursivité, indécidablité

Indécidabilité

Un problème: $\begin{cases} \text{Donnée:} & D \\ \text{Question:} & Q \end{cases}$ où $D \in S$ et $Q \subseteq S$

est indécidable s'il n'existe aucune fonction calculable f de S dans $\{0, 1\}$ telle que $f(D) = 1$ ssi $D \in Q$.

Exemple

Donnée: une machine M et une donnée d

Question: La machine M s'arrête sur la donnée d ?

est indécidable

Quelques problèmes indécidables (1)

Quelques problèmes indécidables (1)

Entscheidungsproblem

Donnée: une formule ϕ de la logique du premier ordre

Question: ϕ est valide ?

est indécidable

Quelques problèmes indécidables (1)

Entscheidungsproblem

Donnée: une formule ϕ de la logique du premier ordre

Question: ϕ est valide ?

est indécidable

Arithmétique

Donnée: une formule ϕ de l'arithmétique (avec $\times, +, 0, 1, =$)

Question: ϕ est valide (dans l'arithmétique de Peano) ?

est indécidable

Quelques problèmes indécidables (2)

Equations Diophantiennes (Davis, Matjasevic)

Donnée: une équation polynomiale E à coefficients entiers

Question: E admet elle au moins une solution entière ?

est indécidable

Quelques problèmes indécidables (2)

Equations Diophantiennes (Davis, Matjasevic)

Donnée: une équation polynomiale E à coefficients entiers

Question: E admet elle au moins une solution entière ?

est indécidable

Matrice nulle

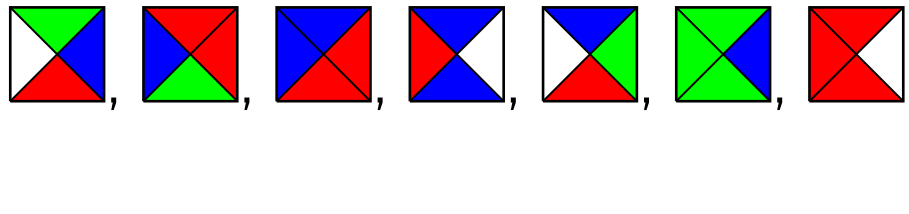
Donnée: un ensemble fini $\mathcal{M}$ de matrices 3X3 à coefficients dans $\mathbb{Z}$

Question: Peut on obtenir la matrice nulle comme produit de matrices de $\mathcal{M}$?

est indécidable

Indécidabilité du pavage

Donnée:

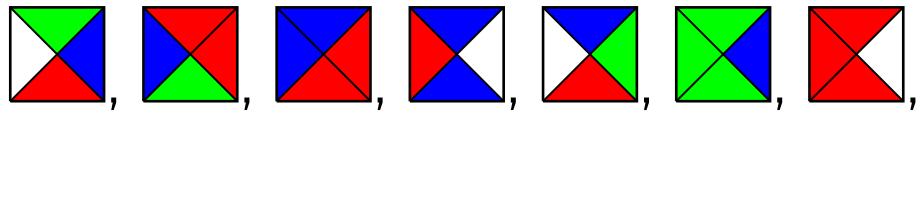
Un ensemble fini de tuiles:  , ...

Question:

Existe-t-il un pavage du quart de plan, à première tuile fixée et couleur-compatible ?

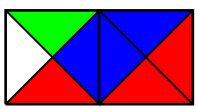
Indécidabilité du pavage

Donnée:

Un ensemble fini de tuiles:  , ...

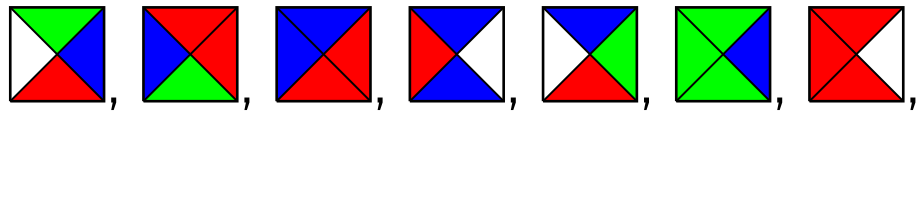
Question:

Existe-t-il un pavage du quart de plan, à première tuile fixée et couleur-compatible ?



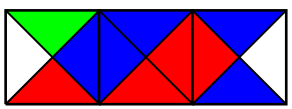
Indécidabilité du pavage

Donnée:

Un ensemble fini de tuiles:  , ...

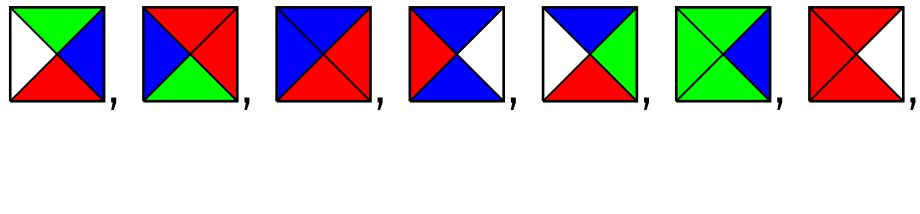
Question:

Existe-t-il un pavage du quart de plan, à première tuile fixée et couleur-compatible ?



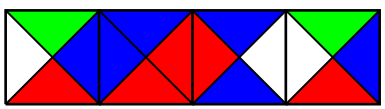
Indécidabilité du pavage

Donnée:

Un ensemble fini de tuiles:  , ...

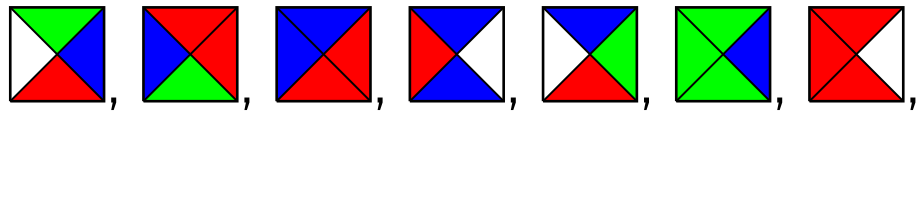
Question:

Existe-t-il un pavage du quart de plan, à première tuile fixée et couleur-compatible ?



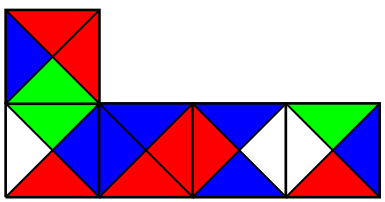
Indécidabilité du pavage

Donnée:

Un ensemble fini de tuiles:  , ...

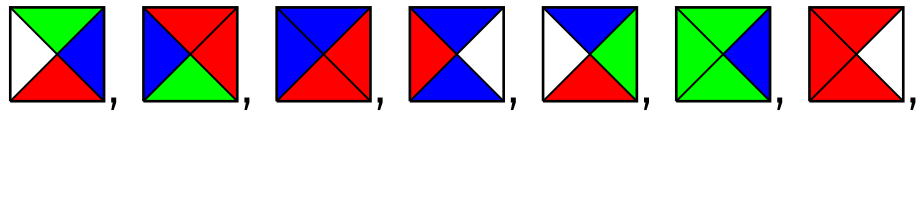
Question:

Existe-t-il un pavage du quart de plan, à première tuile fixée et couleur-compatible ?



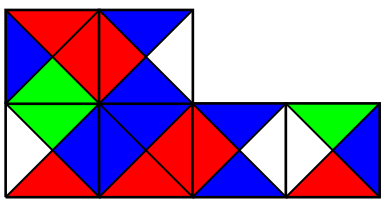
Indécidabilité du pavage

Donnée:

Un ensemble fini de tuiles:  , ...

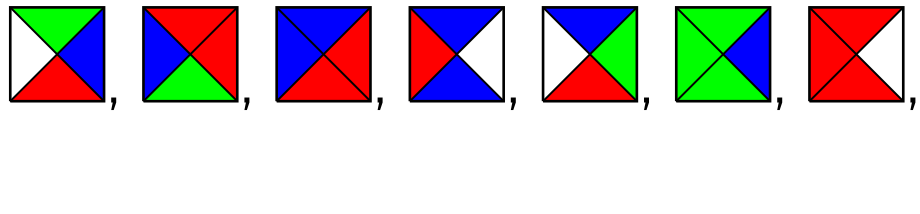
Question:

Existe-t-il un pavage du quart de plan, à première tuile fixée et couleur-compatible ?



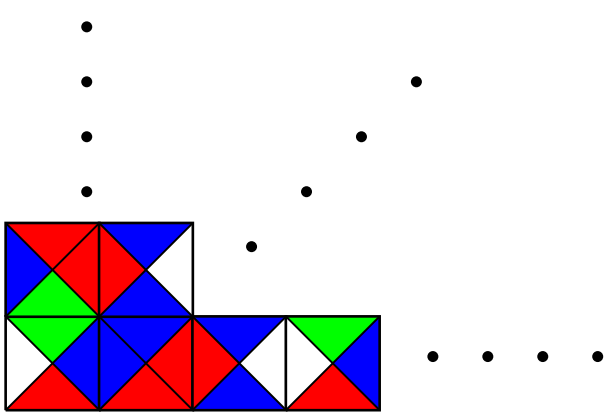
Indécidabilité du pavage

Donnée:

Un ensemble fini de tuiles:  , ...

Question:

Existe-t-il un pavage du quart de plan, à première tuile fixée et couleur-compatible ?

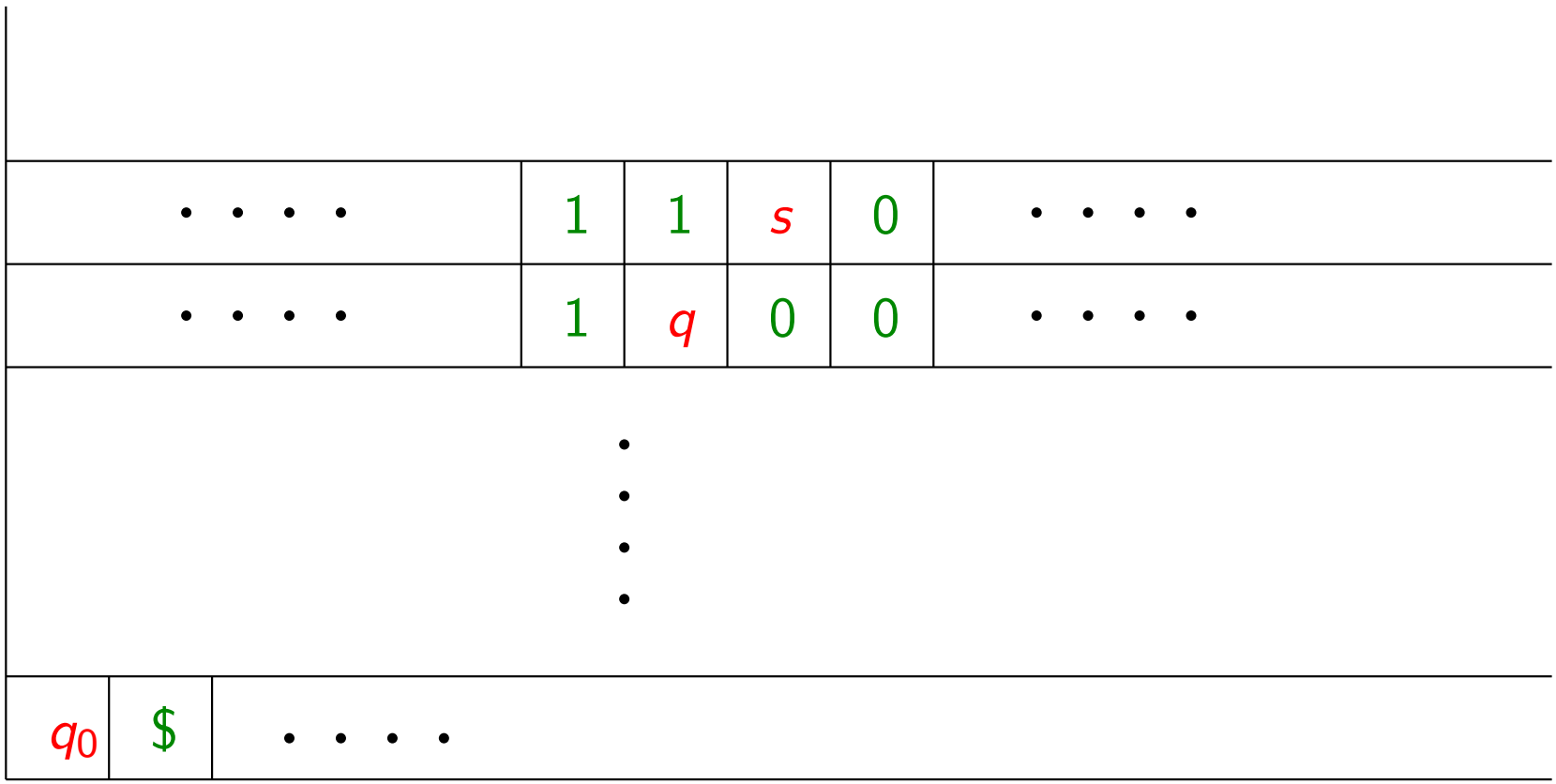


Ébauche de preuve

Réduction du problème du non-arrêt au problème de pavage.

Ébauche de preuve

Réduction du problème du non-arrêt au problème de pavage.



Autres exemples

- Robotique
- Virus
- Jeux

Théorie de la complexité

Difficulté intrinsèque des problèmes (indépendemment des algorithmes)

Théorie de la complexité

Difficulté intrinsèque des problèmes (indépendemment des algorithmes)

La classe **NP**

- Problèmes que l'on peut résoudre en temps polynômial avec une machine de Turing *non-déterministe*.
- Problèmes **NP**-difficiles = au moins aussi difficiles que tous les probèmes de la classe **NP**