

Théorie de la complexité

Partie Calculabilité : Introduction

Thomas Place et Marc Zeitoun

Université de Bordeaux

Septembre 2020

Objectifs de cette partie

1. Présenter quelques limites du calcul automatique.
2. Donner un aperçu de problèmes informatiques difficiles :
 - qu'on **ne peut pas** résoudre par algorithme.

Qu'est-ce qu'un problème en informatique ?

Ceci **n'est pas** un problème en informatique :

Résolution d'une équation du second degré

L'équation $42x^2 + 7x + 8 = 0$ a-t-elle au moins une solution dans \mathbb{R} ?

Qu'est-ce qu'un problème en informatique ?

Ceci **n'est pas** un problème en informatique :

Résolution d'une équation du second degré

L'équation $42x^2 + 7x + 8 = 0$ a-t-elle au moins une solution dans \mathbb{R} ?

Ceci **est** un problème en informatique :

« Résolution » de toutes les équations du second degré

- **Entrée** : étant **donnés** des entiers a, b, c (avec $a \neq 0$),
- **Sortie** : est-ce que $ax^2 + bx + c = 0$ a au moins une solution dans \mathbb{R} ?

Qu'est-ce qu'un problème en informatique ?

Ceci **n'est pas** un problème en informatique :

Résolution d'une équation du second degré

L'équation $42x^2 + 7x + 8 = 0$ a-t-elle au moins une solution dans \mathbb{R} ?

Ceci **est** un problème en informatique :

« Résolution » de toutes les équations du second degré

- **Entrée** : étant **donnés** des entiers a, b, c (avec $a \neq 0$),
- **Sortie** : est-ce que $ax^2 + bx + c = 0$ a au moins une solution dans \mathbb{R} ?

Remarque

Une méthode automatique de résolution de toutes les équations du second degré permet en particulier de résoudre le premier exercice.

Qu'est-ce qu'un problème en informatique?

Qu'est-ce qu'un problème informatique?

Problème : porte sur des **instances** (ou **entrées**).

Pour les problèmes intéressants, l'instance provient d'un **ensemble infini**.

Qu'est-ce qu'un problème en informatique?

Qu'est-ce qu'un problème informatique ?

Problème : porte sur des **instances** (ou **entrées**).

Pour les problèmes intéressants, l'instance provient d'un **ensemble infini**.

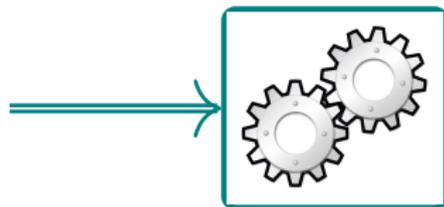
Si la réponse attendue est **oui** ou **non**, on parle de **problème de décision**.

La question que l'on se pose en calculabilité

Est-ce que le problème peut-être **résolu** par un algorithme ?

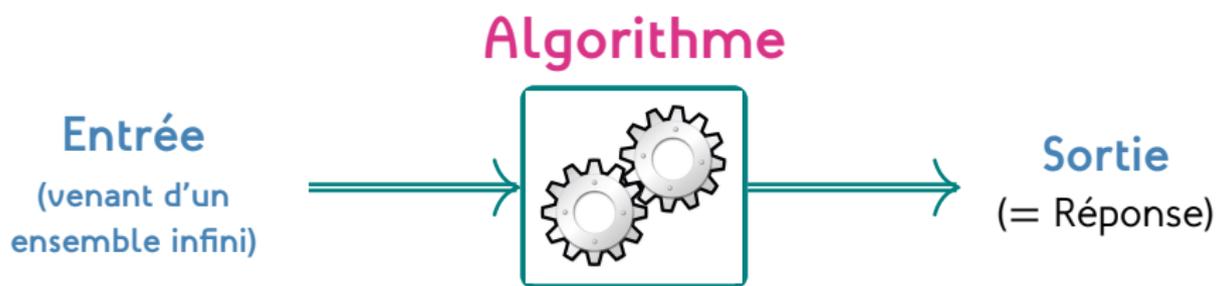
Un algorithme « résout » un problème

Entrée
(venant d'un
ensemble infini)

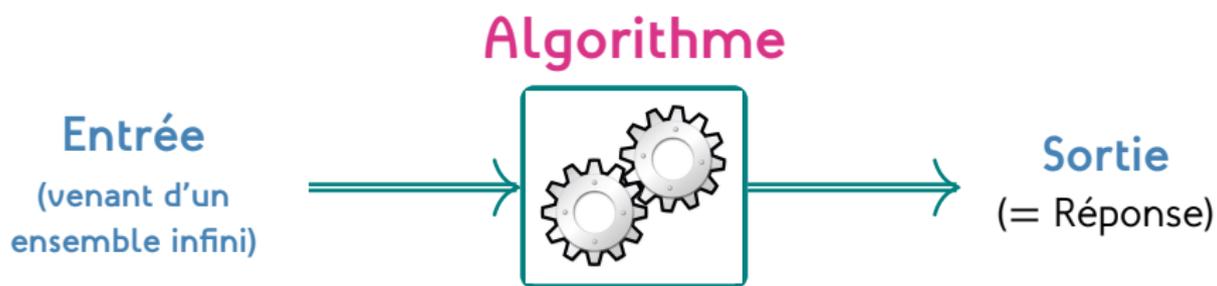


Algorithme

Un algorithme « résout » un problème

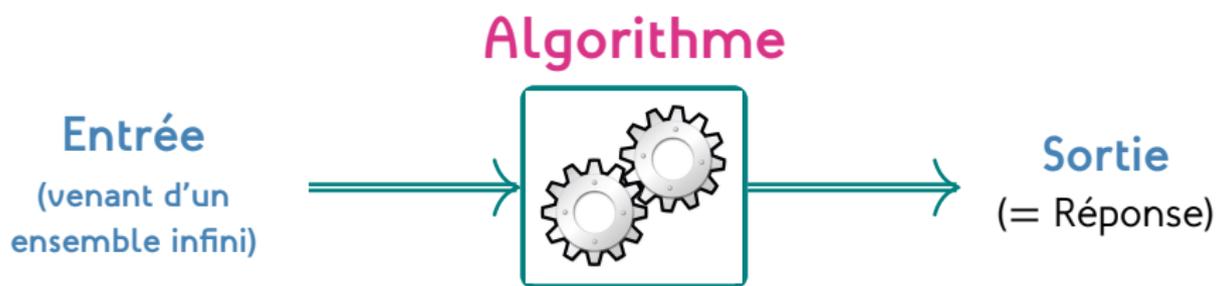


Un algorithme « résout » un problème



En particulier, un algorithme **s'arrête sur chaque entrée.**

Un algorithme « résout » un problème



En particulier, un algorithme **s'arrête sur chaque entrée**.

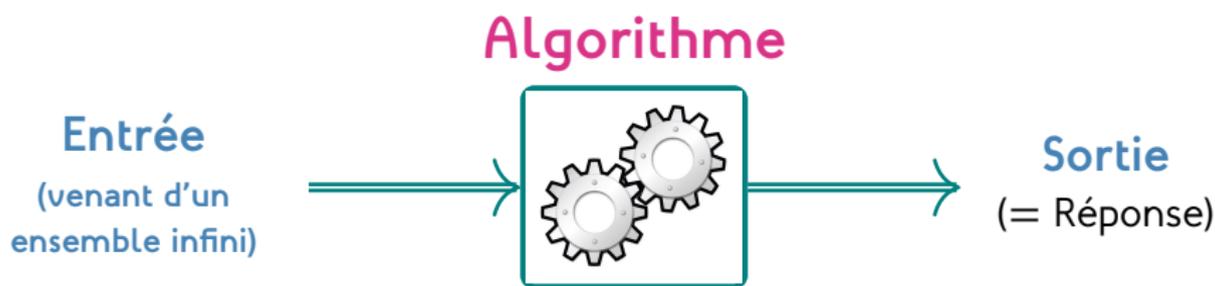
Exemples

- Algorithme de résolution d'équations du second degré.

Entrée attendue : 3 entiers (a, b, c) .

Sortie attendue : Solutions réelles de $ax^2 + bx + c = 0$.

Un algorithme « résout » un problème



En particulier, un algorithme **s'arrête sur chaque entrée**.

Exemples

- Algorithme de résolution d'équations du second degré.
Entrée attendue : 3 entiers (a, b, c) .
Sortie attendue : Solutions réelles de $ax^2 + bx + c = 0$.
- Algorithme de résolution d'équations **Diophantiennes** à **une** variable.
Entrée attendue : Une liste d'entiers a_0, \dots, a_n (où $a_n \neq 0$).
Sortie attendue : Solutions **entières** de $\sum_{i=0}^n a_i x^i = 0$.

Équations Diophantiennes à une variable

Problème : résolution d'équations Diophantiennes à une variable

Instance : Une liste d'entiers a_0, \dots, a_n (avec $a_n \neq 0$).

Sortie : Solutions **entières** de $\sum_{i=0}^n a_i x^i = 0$.

Équations Diophantiennes à une variable

Problème : résolution d'équations Diophantiennes à une variable

Instance : Une liste d'entiers a_0, \dots, a_n (avec $a_n \neq 0$).

Sortie : Solutions **entières** de $\sum_{i=0}^n a_i x^i = 0$.

Une propriété simple

Les éventuelles solutions sont dans $[-M; M]$, avec $M = n \times \max\{|a_i|\}$.

Équations Diophantiennes à une variable

Problème : résolution d'équations Diophantiennes à une variable

Instance : Une liste d'entiers a_0, \dots, a_n (avec $a_n \neq 0$).

Sortie : Solutions **entières** de $\sum_{i=0}^n a_i x^i = 0$.

Une propriété simple

Les éventuelles solutions sont dans $[-M; M]$, avec $M = n \times \max\{|a_i|\}$.

Car si $|x| > M$, alors pour tout k entre 0 et $n - 1$:

$$|x^{n-k}| \geq |x| > n|a_k|$$

par définition de M .

$$|x^n|/n > |a_k x^k|$$

en multipliant par $|x^k|/n$.

Équations Diophantiennes à une variable

Problème : résolution d'équations Diophantiennes à une variable

Instance : Une liste d'entiers a_0, \dots, a_n (avec $a_n \neq 0$).

Sortie : Solutions **entières** de $\sum_{i=0}^n a_i x^i = 0$.

Une propriété simple

Les éventuelles solutions sont dans $[-M; M]$, avec $M = n \times \max\{|a_i|\}$.

Car si $|x| > M$, alors pour tout k entre 0 et $n - 1$:

$$|x^{n-k}| \geq |x| > n|a_k| \quad \text{par définition de } M.$$

$$|x^n|/n > |a_k x^k| \quad \text{en multipliant par } |x^k|/n.$$

$$\text{Donc } |a_n x^n| \geq |x^n| = \sum_{k=0}^{n-1} |x^n|/n > \sum_{k=0}^{n-1} |a_k x^k| \geq \left| \sum_{k=0}^{n-1} a_k x^k \right|.$$

Équations Diophantiennes à une variable

Problème : résolution d'équations Diophantiennes à une variable

Instance : Une liste d'entiers a_0, \dots, a_n (avec $a_n \neq 0$).

Sortie : Solutions **entières** de $\sum_{i=0}^n a_i x^i = 0$.

Une propriété simple

Les éventuelles solutions sont dans $[-M; M]$, avec $M = n \times \max\{|a_i|\}$.

Car si $|x| > M$, alors pour tout k entre 0 et $n - 1$:

$$|x^{n-k}| \geq |x| > n|a_k| \quad \text{par définition de } M.$$

$$|x^n|/n > |a_k x^k| \quad \text{en multipliant par } |x^k|/n.$$

$$\text{Donc } |a_n x^n| \geq |x^n| = \sum_{k=0}^{n-1} |x^n|/n > \sum_{k=0}^{n-1} |a_k x^k| \geq \left| \sum_{k=0}^{n-1} a_k x^k \right|.$$

$$\text{Enfin, } \left| a_n x^n + \sum_{k=0}^{n-1} a_k x^k \right| \geq |a_n x^n| - \left| \sum_{k=0}^{n-1} a_k x^k \right| > 0.$$

Équations Diophantiennes à une variable

Problème : résolution d'équations Diophantiennes à une variable

Instance : Une liste d'entiers a_0, \dots, a_n (avec $a_n \neq 0$).

Sortie : Solutions **entières** de $\sum_{i=0}^n a_i x^i = 0$.

Une propriété simple

Les éventuelles solutions sont dans $[-M; M]$, avec $M = n \times \max\{|a_i|\}$.

Car si $|x| > M$, alors pour tout k entre 0 et $n - 1$:

$$|x^{n-k}| \geq |x| > n|a_k| \quad \text{par définition de } M.$$

$$|x^n|/n > |a_k x^k| \quad \text{en multipliant par } |x^k|/n.$$

$$\text{Donc } |a_n x^n| \geq |x^n| = \sum_{k=0}^{n-1} |x^n|/n > \sum_{k=0}^{n-1} |a_k x^k| \geq \left| \sum_{k=0}^{n-1} a_k x^k \right|.$$

$$\text{Enfin, } \left| a_n x^n + \sum_{k=0}^{n-1} a_k x^k \right| \geq |a_n x^n| - \left| \sum_{k=0}^{n-1} a_k x^k \right| > 0.$$

$$\text{En particulier, on ne pas avoir } a_n x^n + \sum_{k=0}^{n-1} a_k x^k = 0. \quad \square$$

Équations Diophantiennes à une variable

Problème : résolution d'équations Diophantiennes à une variable

Instance : Une liste d'entiers a_0, \dots, a_n (avec $a_n \neq 0$).

Sortie : Solutions **entières** de $\sum_{i=0}^n a_i x^i = 0$.

Une propriété simple

Les éventuelles solutions sont dans $[-M; M]$, avec $M = n \times \max\{|a_i|\}$.

Algorithme de résolution basé sur cette propriété

1. Calculer M .
2. Initialiser une liste vide pour mémoriser les solutions
3. Pour chaque entier z entre $-M$ et M ,
 - 3.1 Tester si $\sum_{i=0}^n a_i z^i = 0$.
 - 3.2 Si oui, ajouter z à la liste des solutions.
4. Retourner la liste des solutions.

Équations Diophantiennes à une variable

Problème : résolution d'équations Diophantiennes à une variable

Instance : Une liste d'entiers a_0, \dots, a_n (avec $a_n \neq 0$).

Sortie : Solutions **entières** de $\sum_{i=0}^n a_i x^i = 0$.

Une propriété simple

Les éventuelles solutions sont dans $[-M; M]$, avec $M = n \times \max\{|a_i|\}$.

Algorithme de résolution basé sur cette propriété

1. Calculer M .
2. Initialiser une liste vide pour mémoriser les solutions
3. Pour chaque entier z entre $-M$ et M ,
 - 3.1 Tester si $\sum_{i=0}^n a_i z^i = 0$.
 - 3.2 Si oui, ajouter z à la liste des solutions.
4. Retourner la liste des solutions.

Algo « force brute »
(pas besoin de comprendre l'équation)

Récapitulatif

Vocabulaire important

- Problème (en informatique).
- Algorithme.
- Instance (ou entrée) d'un problème (ou d'un algorithme).
- Sortie.

Récapitulatif

Vocabulaire important

- Problème (en informatique).
- Algorithme.
- Instance (ou entrée) d'un problème (ou d'un algorithme).
- Sortie.

Problème « intéressant en info » \Rightarrow infinité d'instances

Les problèmes suivants n'entrent **pas** dans notre cadre.

- Conjecture de Syracuse « $3x + 1$ »,
- Conjecture de Goldbach,
- Conjecture des nombres premiers jumeaux.

Exemples de problèmes de décision

Problème 0 Instance Un nombre entier positif n .
Question n est-il pair ?

Exemples de problèmes de décision

Problème 0 Instance Un nombre entier positif n .
Question n est-il pair ?



```
def est_pair(n):  
    return n % 2 == 0
```

Exemples de problèmes de décision

Problème 0

Instance Un nombre entier positif n .
Question n est-il pair ?



Problème 1

I. Un nombre entier positif n .
Q. n est-il premier ?

Exemples de problèmes de décision

Problème 0

Instance Un nombre entier positif n .
Question n est-il pair?



Problème 1

I. Un nombre entier positif n .
Q. n est-il premier?



```
def est_premier(n):  
    return (n == 2 or  
            # cas impair  
            n > 2 and n % 2 != 0 and  
            all(n % d != 0 for d in range(3, int(n**0.5)+1, 2)))
```

Exemples de problèmes de décision

Problème 0

Instance Un nombre entier positif n .
Question n est-il pair?



Problème 1

I. Un nombre entier positif n .
Q. n est-il premier?



```
def est_premier(n):  
    return (n == 2 or  
            # cas impair  
            n > 2 and n % 2 != 0 and  
            all(n % d != 0 for d in range(3, int(n**0.5)+1, 2)))
```

Exemples de problèmes de décision

Problème 0

Instance Un nombre entier positif n .
Question n est-il pair ?



Problème 1

I. Un nombre entier positif n .
Q. n est-il premier ?



Problème 2

I. Un programme en Python.
Q. Le programme est-il syntaxiquement correct ?

Exemples de problèmes de décision

Problème 0

Instance Un nombre entier positif n .
Question n est-il pair ?



Problème 1

I. Un nombre entier positif n .
Q. n est-il premier ?



Problème 2

I. Un programme en Python.
Q. Le programme est-il syntaxiquement correct ?



Exemples de problèmes de décision (2)

Problème 3 (Tschisla) Instance Des entiers $n, k > 0$ et un chiffre $c \neq 0$.

Question Existe-t-il une expression,

- sur les opérateurs $+, -, \times, /, \sqrt{}, !$,
- utilisant au plus k fois le chiffre c ,
- et dont la valeur est n ?

Exemples de problèmes de décision (2)

Problème 3 (Tschisla) Instance Des entiers $n, k > 0$ et un chiffre $c \neq 0$.

Question Existe-t-il une expression,

- sur les opérateurs $+, -, \times, /, \sqrt{}, !$,
- utilisant au plus k fois le chiffre c ,
- et dont la valeur est n ?

Problème 4 (Skolem)

- i. Une suite récurrente à coefficients entiers, donnée par des entiers a_i, b_i ($0 \leq i < k$) et

$$\begin{cases} u_0 = a_0, \dots, u_{k-1} = a_{k-1} \\ u_{n+k} = b_{k-1}u_{n+k-1} + \dots + b_0u_n \end{cases}$$

Q. Existe-t-il n tel que $u_n = 0$?

Exemples de problèmes de décision (2)

Problème 3 (Tschisla) Instance Des entiers $n, k > 0$ et un chiffre $c \neq 0$.

Question Existe-t-il une expression,

- sur les opérateurs $+, -, \times, /, \sqrt{}, !$,
- utilisant au plus k fois le chiffre c ,
- et dont la valeur est n ?

Problème 4 (Skolem)

- I. Une suite récurrente à coefficients entiers, donnée par des entiers a_i, b_i ($0 \leq i < k$) et

$$\begin{cases} u_0 = a_0, \dots, u_{k-1} = a_{k-1} \\ u_{n+k} = b_{k-1}u_{n+k-1} + \dots + b_0u_n \end{cases}$$

Q. Existe-t-il n tel que $u_n = 0$?

Pour ces deux problèmes, **on ne connaît pas d'algorithme.**

Exemples de problèmes de décision (3)

- Problème 5** Instance Une équation Diophantienne à **11 variables**.
Question L'équation a-t-elle une solution dans \mathbb{Z} ?

Exemples de problèmes de décision (3)

Problème 5 Instance Une équation Diophantienne à **11 variables**.

Question L'équation a-t-elle une solution dans \mathbb{Z} ?

Problème 6 I. Un nombre fini de **types** de pièces de Tetris.

Q. Peut-on paver le plan infini avec ces types de pièces?

Exemples de problèmes de décision (3)

Problème 5 Instance Une équation Diophantienne à **11 variables**.

Question L'équation a-t-elle une solution dans \mathbb{Z} ?

Problème 6 I. Un nombre fini de **types** de pièces de Tetris.

Q. Peut-on paver le plan infini avec ces types de pièces?

Problème 7 I. Le code source d'un **programme P** et une entrée x .

Q. Le programme **P** s'arrête-t-il sur l'entrée x ?

Exemples de problèmes de décision (3)

Problème 5 Instance Une équation Diophantienne à **11 variables**.

Question L'équation a-t-elle une solution dans \mathbb{Z} ?

Problème 6 I. Un nombre fini de **types** de pièces de Tetris.

Q. Peut-on paver le plan infini avec ces types de pièces?

Problème 7 I. Le code source d'un **programme P** et une entrée x .

Q. Le programme **P** s'arrête-t-il sur l'entrée x ?

Exemples de problèmes de décision (3)

Problème 5 Instance Une équation Diophantienne à **11 variables**.
Question L'équation a-t-elle une solution dans \mathbb{Z} ?

Problème 6 I. Un nombre fini de **types** de pièces de Tetris.
Q. Peut-on paver le plan infini avec ces types de pièces?

Problème 7 I. Le code source d'un **programme P** et une entrée x .
Q. Le programme **P** s'arrête-t-il sur l'entrée x ?

Bad news...

Pour ces problèmes, on ne connaît pas d'algorithme ...

Exemples de problèmes de décision (3)

Problème 5 Instance Une équation Diophantienne à **11 variables**.
Question L'équation a-t-elle une solution dans \mathbb{Z} ?

Problème 6 I. Un nombre fini de **types** de pièces de Tetris.
Q. Peut-on paver le plan infini avec ces types de pièces?

Problème 7 I. Le code source d'un **programme P** et une entrée x .
Q. Le programme **P** s'arrête-t-il sur l'entrée x ?

Bad news...

Pour ces problèmes, on ne connaît pas d'algorithme ...

Pire : on sait prouver qu'**il n'en existe pas !**

Statut des problèmes en informatique

😊 On connaît un algorithme qui résout le problème.

Statut des problèmes en informatique

😊 On connaît un algorithme qui résout le problème.

? On ne connaît pas d'algorithme qui résout le problème, et on ne sait pas s'il en existe.

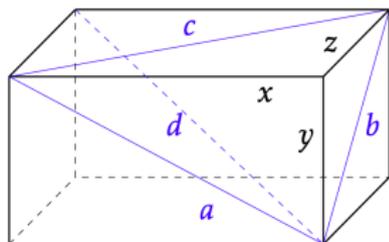
Statut des problèmes en informatique

- 😊 On connaît un algorithme qui résout le problème.
- ? On ne connaît pas d'algorithme qui résout le problème, et on ne sait pas s'il en existe.
- 😞 On **sait** qu'il **n'existe pas** d'algorithme pour ce problème (avec notre compréhension actuelle de ce qu'est un algorithme).

Problème 5 : équations Diophantiennes

Exemple, *brique parfaite d'Euler* (dessin issu du *polycopié de Cyril Gavoille*).

Existe-t-il un parallélépipède aux dimensions indiquées **entières** ?

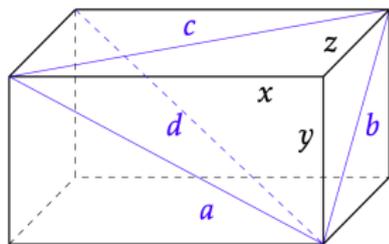


$$\begin{cases} a^2 = x^2 + y^2 \\ b^2 = y^2 + z^2 \\ c^2 = z^2 + x^2 \\ d^2 = x^2 + y^2 + z^2 \end{cases}$$

Problème 5 : équations Diophantiennes

Exemple, *brique parfaite d'Euler* (dessin issu du *polycopié de Cyril Gavoille*).

Existe-t-il un parallélépipède aux dimensions indiquées **entières** ?



$$\begin{cases} a^2 = x^2 + y^2 \\ b^2 = y^2 + z^2 \\ c^2 = z^2 + x^2 \\ d^2 = x^2 + y^2 + z^2 \end{cases}$$

Autrement dit, l'équation Diophantienne

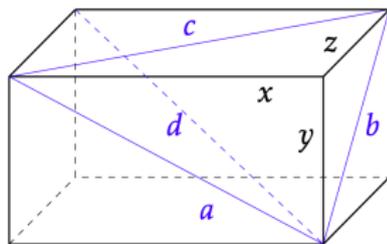
$$(a^2 - x^2 - y^2)^2 + (b^2 - y^2 - z^2)^2 + (c^2 - z^2 - x^2)^2 + (d^2 - x^2 - y^2 - z^2)^2 = 0$$

a-t-elle une solution pour $a, b, c, d, x, y, z \in \mathbb{N}^*$?

Problème 5 : équations Diophantiennes

Exemple, brique parfaite d'Euler (dessin issu du polycopié de Cyril Gavoille).

Existe-t-il un parallélépipède aux dimensions indiquées **entières** ?



$$\begin{cases} a^2 = x^2 + y^2 \\ b^2 = y^2 + z^2 \\ c^2 = z^2 + x^2 \\ d^2 = x^2 + y^2 + z^2 \end{cases}$$

Autrement dit, l'équation Diophantienne

$$(a^2 - x^2 - y^2)^2 + (b^2 - y^2 - z^2)^2 + (c^2 - z^2 - x^2)^2 + (d^2 - x^2 - y^2 - z^2)^2 = 0$$

a-t-elle une solution pour $a, b, c, d, x, y, z \in \mathbb{N}^*$?

C'est une **question ouverte** !

Problème 5 : équations Diophantiennes

Autres exemples

- $x^3 + y^3 + z^3 = 43$?

Problème 5 : équations Diophantiennes

Autres exemples

- $x^3 + y^3 + z^3 = 43?$

Facile : $x = y = 2, z = 3$.

Problème 5 : équations Diophantiennes

Autres exemples

- $x^3 + y^3 + z^3 = 43?$

Facile : $x = y = 2, z = 3$.

- $x^3 + y^3 + z^3 = 42?$

Problème 5 : équations Diophantiennes

Autres exemples

- $x^3 + y^3 + z^3 = 43?$

Facile : $x = y = 2, z = 3$.

- $x^3 + y^3 + z^3 = 42?$

Solution

$$x = -80538738812075974$$

$$y = 80435758145817515$$

$$z = 12602123297335631$$

Problème 5 : équations Diophantiennes

Autres exemples

- $x^3 + y^3 + z^3 = 43?$

Facile : $x = y = 2, z = 3$.

- $x^3 + y^3 + z^3 = 42?$

Booker, Sutherland, 9/2019.

Solution

$$x = -80538738812075974$$

$$y = 80435758145817515$$

$$z = 12602123297335631$$

- Calcul fait (après analyse) sur [Charity Engine](#).

- Plus de [un million d'heures](#) de temps CPU (soit \approx **114 années**!).

Problème 5 : équations Diophantiennes

Autres exemples

○ $x^3 + y^3 + z^3 = 43?$

Facile : $x = y = 2, z = 3$.

○ $x^3 + y^3 + z^3 = 42?$

Booker, Sutherland, 9/2019.

Solution

$x = -80538738812075974$

$y = 80435758145817515$

$z = 12602123297335631$

- Calcul fait (après analyse) sur [Charity Engine](#).
- Plus de **un million d'heures** de temps CPU (soit \approx **114 années**!).
- Ce n'est **pas** une simple triple boucle ! Entier précédent, **33**, mars 2019.

A. R. Booker, 33



A. R. Booker, 42

Historique rapide

1900 Hilbert, 10^e problème : peut-on **décider**, de façon **mécanique**, si une équation Diophantienne a une solution en nombres entiers ?

Historique rapide

1900 Hilbert, 10^e problème : peut-on **décider**, de façon **mécanique**, si une équation Diophantienne a une solution en nombres entiers ?

$$x^7 = 16x^4 - 4x^3 + 5x + 2020$$

Historique rapide

1900 Hilbert, 10^e problème : peut-on **décider**, de façon **mécanique**, si une équation Diophantienne a une solution en nombres entiers ?

$$x^7 = 16x^4 - 4x^3 + 5x + 2020$$

😊 Solutions $\leq 7 \times 2020 \leftrightarrow$ **nombre fini de solutions potentielles**
 \hookrightarrow Résolution automatique d'équations Diophantiennes à 1 variable.

Historique rapide

1900 Hilbert, 10^e problème : peut-on **décider**, de façon **mécanique**, si une équation Diophantienne a une solution en nombres entiers ?

$$x^7y^2 + 16xyz^4 = 4xz^5 + 5xyz + 2020 \quad \text{????}$$

Historique rapide

- 1900 Hilbert, 10^e problème : peut-on **décider**, de façon **mécanique**, si une équation Diophantienne a une solution en nombres entiers ?
- 1928 Hilbert *Entscheidungsproblem*. Peut-on mécaniser **les maths** (1^{er} ordre) ?

Historique rapide

- 1900 **Hilbert**, 10^e problème : peut-on **décider**, de façon **mécanique**, si une équation Diophantienne a une solution en nombres entiers ?
- 1928 **Hilbert** *Entscheidungsproblem*. Peut-on mécaniser **les maths** (1^{er} ordre) ?
- 1935 **Turing** formalise une définition de machine et de calcul.
- 1936 **Turing** prouve que le **problème de l'arrêt** est **indécidable**.
- 1936 **Church** exhibe un problème non résoluble par machine.

Historique rapide

- 1900 **Hilbert**, 10^e problème : peut-on **décider**, de façon **mécanique**, si une équation Diophantienne a une solution en nombres entiers ?
- 1928 **Hilbert** *Entscheidungsproblem*. Peut-on mécaniser **les maths** (1^{er} ordre) ?
- 1935 **Turing** formalise une définition de machine et de calcul.
- 1936 **Turing** prouve que le **problème de l'arrêt** est **indécidable**.
- 1936 **Church** exhibe un problème non résoluble par machine.
- 1938 **Kleene** équivalence machines de Turing, λ -calcul, fonctions récursives

Historique rapide

- 1900 **Hilbert**, 10^e problème : peut-on **décider**, de façon **mécanique**, si une équation Diophantienne a une solution en nombres entiers ?
- 1928 **Hilbert** *Entscheidungsproblem*. Peut-on mécaniser **les maths** (1^{er} ordre) ?
- 1935 **Turing** formalise une définition de machine et de calcul.
- 1936 **Turing** prouve que le **problème de l'arrêt** est **indécidable**.
- 1936 **Church** exhibe un problème non résoluble par machine.
- 1938 **Kleene** équivalence machines de Turing, λ -calcul, fonctions récursives
- 1947 **Post & Markov** prouvent qu'un problème posé par **Thue** en 1914 n'est pas résoluble mécaniquement.
- 1970 **Matiyasevich** répond **négativement** au 10^e problème de Hilbert.