laboratoire de mathématiques de besançon
UNIVERSITÉ DE FRANCHE-COMTÉ • CNRS • UMR 6623

Séminaires croisés
⟨ LMB | FEMTO ⟩

-

Aspects théoriques et explicites de l'arithmétique et application à la cryptographie

Christophe Delaunay

Mardi 2 juillet 2013

Méthodes explicites

▷ La théorie des nombres est un domaine des mathématiques pures qui possède une forte dimension expérimentale.

▷ Nombreux calculs pour :

▷ Plusieurs outils : Kant, Magma, PARI/GP, Sage.

Méthodes explicites

▷ La théorie des nombres est un domaine des mathématiques pures qui possède une forte dimension expérimentale.

▷ Nombreux calculs pour :

1. énoncer et affiner des conjectures (ex. Birch et Swinnerton-Dyer) ;
2. trouver des solutions à des équations (ex. $30 = 2220422932^3 + (-2218888517)^3 + (-283059965)^3$) ;
3. vérifier (ou contredire) des conjectures (ex. hyp. de Rieman) ;
4. compléter une démonstration (ex. Conjecture de Goldbach) ;
5. se faire une idée, une intuition, etc.

▷ Plusieurs outils : Kant, Magma, PARI/GP, Sage.

Méthodes explicites

▷ La théorie des nombres est un domaine des mathématiques pures qui possède une forte dimension expérimentale.

▷ Nombreux calculs pour :

1. énoncer et affiner des conjectures (ex. Birch et Swinnerton-Dyer) ;
2. trouver des solutions à des équations (ex. $30 = 2220422932^3 + (-2218888517)^3 + (-283059965)^3$) ;
3. vérifier (ou contredire) des conjectures (ex. hyp. de Rieman) ;
4. compléter une démonstration (ex. Conjecture de Goldbach) ;
5. se faire une idée, une intuition, etc.

▷ Plusieurs outils : Kant, Magma, PARI/GP, Sage.

Méthodes explicites

▷ La théorie des nombres est un domaine des mathématiques pures qui possède une forte dimension expérimentale.

▷ Nombreux calculs pour :

1. énoncer et affiner des conjectures (ex. Birch et Swinnerton-Dyer) ;
2. trouver des solutions à des équations (ex. $30 = 2220422932^3 + (-2218888517)^3 + (-283059965)^3$) ;
3. vérifier (ou contredire) des conjectures (ex. hyp. de Rieman) ;
4. compléter une démonstration (ex. Conjecture de Goldbach) ;
5. se faire une idée, une intuition, etc.

▷ Plusieurs outils : Kant, Magma, PARI/GP, Sage.

Méthodes explicites

▷ La théorie des nombres est un domaine des mathématiques pures qui possède une forte dimension expérimentale.

▷ Nombreux calculs pour :

1. énoncer et affiner des conjectures (ex. Birch et Swinnerton-Dyer) ;
2. trouver des solutions à des équations (ex. $30 = 2220422932^3 + (-2218888517)^3 + (-283059965)^3$) ;
3. vérifier (ou contredire) des conjectures (ex. hyp. de Rieman) ;
4. compléter une démonstration (ex. Conjecture de Goldbach) ;
5. se faire une idée, une intuition, etc.

▷ Plusieurs outils : Kant, Magma, PARI/GP, Sage.

Méthodes explicites

▷ La théorie des nombres est un domaine des mathématiques pures qui possède une forte dimension expérimentale.

▷ Nombreux calculs pour :

1. énoncer et affiner des conjectures (ex. Birch et Swinnerton-Dyer) ;
2. trouver des solutions à des équations (ex. $30 = 2220422932^3 + (-2218888517)^3 + (-283059965)^3$) ;
3. vérifier (ou contredire) des conjectures (ex. hyp. de Rieman) ;
4. compléter une démonstration (ex. Conjecture de Goldbach) ;
5. se faire une idée, une intuition, etc.

▷ Plusieurs outils : Kant, Magma, PARI/GP, Sage.

Méthodes explicites

▷ La théorie des nombres est un domaine des mathématiques pures qui possède une forte dimension expérimentale.

▷ Nombreux calculs pour :

1. énoncer et affiner des conjectures (ex. Birch et Swinnerton-Dyer) ;
2. trouver des solutions à des équations (ex. $30 = 2220422932^3 + (-2218888517)^3 + (-283059965)^3$) ;
3. vérifier (ou contredire) des conjectures (ex. hyp. de Rieman) ;
4. compléter une démonstration (ex. Conjecture de Goldbach) ;
5. se faire une idée, une intuition, etc.

▷ Plusieurs outils : Kant, Magma, PARI/GP, Sage.

Méthodes explicites

▷ La théorie des nombres est un domaine des mathématiques pures qui possède une forte dimension expérimentale.

▷ Nombreux calculs pour :

1. énoncer et affiner des conjectures (ex. Birch et Swinnerton-Dyer) ;
2. trouver des solutions à des équations (ex. $30 = 2220422932^3 + (-2218888517)^3 + (-283059965)^3$) ;
3. vérifier (ou contredire) des conjectures (ex. hyp. de Rieman) ;
4. compléter une démonstration (ex. Conjecture de Goldbach) ;
5. se faire une idée, une intuition, etc.

▷ Plusieurs outils : Kant, Magma, PARI/GP, Sage.

Méthodes explicites : exemple récent

▷ **Conjecture de Goldbach(1742)** : tout nombre pair ≥ 4 est la somme de 2 nombres premiers.

$\rightsquigarrow$ Vérifications numériques (2010) $\approx 10^{18}$.

▷ **Conjecture de Goldbach ternaire** : tout nombre impair ≥ 7 est somme de 3 nombres premiers.

$\rightsquigarrow$ Conséquence de la conjecture de Goldbach.

▷ **Théorème de Vinogradov (1937)** : la conjecture de Goldbach ternaire est vraie pour tout $n \geq e^{3100}$.

▷ **Théorème de Helfgott (mai 2013)** : la conjecture de Goldbach ternaire est vraie.

$\rightsquigarrow$ arguments théoriques pour $n \geq 10^{29}$ ($\approx$ 100 pages).

$\rightsquigarrow$ vérification numériques (arithmétique des intervalles) avec Platt.

Méthodes explicites : exemple récent

▷ **Conjecture de Goldbach(1742)** : tout nombre pair ≥ 4 est la somme de 2 nombres premiers.

$\rightsquigarrow$ Vérifications numériques (2010) $\approx 10^{18}$.

▷ **Conjecture de Goldbach ternaire** : tout nombre impair ≥ 7 est somme de 3 nombres premiers.

$\rightsquigarrow$ Conséquence de la conjecture de Goldbach.

▷ **Théorème de Vinogradov (1937)** : la conjecture de Goldbach ternaire est vraie pour tout $n \geq e^{3100}$.

▷ **Théorème de Helfgott (mai 2013)** : la conjecture de Goldbach ternaire est vraie.

$\rightsquigarrow$ arguments théoriques pour $n \geq 10^{29}$ ($\approx$ 100 pages).

$\rightsquigarrow$ vérification numériques (arithmétique des intervalles) avec Platt.

Méthodes explicites : exemple récent

▷ **Conjecture de Goldbach(1742)** : tout nombre pair ≥ 4 est la somme de 2 nombres premiers.

$\rightsquigarrow$ Vérifications numériques (2010) $\approx 10^{18}$.

▷ **Conjecture de Goldbach ternaire** : tout nombre impair ≥ 7 est somme de 3 nombres premiers.

$\rightsquigarrow$ Conséquence de la conjecture de Goldbach.

▷ **Théorème de Vinogradov (1937)** : la conjecture de Goldbach ternaire est vraie pour tout $n \geq e^{3100}$.

▷ **Théorème de Helfgott (mai 2013)** : la conjecture de Goldbach ternaire est vraie.

$\rightsquigarrow$ arguments théoriques pour $n \geq 10^{29}$ ($\approx$ 100 pages).

$\rightsquigarrow$ vérification numériques (arithmétique des intervalles) avec Platt.

Méthodes explicites : exemple récent

▷ **Conjecture de Goldbach(1742)** : tout nombre pair ≥ 4 est la somme de 2 nombres premiers.

$\rightsquigarrow$ Vérifications numériques (2010) $\approx 10^{18}$.

▷ **Conjecture de Goldbach ternaire** : tout nombre impair ≥ 7 est somme de 3 nombres premiers.

$\rightsquigarrow$ Conséquence de la conjecture de Goldbach.

▷ **Théorème de Vinogradov (1937)** : la conjecture de Goldbach ternaire est vraie pour tout $n \geq e^{3100}$.

▷ **Théorème de Helfgott (mai 2013)** : la conjecture de Goldbach ternaire est vraie.

$\rightsquigarrow$ arguments théoriques pour $n \geq 10^{29}$ ($\approx$ 100 pages).

$\rightsquigarrow$ vérification numériques (arithmétique des intervalles) avec Platt.

Méthodes explicites : exemple récent

▷ **Conjecture de Goldbach(1742)** : tout nombre pair ≥ 4 est la somme de 2 nombres premiers.

$\rightsquigarrow$ Vérifications numériques (2010) $\approx 10^{18}$.

▷ **Conjecture de Goldbach ternaire** : tout nombre impair ≥ 7 est somme de 3 nombres premiers.

$\rightsquigarrow$ Conséquence de la conjecture de Goldbach.

▷ **Théorème de Vinogradov (1937)** : la conjecture de Goldbach ternaire est vraie pour tout $n \geq e^{3100}$.

▷ **Théorème de Helfgott (mai 2013)** : la conjecture de Goldbach ternaire est vraie.

$\rightsquigarrow$ arguments théoriques pour $n \geq 10^{29}$ ($\approx$ 100 pages).

$\rightsquigarrow$ vérification numériques (arithmétique des intervalles) avec Platt.

Méthodes explicites : exemple récent

▷ **Conjecture de Goldbach(1742)** : tout nombre pair ≥ 4 est la somme de 2 nombres premiers.

$\rightsquigarrow$ Vérifications numériques (2010) $\approx 10^{18}$.

▷ **Conjecture de Goldbach ternaire** : tout nombre impair ≥ 7 est somme de 3 nombres premiers.

$\rightsquigarrow$ Conséquence de la conjecture de Goldbach.

▷ **Théorème de Vinogradov (1937)** : la conjecture de Goldbach ternaire est vraie pour tout $n \geq e^{3100}$.

▷ **Théorème de Helfgott (mai 2013)** : la conjecture de Goldbach ternaire est vraie.

$\rightsquigarrow$ arguments théoriques pour $n \geq 10^{29}$ ($\approx$ 100 pages).

$\rightsquigarrow$ vérification numériques (arithmétique des intervalles) avec Platt.

Méthodes explicites : exemple récent

▷ **Conjecture de Goldbach(1742)** : tout nombre pair ≥ 4 est la somme de 2 nombres premiers.

$\rightsquigarrow$ Vérifications numériques (2010) $\approx 10^{18}$.

▷ **Conjecture de Goldbach ternaire** : tout nombre impair ≥ 7 est somme de 3 nombres premiers.

$\rightsquigarrow$ Conséquence de la conjecture de Goldbach.

▷ **Théorème de Vinogradov (1937)** : la conjecture de Goldbach ternaire est vraie pour tout $n \geq e^{3100}$.

▷ **Théorème de Helfgott (mai 2013)** : la conjecture de Goldbach ternaire est vraie.

$\rightsquigarrow$ arguments théoriques pour $n \geq 10^{29}$ ($\approx$ 100 pages).

$\rightsquigarrow$ vérification numériques (arithmétique des intervalles) avec Platt.

Méthodes explicites : exemple récent

▷ **Conjecture de Goldbach(1742)** : tout nombre pair ≥ 4 est la somme de 2 nombres premiers.

$\rightsquigarrow$ Vérifications numériques (2010) $\approx 10^{18}$.

▷ **Conjecture de Goldbach ternaire** : tout nombre impair ≥ 7 est somme de 3 nombres premiers.

$\rightsquigarrow$ Conséquence de la conjecture de Goldbach.

▷ **Théorème de Vinogradov (1937)** : la conjecture de Goldbach ternaire est vraie pour tout $n \geq e^{3100}$.

▷ **Théorème de Helfgott (mai 2013)** : la conjecture de Goldbach ternaire est vraie.

$\rightsquigarrow$ arguments théoriques pour $n \geq 10^{29}$ ($\approx$ 100 pages).

$\rightsquigarrow$ vérification numériques (arithmétique des intervalles) avec Platt.

Méthodes explicites : exemple récent

▷ **Conjecture de Goldbach(1742)** : tout nombre pair ≥ 4 est la somme de 2 nombres premiers.

$\rightsquigarrow$ Vérifications numériques (2010) $\approx 10^{18}$.

▷ **Conjecture de Goldbach ternaire** : tout nombre impair ≥ 7 est somme de 3 nombres premiers.

$\rightsquigarrow$ Conséquence de la conjecture de Goldbach.

▷ **Théorème de Vinogradov (1937)** : la conjecture de Goldbach ternaire est vraie pour tout $n \geq e^{3100}$.

▷ **Théorème de Helfgott (mai 2013)** : la conjecture de Goldbach ternaire est vraie.

$\rightsquigarrow$ arguments théoriques pour $n \geq 10^{29}$ ($\approx$ 100 pages).

$\rightsquigarrow$ vérification numériques (arithmétique des intervalles) avec Platt.

Principes et buts de la cryptologie

1.
2.
3.
4.

Principes et buts de la cryptologie

1. Confidentialité : le texte codé ne peut pas être lu par un intrus.
2. Authentification : Le destinataire doit être sûr de l'auteur du message.
3. Intégrité : le message n'a pas été modifié pendant la transmission.
4. La non-répudiation : l'expéditeur ne peut pas nier être l'auteur du message.
5. Et quelques autres... ex. : jouer à pile ou face par téléphone.

La cryptologie moderne répond à ces demandes.

Principes et buts de la cryptologie

1. Confidentialité : le texte codé ne peut pas être lu par un intrus.
2. Authentification : Le destinataire doit être sûr de l'auteur du message.
3. Intégrité : le message n'a pas été modifié pendant la transmission.
4. La non-répudiation : l'expéditeur ne peut pas nier être l'auteur du message.
5. Et quelques autres... ex. : jouer à pile ou face par téléphone.

La cryptologie moderne répond à ces demandes.

Principes et buts de la cryptologie

1. Confidentialité : le texte codé ne peut pas être lu par un intrus.
2. Authentification : Le destinataire doit être sûr de l'auteur du message.
3. Intégrité : le message n'a pas été modifié pendant la transmission.
4. La non-répudiation : l'expéditeur ne peut pas nier être l'auteur du message.
5. Et quelques autres... ex. : jouer à pile ou face par téléphone.

La cryptologie moderne répond à ces demandes.

Principes et buts de la cryptologie

1. Confidentialité : le texte codé ne peut pas être lu par un intrus.
2. Authentification : Le destinataire doit être sûr de l'auteur du message.
3. Intégrité : le message n'a pas été modifié pendant la transmission.
4. La non-répudiation : l'expéditeur ne peut pas nier être l'auteur du message.
5. Et quelques autres... ex. : jouer à pile ou face par téléphone.

La cryptologie moderne répond à ces demandes.

Principes et buts de la cryptologie

1. Confidentialité : le texte codé ne peut pas être lu par un intrus.
2. Authentification : Le destinataire doit être sûr de l'auteur du message.
3. Intégrité : le message n'a pas été modifié pendant la transmission.
4. La non-répudiation : l'expéditeur ne peut pas nier être l'auteur du message.
5. Et quelques autres... ex. : jouer à pile ou face par téléphone.

La cryptologie moderne répond à ces demandes.

Principes et buts de la cryptologie

1. Confidentialité : le texte codé ne peut pas être lu par un intrus.
2. Authentification : Le destinataire doit être sûr de l'auteur du message.
3. Intégrité : le message n'a pas été modifié pendant la transmission.
4. La non-répudiation : l'expéditeur ne peut pas nier être l'auteur du message.
5. Et quelques autres... ex. : jouer à pile ou face par téléphone.

La cryptologie moderne répond à ces demandes.

Principes et buts de la cryptologie

1. Confidentialité : le texte codé ne peut pas être lu par un intrus.
2. Authentification : Le destinataire doit être sûr de l'auteur du message.
3. Intégrité : le message n'a pas été modifié pendant la transmission.
4. La non-répudiation : l'expéditeur ne peut pas nier être l'auteur du message.
5. Et quelques autres... ex. : jouer à pile ou face par téléphone.

La cryptologie moderne répond à ces demandes.

Principes et buts de la cryptologie

1. Confidentialité : le texte codé ne peut pas être lu par un intrus.
2. Authentification : Le destinataire doit être sûr de l'auteur du message.
3. Intégrité : le message n'a pas été modifié pendant la transmission.
4. La non-répudiation : l'expéditeur ne peut pas nier être l'auteur du message.
5. Et quelques autres... ex. : jouer à pile ou face par téléphone.

La cryptologie moderne répond à ces demandes.

Les 2 familles de cryptographie

1. La cryptographie à clef secrète (ou symétrique).

   PRINCIPE : L'émetteur et le récepteur partagent le même secret pour crypter et décrypter.

2. La cryptographie à clef publique (ou asymétrique).

   PRINCIPE : Une clé publique (tout le monde la connaît) pour crypter. Une clé secrète (seulement) détenu par le récepteur permet de décrypter.

Les 2 familles de cryptographie

1. La cryptographie à clef secrète (ou symétrique).

   PRINCIPE : L'émetteur et le récepteur partagent le même secret pour crypter et décrypter.

2. La cryptographie à clef publique (ou asymétrique).

   PRINCIPE : Une clé publique (tout le monde la connaît) pour crypter. Une clé secrète (seulement) détenu par le récepteur permet de décrypter.

Les 2 familles de cryptographie

1. La cryptographie à clef secrète (ou symétrique).

   PRINCIPE : L'émetteur et le récepteur partagent le même secret pour crypter et décrypter.

2. La cryptographie à clef publique (ou asymétrique).

   PRINCIPE : Une clé publique (tout le monde la connaît) pour crypter. Une clé secrète (seulement) détenu par le récepteur permet de décrypter.

Les 2 familles de cryptographie

1. La cryptographie à clef secrète (ou symétrique).

   PRINCIPE : L'émetteur et le récepteur partagent le même secret pour crypter et décrypter.

2. La cryptographie à clef publique (ou asymétrique).

   PRINCIPE : Une clé publique (tout le monde la connaît) pour crypter. Une clé secrète (seulement) détenu par le récepteur permet de décrypter.

Les 2 familles de cryptographie

1. La cryptographie à clef secrète (ou symétrique).

   PRINCIPE : L'émetteur et le récepteur partagent le même secret pour crypter et décrypter.

2. La cryptographie à clef publique (ou asymétrique).

   PRINCIPE : Une clé publique (tout le monde la connaît) pour crypter. Une clé secrète (seulement) détenu par le récepteur permet de décrypter.

La cryptographie à clef publique

PRINCIPE : Une clef pour coder et un secret pour décoder.

$\rightsquigarrow$ Pour créer un tel système, il faut une fonction à sens unique :

C'est une fonction f telle que :

- Il est facile de calculer $f(x)$.
- Connaissant $f(x)$, il est difficile de trouver x.

Premier système apparu $\approx$ 1977 :

Il s'agit du système R.S.A.

Inventé par Rivest, Shamir, Adleman.

Autres systèmes

- Cryptosystème de ElGamal.
- DSA : Digital Signature Algorithm.
- ECDSA : Elliptic Curve Digital Signature Algorithm.
- etc.

La cryptographie à clef publique

PRINCIPE : Une clef pour coder et un secret pour décoder.

$\rightsquigarrow$ Pour créer un tel système, il faut une fonction à sens unique :

C'est une fonction f telle que :

- Il est facile de calculer $f(x)$.
- Connaissant $f(x)$, il est difficile de trouver x.

Premier système apparu $\approx$ 1977 :

Il s'agit du système R.S.A.

Inventé par Rivest, Shamir, Adleman.

Autres systèmes

- Cryptosystème de ElGamal.
- DSA : Digital Signature Algorithm.
- ECDSA : Elliptic Curve Digital Signature Algorithm.
- etc.

La cryptographie à clef publique

PRINCIPE : Une clef pour coder et un secret pour décoder.

$\rightsquigarrow$ Pour créer un tel système, il faut une fonction à sens unique :

C'est une fonction f telle que :

- Il est facile de calculer $f(\mathbf{x})$.
- Connaissant $f(\mathbf{x})$, il est difficile de trouver $\mathbf{x}$.

Premier système apparu $\approx$ 1977 :

Il s'agit du système R.S.A.

Inventé par Rivest, Shamir, Adleman.

Autres systèmes

- Cryptosystème de ElGamal.
- DSA : Digital Signature Algorithm.
- ECDSA : Elliptic Curve Digital Signature Algorithm.
- etc.

La cryptographie à clef publique

PRINCIPE : Une clef pour coder et un secret pour décoder.

$\rightsquigarrow$ Pour créer un tel système, il faut une fonction à sens unique :

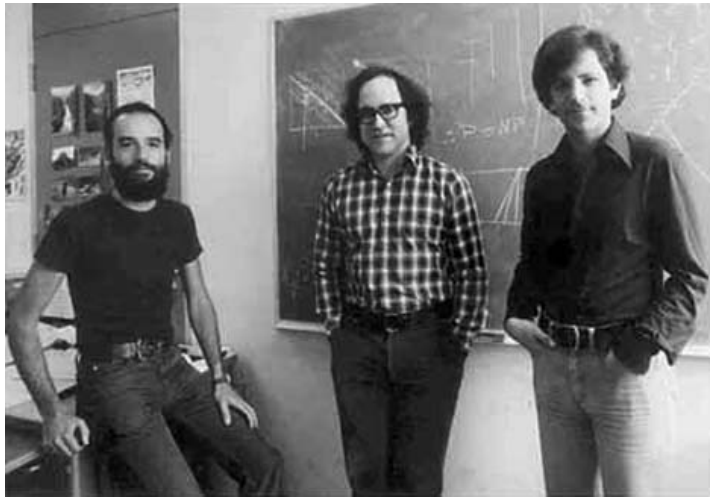
C'est une fonction f telle que :

- Il est facile de calculer $f(\mathbf{x})$.
- Connaissant $f(\mathbf{x})$, il est difficile de trouver $\mathbf{x}$.

Premier système apparu $\approx$ 1977 :

Il s'agit du système R.S.A.

Inventé par Rivest, Shamir, Adleman.



Autres systèmes

- Cryptosystème de ElGamal.
- DSA : Digital Signature Algorithm.
- ECDSA : Elliptic Curve Digital Signature Algorithm.
- etc.

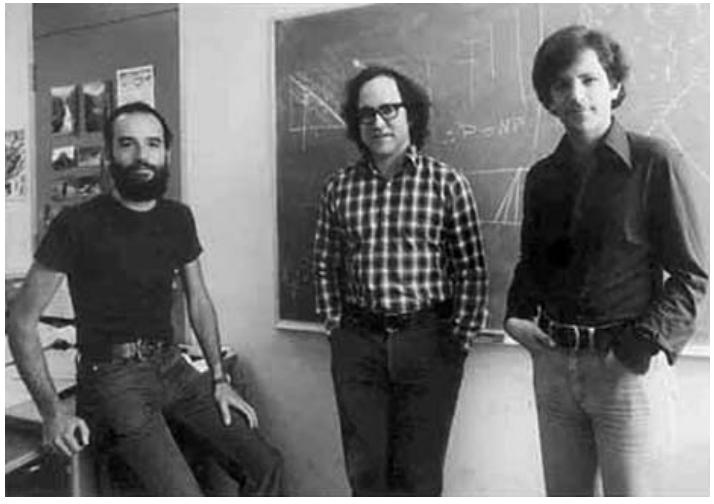
La cryptographie à clef publique

PRINCIPE : Une clef pour coder et un secret pour décoder.

$\rightsquigarrow$ Pour créer un tel système, il faut une fonction à sens unique :

C'est une fonction f telle que :

- Il est facile de calculer $f(\mathbf{x})$.
- Connaissant $f(\mathbf{x})$, il est difficile de trouver $\mathbf{x}$.



Premier système apparu $\approx$ 1977 :

Il s'agit du système R.S.A.

Inventé par Rivest, Shamir, Adleman.

Autres systèmes

- Cryptosystème de ElGamal.
- DSA : Digital Signature Algorithm.
- ECDSA : Elliptic Curve Digital Signature Algorithm.
- etc.

Facile ? Pas facile ?

$\rightsquigarrow$ Notion de complexité d'un algorithme / programme.

Entrée d'un programme est ℓ :

- ▷ Calcul en $O(\log(\ell)) = O(e^{\log\log \ell})$ $\rightarrow$ linéaire (très facile).
- ▷ Calcul en $O(\log(\ell)^k) = O(e^{k\log\log \ell})$ $\rightarrow$ polynomial (facile).
- ▷ Calcul en $O(e^{k\log(\ell)^a(\log\log \ell)^{1-a}})$ $\rightarrow$ sous-exponentiel (compliqué).
- ▷ Calcul en $O(\ell) = O(e^{\log \ell})$ $\rightarrow$ exponentiel (très compliqué).

Exemple : soient G un groupe, $g \in G \rightsquigarrow$ Pour $\ell \in \mathbb{Z}$, calculer g^ℓ.

$\rightsquigarrow$ Exponentiation naïve : faire $g \times g \cdots \times g$

$\rightsquigarrow$ Exponentiation rapide : utiliser le principe suivant :

$$g^\ell = \begin{cases} \left(g^{\frac{\ell}{2}}\right)^2 & \text{si } \ell \text{ est pair} \\ g \cdot \left(g^{\frac{\ell-1}{2}}\right)^2 & \text{si } \ell \text{ est impair} \end{cases}$$

et on réitère.

Facile ? Pas facile ?

$\rightsquigarrow$ Notion de complexité d'un algorithme / programme.

Entrée d'un programme est ℓ :

- ▷ Calcul en $O(\log(\ell)) = O(e^{\log\log \ell})$ $\rightarrow$ linéaire (très facile).
- ▷ Calcul en $O(\log(\ell)^k) = O(e^{k\log\log \ell})$ $\rightarrow$ polynomial (facile).
- ▷ Calcul en $O(e^{k\log(\ell)^a(\log\log \ell)^{1-a}})$ $\rightarrow$ sous-exponentiel (compliqué).
- ▷ Calcul en $O(\ell) = O(e^{\log \ell})$ $\rightarrow$ exponentiel (très compliqué).

Exemple : soient G un groupe, $g \in G \rightsquigarrow$ Pour $\ell \in \mathbb{Z}$, calculer g^ℓ.

$\rightsquigarrow$ Exponentiation naïve : faire $g \times g \cdots \times g$

$\rightsquigarrow$ Exponentiation rapide : utiliser le principe suivant :

$$g^\ell = \begin{cases} \left(g^{\frac{\ell}{2}}\right)^2 & \text{si } \ell \text{ est pair} \\ g \cdot \left(g^{\frac{\ell-1}{2}}\right)^2 & \text{si } \ell \text{ est impair} \end{cases}$$

et on réitère.

Facile ? Pas facile ?

$\rightsquigarrow$ Notion de complexité d'un algorithme / programme.

Entrée d'un programme est ℓ : complexité $=$ le nbr. de multiplications.

- $\triangleright$ Calcul en $O(\log(\ell)) = O(e^{\log\log\ell})$ $\rightarrow$ linéaire (très facile).
- $\triangleright$ Calcul en $O(\log(\ell)^k) = O(e^{k\log\log\ell})$ $\rightarrow$ polynomial (facile).
- $\triangleright$ Calcul en $O(e^{k\log(\ell)^a(\log\log\ell)^{1-a}})$ $\rightarrow$ sous-exponentiel (compliqué).
- $\triangleright$ Calcul en $O(\ell) = O(e^{\log\ell})$ $\rightarrow$ exponentiel (très compliqué).

Exemple : soient G un groupe, $g \in G \rightsquigarrow$ Pour $\ell \in \mathbb{Z}$, calculer g^ℓ.

$\rightsquigarrow$ Exponentiation naïve : faire $g \times g \cdots \times g$

$\rightsquigarrow$ Exponentiation rapide : utiliser le principe suivant :

$$g^\ell = \begin{cases} \left(g^{\frac{\ell}{2}}\right)^2 & \text{si } \ell \text{ est pair} \\ g \cdot \left(g^{\frac{\ell-1}{2}}\right)^2 & \text{si } \ell \text{ est impair} \end{cases}$$

et on réitère.

Facile ? Pas facile ?

$\rightsquigarrow$ Notion de complexité d'un algorithme / programme.

Entrée d'un programme est ℓ : complexité $=$ le nbr. de multiplications.

$\triangleright$ Calcul en $O(\log(\ell)) = O(e^{\log\log\ell})$ $\rightarrow$ linéaire (très facile).

$\triangleright$ Calcul en $O(\log(\ell)^k) = O(e^{k\log\log\ell})$ $\rightarrow$ polynomial (facile).

$\triangleright$ Calcul en $O(e^{k\log(\ell)^a(\log\log\ell)^{1-a}})$ $\rightarrow$ sous-exponentiel (compliqué).

$\triangleright$ Calcul en $O(\ell) = O(e^{\log\ell})$ $\rightarrow$ exponentiel (très compliqué).

Exemple : soient G un groupe, $g \in G \rightsquigarrow$ Pour $\ell \in \mathbb{Z}$, calculer g^ℓ.

$\rightsquigarrow$ Exponentiation naïve : faire $g \times g \cdots \times g$

$\rightsquigarrow$ Exponentiation rapide : utiliser le principe suivant :

$$g^\ell = \begin{cases} \left(g^{\frac{\ell}{2}}\right)^2 & \text{si } \ell \text{ est pair} \\ g \cdot \left(g^{\frac{\ell-1}{2}}\right)^2 & \text{si } \ell \text{ est impair} \end{cases}$$

et on réitère.

Facile ? Pas facile ?

$\rightsquigarrow$ Notion de complexité d'un algorithme / programme.

Entrée d'un programme est ℓ : complexité $=$ le nbr. de multiplications.

▷ Calcul en $O(\log(\ell)) = O(e^{\log\log\ell})$ $\rightarrow$ linéaire (très facile).

▷ Calcul en $O(\log(\ell)^k) = O(e^{k\log\log\ell})$ $\rightarrow$ polynomial (facile).

▷ Calcul en $O(e^{k\log(\ell)^a(\log\log\ell)^{1-a}})$ $\rightarrow$ sous-exponentiel (compliqué).

▷ Calcul en $O(\ell) = O(e^{\log\ell})$ $\rightarrow$ exponentiel (très compliqué).

Exemple : soient G un groupe, $g \in G \rightsquigarrow$ Pour $\ell \in \mathbb{Z}$, calculer g^ℓ.

$\rightsquigarrow$ Exponentiation naïve : faire $g \times g \cdots \times g$

$\rightsquigarrow$ Exponentiation rapide : utiliser le principe suivant :

$$g^\ell = \begin{cases} \left(g^{\frac{\ell}{2}}\right)^2 & \text{si } \ell \text{ est pair} \\ g \cdot \left(g^{\frac{\ell-1}{2}}\right)^2 & \text{si } \ell \text{ est impair} \end{cases}$$

et on réitère.

Facile ? Pas facile ?

$\rightsquigarrow$ Notion de complexité d'un algorithme / programme.

Entrée d'un programme est ℓ : complexité $=$ le nbr. de multiplications.

▷ Calcul en $O(\log(\ell)) = O(e^{\log\log\ell})$ $\rightarrow$ linéaire (très facile).

▷ Calcul en $O(\log(\ell)^k) = O(e^{k\log\log\ell})$ $\rightarrow$ polynomial (facile).

▷ Calcul en $O(e^{k\log(\ell)^a(\log\log\ell)^{1-a}})$ $\rightarrow$ sous-exponentiel (compliqué).

▷ Calcul en $O(\ell) = O(e^{\log\ell})$ $\rightarrow$ exponentiel (très compliqué).

Exemple : soient G un groupe, $g \in G \rightsquigarrow$ Pour $\ell \in \mathbb{Z}$, calculer g^ℓ.

$\rightsquigarrow$ Exponentiation naïve : faire $g \times g \cdots \times g$

$\rightsquigarrow$ Exponentiation rapide : utiliser le principe suivant :

$$g^\ell = \begin{cases} \left(g^{\frac{\ell}{2}}\right)^2 & \text{si } \ell \text{ est pair} \\ g \cdot \left(g^{\frac{\ell-1}{2}}\right)^2 & \text{si } \ell \text{ est impair} \end{cases}$$

et on réitère.

Facile ? Pas facile ?

$\rightsquigarrow$ Notion de complexité d'un algorithme / programme.

Entrée d'un programme est ℓ : complexité $=$ le nbr. de multiplications.

- ▷ Calcul en $O(\log(\ell)) = O(e^{\log\log\ell})$ $\rightarrow$ linéaire (très facile).
- ▷ Calcul en $O(\log(\ell)^k) = O(e^{k\log\log\ell})$ $\rightarrow$ polynomial (facile).
- ▷ Calcul en $O(e^{k\log(\ell)^a(\log\log\ell)^{1-a}})$ $\rightarrow$ sous-exponentiel (compliqué).
- ▷ Calcul en $O(\ell) = O(e^{\log\ell})$ $\rightarrow$ exponentiel (très compliqué).

Exemple : soient G un groupe, $g \in G \rightsquigarrow$ Pour $\ell \in \mathbb{Z}$, calculer g^ℓ.

$\rightsquigarrow$ Exponentiation naïve : faire $g \times g \cdots \times g$

$\rightsquigarrow$ Exponentiation rapide : utiliser le principe suivant :

$$g^\ell = \begin{cases} \left(g^{\frac{\ell}{2}}\right)^2 & \text{si } \ell \text{ est pair} \\ g \cdot \left(g^{\frac{\ell-1}{2}}\right)^2 & \text{si } \ell \text{ est impair} \end{cases}$$

et on réitère.

Facile ? Pas facile ?

$\rightsquigarrow$ Notion de complexité d'un algorithme / programme.

Entrée d'un programme est ℓ : complexité $=$ le nbr. de multiplications.

▷ Calcul en $O(\log(\ell)) = O(e^{\log\log\ell})$ $\rightarrow$ linéaire (très facile).

▷ Calcul en $O(\log(\ell)^k) = O(e^{k\log\log\ell})$ $\rightarrow$ polynomial (facile).

▷ Calcul en $O(e^{k\log(\ell)^a(\log\log\ell)^{1-a}})$ $\rightarrow$ sous-exponentiel (compliqué).

▷ Calcul en $O(\ell) = O(e^{\log\ell})$ $\rightarrow$ exponentiel (très compliqué).

Exemple : soient G un groupe, $g \in G$ $\rightsquigarrow$ Pour $\ell \in \mathbb{Z}$, calculer g^ℓ.

$\rightsquigarrow$ Exponentiation naïve : faire $g \times g \cdots \times g$

$\rightsquigarrow$ Exponentiation rapide : utiliser le principe suivant :

$$g^\ell = \begin{cases} \left(g^{\frac{\ell}{2}}\right)^2 & \text{si } \ell \text{ est pair} \\ g \cdot \left(g^{\frac{\ell-1}{2}}\right)^2 & \text{si } \ell \text{ est impair} \end{cases}$$

et on réitère.

Facile ? Pas facile ?

$\rightsquigarrow$ Notion de complexité d'un algorithme / programme.

Entrée d'un programme est ℓ : complexité $=$ le nbr. de multiplications.

▷ Calcul en $O(\log(\ell)) = O(e^{\log\log\ell})$ $\rightarrow$ linéaire (très facile).

▷ Calcul en $O(\log(\ell)^k) = O(e^{k\log\log\ell})$ $\rightarrow$ polynomial (facile).

▷ Calcul en $O(e^{k\log(\ell)^a(\log\log\ell)^{1-a}})$ $\rightarrow$ sous-exponentiel (compliqué).

▷ Calcul en $O(\ell) = O(e^{\log\ell})$ $\rightarrow$ exponentiel (très compliqué).

Exemple : soient G un groupe, $g \in G$ $\rightsquigarrow$ Pour $\ell \in \mathbb{Z}$, calculer g^ℓ.

$\rightsquigarrow$ Exponentiation naïve : faire $g \times g \cdots \times g$ $\rightarrow$ exponentiel.

$\rightsquigarrow$ Exponentiation rapide : utiliser le principe suivant :

$$g^\ell = \begin{cases} \left(g^{\frac{\ell}{2}}\right)^2 & \text{si } \ell \text{ est pair} \\ g \cdot \left(g^{\frac{\ell-1}{2}}\right)^2 & \text{si } \ell \text{ est impair} \end{cases}$$

et on réitère.

Facile ? Pas facile ?

$\rightsquigarrow$ Notion de complexité d'un algorithme / programme.

Entrée d'un programme est ℓ : complexité $=$ le nbr. de multiplications.

▷ Calcul en $O(\log(\ell)) = O(e^{\log\log\ell})$ $\rightarrow$ linéaire (très facile).

▷ Calcul en $O(\log(\ell)^k) = O(e^{k\log\log\ell})$ $\rightarrow$ polynomial (facile).

▷ Calcul en $O(e^{k\log(\ell)^a(\log\log\ell)^{1-a}})$ $\rightarrow$ sous-exponentiel (compliqué).

▷ Calcul en $O(\ell) = O(e^{\log\ell})$ $\rightarrow$ exponentiel (très compliqué).

Exemple : soient G un groupe, $g \in G \rightsquigarrow$ Pour $\ell \in \mathbb{Z}$, calculer g^ℓ.

$\rightsquigarrow$ Exponentiation naïve : faire $g \times g \cdots \times g \rightarrow$ exponentiel.

$\rightsquigarrow$ Exponentiation rapide : utiliser le principe suivant :

$$g^\ell = \begin{cases} \left(g^{\frac{\ell}{2}}\right)^2 & \text{si } \ell \text{ est pair} \\ g \cdot \left(g^{\frac{\ell-1}{2}}\right)^2 & \text{si } \ell \text{ est impair} \end{cases}$$

et on réitère.

Facile ? Pas facile ?

$\rightsquigarrow$ Notion de complexité d'un algorithme / programme.

Entrée d'un programme est ℓ : complexité $=$ le nbr. de multiplications.

$\triangleright$ Calcul en $O(\log(\ell)) = O(e^{\log\log\ell})$ $\rightarrow$ linéaire (très facile).

$\triangleright$ Calcul en $O(\log(\ell)^k) = O(e^{k\log\log\ell})$ $\rightarrow$ polynomial (facile).

$\triangleright$ Calcul en $O(e^{k\log(\ell)^a(\log\log\ell)^{1-a}})$ $\rightarrow$ sous-exponentiel (compliqué).

$\triangleright$ Calcul en $O(\ell) = O(e^{\log\ell})$ $\rightarrow$ exponentiel (très compliqué).

Exemple : soient G un groupe, $g \in G \rightsquigarrow$ Pour $\ell \in \mathbb{Z}$, calculer g^ℓ.

$\rightsquigarrow$ Exponentiation naïve : faire $g \times g \cdots \times g \rightarrow$ exponentiel.

$\rightsquigarrow$ Exponentiation rapide : utiliser le principe suivant :

$$g^\ell = \begin{cases} \left(g^{\frac{\ell}{2}}\right)^2 & \text{si } \ell \text{ est pair} \\ g \cdot \left(g^{\frac{\ell-1}{2}}\right)^2 & \text{si } \ell \text{ est impair} \end{cases}$$

et on réitère. $\rightarrow$ linéaire.

Facile ? Pas facile ?

Entrée d'un programme est ℓ : complexité $=$ le nbr. de multiplications.

▷ Calcul en $O(\log(\ell)) = O(e^{\log\log\ell})$ $\rightarrow$ linéaire (très facile).

▷ Calcul en $O(\log(\ell)^k) = O(e^{k\log\log\ell})$ $\rightarrow$ polynomial (facile).

▷ Calcul en $O(e^{k\log(\ell)^a(\log\log\ell)^{1-a}})$ $\rightarrow$ sous-exponentiel (compliqué).

▷ Calcul en $O(\ell) = O(e^{\log\ell})$ $\rightarrow$ exponentiel (très compliqué).

Exemples :

▷ Calculer $\mathrm{pgcd}(\ell_1, \ell_2)$

▷ Décider si un entier ℓ est premier

▷ Factoriser un entier ℓ

▷ Calculer g^ℓ en faisant $g \times g \cdots \times g$

$\rightsquigarrow$ La multiplication est (probablement) une fonction à sens unique !
($\rightarrow$ RSA).

Facile ? Pas facile ?

Entrée d'un programme est ℓ : complexité $=$ le nbr. de multiplications.

▷ Calcul en $O(\log(\ell)) = O(e^{\log\log\ell})$ $\rightarrow$ linéaire (très facile).

▷ Calcul en $O(\log(\ell)^k) = O(e^{k\log\log\ell})$ $\rightarrow$ polynomial (facile).

▷ Calcul en $O(e^{k\log(\ell)^a(\log\log\ell)^{1-a}})$ $\rightarrow$ sous-exponentiel (compliqué).

▷ Calcul en $O(\ell) = O(e^{\log\ell})$ $\rightarrow$ exponentiel (très compliqué).

Exemples :

▷ Calculer $\mathrm{pgcd}(\ell_1, \ell_2)$

▷ Décider si un entier ℓ est premier

▷ Factoriser un entier ℓ

▷ Calculer g^ℓ en faisant $g \times g \cdots \times g$

$\rightsquigarrow$ La multiplication est (probablement) une fonction à sens unique !
($\rightarrow$ RSA).

Facile ? Pas facile ?

Entrée d'un programme est ℓ : complexité $=$ le nbr. de multiplications.

▷ Calcul en $O(\log(\ell)) = O(e^{\log\log\ell})$ → linéaire (très facile).

▷ Calcul en $O(\log(\ell)^k) = O(e^{k\log\log\ell})$ → polynomial (facile).

▷ Calcul en $O(e^{k\log(\ell)^a(\log\log\ell)^{1-a}})$ → sous-exponentiel (compliqué).

▷ Calcul en $O(\ell) = O(e^{\log\ell})$ → exponentiel (très compliqué).

Exemples :

▷ Calculer $\mathrm{pgcd}(\ell_1, \ell_2)$ → linéaire.

▷ Décider si un entier ℓ est premier

▷ Factoriser un entier ℓ

▷ Calculer g^ℓ en faisant $g \times g \cdot \cdot \cdot \times g$

$\rightsquigarrow$ La multiplication est (probablement) une fonction à sens unique !
($\rightarrow$ RSA).

Facile ? Pas facile ?

Entrée d'un programme est ℓ : complexité $=$ le nbr. de multiplications.

▷ Calcul en $O(\log(\ell)) = O(e^{\log\log\ell})$ → linéaire (très facile).

▷ Calcul en $O(\log(\ell)^k) = O(e^{k\log\log\ell})$ → polynomial (facile).

▷ Calcul en $O(e^{k\log(\ell)^a(\log\log\ell)^{1-a}})$ → sous-exponentiel (compliqué).

▷ Calcul en $O(\ell) = O(e^{\log\ell})$ → exponentiel (très compliqué).

Exemples :

▷ Calculer $\mathrm{pgcd}(\ell_1, \ell_2)$ → linéaire.

▷ Décider si un entier ℓ est premier

▷ Factoriser un entier ℓ

▷ Calculer g^ℓ en faisant $g \times g \cdots \times g$

⇝ La multiplication est (probablement) une fonction à sens unique !
(→ RSA).

Facile ? Pas facile ?

Entrée d'un programme est ℓ : complexité $=$ le nbr. de multiplications.

▷ Calcul en $O(\log(\ell)) = O(e^{\log\log\ell})$ → linéaire (très facile).

▷ Calcul en $O(\log(\ell)^k) = O(e^{k\log\log\ell})$ → polynomial (facile).

▷ Calcul en $O(e^{k\log(\ell)^a(\log\log\ell)^{1-a}})$ → sous-exponentiel (compliqué).

▷ Calcul en $O(\ell) = O(e^{\log\ell})$ → exponentiel (très compliqué).

Exemples :

▷ Calculer $\mathrm{pgcd}(\ell_1, \ell_2)$ → linéaire.

▷ Décider si un entier ℓ est premier → polynomial.

▷ Factoriser un entier ℓ

▷ Calculer g^ℓ en faisant $g \times g \cdots \times g$

⇝ La multiplication est (probablement) une fonction à sens unique !
(→ RSA).

Facile ? Pas facile ?

Entrée d'un programme est ℓ : complexité $=$ le nbr. de multiplications.

▷ Calcul en $O(\log(\ell)) = O(e^{\log\log\ell})$ → linéaire (très facile).

▷ Calcul en $O(\log(\ell)^k) = O(e^{k\log\log\ell})$ → polynomial (facile).

▷ Calcul en $O(e^{k\log(\ell)^a(\log\log\ell)^{1-a}})$ → sous-exponentiel (compliqué).

▷ Calcul en $O(\ell) = O(e^{\log\ell})$ → exponentiel (très compliqué).

Exemples :

▷ Calculer $\mathrm{pgcd}(\ell_1, \ell_2)$ → linéaire.

▷ Décider si un entier ℓ est premier → polynomial.

▷ Factoriser un entier ℓ

▷ Calculer g^ℓ en faisant $g \times g \cdots \times g$

⇝ La multiplication est (probablement) une fonction à sens unique !
(→ RSA).

Facile ? Pas facile ?

Entrée d'un programme est ℓ : complexité $=$ le nbr. de multiplications.

▷ Calcul en $O(\log(\ell)) = O(e^{\log\log\ell})$ → linéaire (très facile).

▷ Calcul en $O(\log(\ell)^k) = O(e^{k\log\log\ell})$ → polynomial (facile).

▷ Calcul en $O(e^{k\log(\ell)^a(\log\log\ell)^{1-a}})$ → sous-exponentiel (compliqué).

▷ Calcul en $O(\ell) = O(e^{\log\ell})$ → exponentiel (très compliqué).

Exemples :

▷ Calculer $\mathrm{pgcd}(\ell_1, \ell_2)$ → linéaire.

▷ Décider si un entier ℓ est premier → polynomial.

▷ Factoriser un entier ℓ → sous-exponentiel ($a = 1/3$).

▷ Calculer g^ℓ en faisant $g \times g \cdots \times g$

⇝ La multiplication est (probablement) une fonction à sens unique !
(→ RSA).

Facile ? Pas facile ?

Entrée d'un programme est ℓ : complexité $=$ le nbr. de multiplications.

▷ Calcul en $O(\log(\ell)) = O(e^{\log\log\ell})$ → linéaire (très facile).

▷ Calcul en $O(\log(\ell)^k) = O(e^{k\log\log\ell})$ → polynomial (facile).

▷ Calcul en $O(e^{k\log(\ell)^a(\log\log\ell)^{1-a}})$ → sous-exponentiel (compliqué).

▷ Calcul en $O(\ell) = O(e^{\log\ell})$ → exponentiel (très compliqué).

Exemples :

▷ Calculer $\mathrm{pgcd}(\ell_1, \ell_2)$ → linéaire.

▷ Décider si un entier ℓ est premier → polynomial.

▷ Factoriser un entier ℓ → sous-exponentiel ($a = 1/3$).

▷ Calculer g^ℓ en faisant $g \times g \cdots \times g$

↝ La multiplication est (probablement) une fonction à sens unique !
($\to$ RSA).

Facile ? Pas facile ?

Entrée d'un programme est ℓ : complexité $=$ le nbr. de multiplications.

▷ Calcul en $O(\log(\ell)) = O(e^{\log\log\ell})$ → linéaire (très facile).

▷ Calcul en $O(\log(\ell)^k) = O(e^{k\log\log\ell})$ → polynomial (facile).

▷ Calcul en $O(e^{k\log(\ell)^a(\log\log\ell)^{1-a}})$ → sous-exponentiel (compliqué).

▷ Calcul en $O(\ell) = O(e^{\log\ell})$ → exponentiel (très compliqué).

Exemples :

▷ Calculer $\mathrm{pgcd}(\ell_1, \ell_2)$ → linéaire.

▷ Décider si un entier ℓ est premier → polynomial.

▷ Factoriser un entier ℓ → sous-exponentiel ($a = 1/3$).

▷ Calculer g^ℓ en faisant $g \times g \cdots \times g$ → exponentiel.

⇝ La multiplication est (probablement) une fonction à sens unique !
(→ RSA).

Facile ? Pas facile ?

Entrée d'un programme est ℓ : complexité $=$ le nbr. de multiplications.

▷ Calcul en $O(\log(\ell)) = O(e^{\log\log\ell})$ → linéaire (très facile).

▷ Calcul en $O(\log(\ell)^k) = O(e^{k\log\log\ell})$ → polynomial (facile).

▷ Calcul en $O(e^{k\log(\ell)^a(\log\log\ell)^{1-a}})$ → sous-exponentiel (compliqué).

▷ Calcul en $O(\ell) = O(e^{\log\ell})$ → exponentiel (très compliqué).

Exemples :

▷ Calculer $\mathrm{pgcd}(\ell_1, \ell_2)$ → linéaire.

▷ Décider si un entier ℓ est premier → polynomial.

▷ Factoriser un entier ℓ → sous-exponentiel ($a = 1/3$).

▷ Calculer g^ℓ en faisant $g \times g \cdots \times g$ → exponentiel.

⇝ La multiplication est (probablement) une fonction à sens unique !
(→ RSA).

Facile ? Pas facile ?

Entrée d'un programme est ℓ : complexité $=$ le nbr. de multiplications.

▷ Calcul en $O(\log(\ell)) = O(e^{\log\log\ell})$ → linéaire (très facile).

▷ Calcul en $O(\log(\ell)^k) = O(e^{k\log\log\ell})$ → polynomial (facile).

▷ Calcul en $O(e^{k\log(\ell)^a(\log\log\ell)^{1-a}})$ → sous-exponentiel (compliqué).

▷ Calcul en $O(\ell) = O(e^{\log\ell})$ → exponentiel (très compliqué).

Exemples :

▷ Calculer $\mathrm{pgcd}(\ell_1, \ell_2)$ → linéaire.

▷ Décider si un entier ℓ est premier → polynomial.

▷ Factoriser un entier ℓ → sous-exponentiel ($a = 1/3$).

▷ Calculer g^ℓ en faisant $g \times g \cdots \times g$ → exponentiel.

⇝ La multiplication est (probablement) une fonction à sens unique ! (→ RSA).

Le problème du logarithme discret

Soit G un groupe cyclique engendré par g d'ordre n :

$$G = \{1, g, g^2, \cdots, g^{n-1}\}$$

Les opérations se font rapidement. (typiquement $G = (\mathbb{Z}/p\mathbb{Z})^\times$.)

$\rightsquigarrow$ Connaissant g et $\ell \in \mathbb{Z}$, il est facile de calculer $y = g^\ell \in G$.

En revanche, la réciproque...

Pb du log discret en base g

Soit $y \in G$, trouver $\ell \in \mathbb{Z}$ tel que $g^\ell = y$ est le problème du logarithme discret en base g. On note :

$$\ell = \log_g y$$

$\rightsquigarrow$ En principe, ce problème est difficile.

$\rightsquigarrow$ En principe, la fonction puissance dans G est une fonction à sens unique.

Le problème du logarithme discret

Soit G un groupe cyclique engendré par g d'ordre n :

$$G = \{1, g, g^2, \cdots, g^{n-1}\}$$

Les opérations se font rapidement. (typiquement $G = (\mathbb{Z}/p\mathbb{Z})^\times$.)

$\rightsquigarrow$ Connaissant g et $\ell \in \mathbb{Z}$, il est facile de calculer $y = g^\ell \in G$.

En revanche, la réciproque...

Pb du log discret en base g

Soit $y \in G$, trouver $\ell \in \mathbb{Z}$ tel que $g^\ell = y$ est le problème du logarithme discret en base g. On note :

$$\ell = \log_g y$$

$\rightsquigarrow$ En principe, ce problème est difficile.

$\rightsquigarrow$ En principe, la fonction puissance dans G est une fonction à sens unique.

Le problème du logarithme discret

Soit G un groupe cyclique engendré par g d'ordre n :

$$G = \{1, g, g^2, \cdots, g^{n-1}\}$$

Les opérations se font rapidement. (typiquement $G = (\mathbb{Z}/p\mathbb{Z})^\times$.)

$\rightsquigarrow$ Connaissant g et $\ell \in \mathbb{Z}$, il est facile de calculer $y = g^\ell \in G$.

En revanche, la réciproque...

Pb du log discret en base g

Soit $y \in G$, trouver $\ell \in \mathbb{Z}$ tel que $g^\ell = y$ est le problème du logarithme discret en base g. On note :

$$\ell = \log_g y$$

$\rightsquigarrow$ En principe, ce problème est difficile.

$\rightsquigarrow$ En principe, la fonction puissance dans G est une fonction à sens unique.

Le problème du logarithme discret

Soit G un groupe cyclique engendré par g d'ordre n :

$$G = \{1, g, g^2, \cdots, g^{n-1}\}$$

Les opérations se font rapidement. (typiquement $G = (\mathbb{Z}/p\mathbb{Z})^\times$.)

$\rightsquigarrow$ Connaissant g et $\ell \in \mathbb{Z}$, il est facile de calculer $y = g^\ell \in G$.

En revanche, la réciproque...

Pb du log discret en base g

Soit $y \in G$, trouver $\ell \in \mathbb{Z}$ tel que $g^\ell = y$ est le problème du logarithme discret en base g. On note :

$$\ell = \log_g y$$

$\rightsquigarrow$ En principe, ce problème est difficile.

$\rightsquigarrow$ En principe, la fonction puissance dans G est une fonction à sens unique.

Le problème du logarithme discret

Soit G un groupe cyclique engendré par g d'ordre n :

$$G = \{1, g, g^2, \cdots, g^{n-1}\}$$

Les opérations se font rapidement. (typiquement $G = (\mathbb{Z}/p\mathbb{Z})^\times$.)

$\rightsquigarrow$ Connaissant g et $\ell \in \mathbb{Z}$, il est facile de calculer $y = g^\ell \in G$.

En revanche, la réciproque...

Pb du log discret en base g

Soit $y \in G$, trouver $\ell \in \mathbb{Z}$ tel que $g^\ell = y$ est le problème du logarithme discret en base g. On note :

$$\ell = \log_g y$$

$\rightsquigarrow$ En principe, ce problème est difficile.

$\rightsquigarrow$ En principe, la fonction puissance dans G est une fonction à sens unique.

Protocole Diffie-Hellman

Alice et Bob se mettent d'accord sur un groupe G et $g \in G$ d'ordre n.

$\rightsquigarrow$ Alice choisit k_A secrètement, calcule $y_A = g^{k_A}$ et l'envoie à Bob.

$\rightsquigarrow$ Bob choisit k_B secrètement, calcule $y_B = g^{k_B}$ et l'envoie à Alice.

$\rightsquigarrow$ Alice reçoit y_B et calcule $(y_B)^{k_A} = g^{k_B k_A}$.

$\rightsquigarrow$ Bob reçoit y_A et calcule $(y_A)^{k_B} = g^{k_A k_B}$.

$\rightsquigarrow$ Alice et Bob partagent le secret commun $g^{k_A k_B}$.

$\triangleright$ Un espion, Eve ou Charlie, connaît : G, g, g^{k_A} et g^{k_B}.
Il doit calculer $g^{k_A k_B}$.

Heuristique

Résoudre le problème de Diffie-Hellman i.e. trouver $g^{k_A k_B}$ en connaissant g, g^{k_A} et g^{k_B} est aussi difficile que de résoudre le problème du logarithme discret.

$\rightsquigarrow$ Il y a des pièges à éviter (ex. l'attaque de l'homme du milieu).

$\rightsquigarrow$ Le protocole Diffie-Hellman est à la base de nombreux autres.

Protocole Diffie-Hellman

Alice et Bob se mettent d'accord sur un groupe G et $g \in G$ d'ordre n.

$\rightsquigarrow$ Alice choisit k_A secrètement, calcule $y_A = g^{k_A}$ et l'envoie à Bob.

$\rightsquigarrow$ Bob choisit k_B secrètement, calcule $y_B = g^{k_B}$ et l'envoie à Alice.

$\rightsquigarrow$ Alice reçoit y_B et calcule $(y_B)^{k_A} = g^{k_B k_A}$.

$\rightsquigarrow$ Bob reçoit y_A et calcule $(y_A)^{k_B} = g^{k_A k_B}$.

$\rightsquigarrow$ Alice et Bob partagent le secret commun $g^{k_A k_B}$.

$\triangleright$ Un espion, Eve ou Charlie, connaît : G, g, g^{k_A} et g^{k_B}.
Il doit calculer $g^{k_A k_B}$.

Heuristique

Résoudre le problème de Diffie-Hellman i.e. trouver $g^{k_A k_B}$ en connaissant g, g^{k_A} et g^{k_B} est aussi difficile que de résoudre le problème du logarithme discret.

$\rightsquigarrow$ Il y a des pièges à éviter (ex. l'attaque de l'homme du milieu).

$\rightsquigarrow$ Le protocole Diffie-Hellman est à la base de nombreux autres.

Protocole Diffie-Hellman

Alice et Bob se mettent d'accord sur un groupe G et $g \in G$ d'ordre n.

$\rightsquigarrow$ Alice choisit k_A secrètement, calcule $y_A = g^{k_A}$ et l'envoie à Bob.

$\rightsquigarrow$ Bob choisit k_B secrètement, calcule $y_B = g^{k_B}$ et l'envoie à Alice.

$\rightsquigarrow$ Alice reçoit y_B et calcule $(y_B)^{k_A} = g^{k_B k_A}$.

$\rightsquigarrow$ Bob reçoit y_A et calcule $(y_A)^{k_B} = g^{k_A k_B}$.

$\rightsquigarrow$ Alice et Bob partagent le secret commun $g^{k_A k_B}$.

$\triangleright$ Un espion, Eve ou Charlie, connaît : G, g, g^{k_A} et g^{k_B}.
Il doit calculer $g^{k_A k_B}$.

Heuristique

Résoudre le problème de Diffie-Hellman i.e. trouver $g^{k_A k_B}$ en connaissant g, g^{k_A} et g^{k_B} est aussi difficile que de résoudre le problème du logarithme discret.

$\rightsquigarrow$ Il y a des pièges à éviter (ex. l'attaque de l'homme du milieu).

$\rightsquigarrow$ Le protocole Diffie-Hellman est à la base de nombreux autres.

Protocole Diffie-Hellman

Alice et Bob se mettent d'accord sur un groupe G et $g \in G$ d'ordre n.

$\rightsquigarrow$ Alice choisit k_A secrètement, calcule $y_A = g^{k_A}$ et l'envoie à Bob.

$\rightsquigarrow$ Bob choisit k_B secrètement, calcule $y_B = g^{k_B}$ et l'envoie à Alice.

$\rightsquigarrow$ Alice reçoit y_B et calcule $(y_B)^{k_A} = g^{k_B k_A}$.

$\rightsquigarrow$ Bob reçoit y_A et calcule $(y_A)^{k_B} = g^{k_A k_B}$.

$\rightsquigarrow$ Alice et Bob partagent le secret commun $g^{k_A k_B}$.

$\triangleright$ Un espion, Eve ou Charlie, connaît : G, g, g^{k_A} et g^{k_B}.
Il doit calculer $g^{k_A k_B}$.

Heuristique

Résoudre le problème de Diffie-Hellman i.e. trouver $g^{k_A k_B}$ en connaissant g, g^{k_A} et g^{k_B} est aussi difficile que de résoudre le problème du logarithme discret.

$\rightsquigarrow$ Il y a des pièges à éviter (ex. l'attaque de l'homme du milieu).

$\rightsquigarrow$ Le protocole Diffie-Hellman est à la base de nombreux autres.

Protocole Diffie-Hellman

Alice et Bob se mettent d'accord sur un groupe G et $g \in G$ d'ordre n.

$\rightsquigarrow$ Alice choisit k_A secrètement, calcule $y_A = g^{k_A}$ et l'envoie à Bob.

$\rightsquigarrow$ Bob choisit k_B secrètement, calcule $y_B = g^{k_B}$ et l'envoie à Alice.

$\rightsquigarrow$ Alice reçoit y_B et calcule $(y_B)^{k_A} = g^{k_B k_A}$.

$\rightsquigarrow$ Bob reçoit y_A et calcule $(y_A)^{k_B} = g^{k_A k_B}$.

$\rightsquigarrow$ Alice et Bob partagent le secret commun $g^{k_A k_B}$.

$\triangleright$ Un espion, Eve ou Charlie, connaît : G, g, g^{k_A} et g^{k_B}.
Il doit calculer $g^{k_A k_B}$.

Heuristique

Résoudre le problème de Diffie-Hellman i.e. trouver $g^{k_A k_B}$ en connaissant g, g^{k_A} et g^{k_B} est aussi difficile que de résoudre le problème du logarithme discret.

$\rightsquigarrow$ Il y a des pièges à éviter (ex. l'attaque de l'homme du milieu).

$\rightsquigarrow$ Le protocole Diffie-Hellman est à la base de nombreux autres.

Protocole Diffie-Hellman

Alice et Bob se mettent d'accord sur un groupe G et $g \in G$ d'ordre n.

$\rightsquigarrow$ Alice choisit k_A secrètement, calcule $y_A = g^{k_A}$ et l'envoie à Bob.

$\rightsquigarrow$ Bob choisit k_B secrètement, calcule $y_B = g^{k_B}$ et l'envoie à Alice.

$\rightsquigarrow$ Alice reçoit y_B et calcule $(y_B)^{k_A} = g^{k_B k_A}$.

$\rightsquigarrow$ Bob reçoit y_A et calcule $(y_A)^{k_B} = g^{k_A k_B}$.

$\rightsquigarrow$ Alice et Bob partagent le secret commun $g^{k_A k_B}$.

$\triangleright$ Un espion, Eve ou Charlie, connaît : G, g, g^{k_A} et g^{k_B}.
Il doit calculer $g^{k_A k_B}$.

Heuristique

Résoudre le problème de Diffie-Hellman i.e. trouver $g^{k_A k_B}$ en connaissant g, g^{k_A} et g^{k_B} est aussi difficile que de résoudre le problème du logarithme discret.

$\rightsquigarrow$ Il y a des pièges à éviter (ex. l'attaque de l'homme du milieu).

$\rightsquigarrow$ Le protocole Diffie-Hellman est à la base de nombreux autres.

Protocole Diffie-Hellman

Alice et Bob se mettent d'accord sur un groupe G et $g \in G$ d'ordre n.

$\rightsquigarrow$ Alice choisit k_A secrètement, calcule $y_A = g^{k_A}$ et l'envoie à Bob.

$\rightsquigarrow$ Bob choisit k_B secrètement, calcule $y_B = g^{k_B}$ et l'envoie à Alice.

$\rightsquigarrow$ Alice reçoit y_B et calcule $(y_B)^{k_A} = g^{k_B k_A}$.

$\rightsquigarrow$ Bob reçoit y_A et calcule $(y_A)^{k_B} = g^{k_A k_B}$.

$\rightsquigarrow$ Alice et Bob partagent le secret commun $g^{k_A k_B}$.

$\triangleright$ Un espion, Eve ou Charlie, connaît : G, g, g^{k_A} et g^{k_B}.
Il doit calculer $g^{k_A k_B}$.

Heuristique

Résoudre le problème de Diffie-Hellman i.e. trouver $g^{k_A k_B}$ en connaissant g, g^{k_A} et g^{k_B} est aussi difficile que de résoudre le problème du logarithme discret.

$\rightsquigarrow$ Il y a des pièges à éviter (ex. l'attaque de l'homme du milieu).

$\rightsquigarrow$ Le protocole Diffie-Hellman est à la base de nombreux autres.

Protocole Diffie-Hellman

Alice et Bob se mettent d'accord sur un groupe G et $g \in G$ d'ordre n.

$\rightsquigarrow$ Alice choisit k_A secrètement, calcule $y_A = g^{k_A}$ et l'envoie à Bob.

$\rightsquigarrow$ Bob choisit k_B secrètement, calcule $y_B = g^{k_B}$ et l'envoie à Alice.

$\rightsquigarrow$ Alice reçoit y_B et calcule $(y_B)^{k_A} = g^{k_B k_A}$.

$\rightsquigarrow$ Bob reçoit y_A et calcule $(y_A)^{k_B} = g^{k_A k_B}$.

$\rightsquigarrow$ Alice et Bob partagent le secret commun $g^{k_A k_B}$.

$\triangleright$ Un espion, Eve ou Charlie, connaît : G, g, g^{k_A} et g^{k_B}.
Il doit calculer $g^{k_A k_B}$.

Heuristique

Résoudre le problème de Diffie-Hellman i.e. trouver $g^{k_A k_B}$ en connaissant g, g^{k_A} et g^{k_B} est aussi difficile que de résoudre le problème du logarithme discret.

$\rightsquigarrow$ Il y a des pièges à éviter (ex. l'attaque de l'homme du milieu).

$\rightsquigarrow$ Le protocole Diffie-Hellman est à la base de nombreux autres.

Protocole Diffie-Hellman

Alice et Bob se mettent d'accord sur un groupe G et $g \in G$ d'ordre n.

$\rightsquigarrow$ Alice choisit k_A secrètement, calcule $y_A = g^{k_A}$ et l'envoie à Bob.

$\rightsquigarrow$ Bob choisit k_B secrètement, calcule $y_B = g^{k_B}$ et l'envoie à Alice.

$\rightsquigarrow$ Alice reçoit y_B et calcule $(y_B)^{k_A} = g^{k_B k_A}$.

$\rightsquigarrow$ Bob reçoit y_A et calcule $(y_A)^{k_B} = g^{k_A k_B}$.

$\rightsquigarrow$ Alice et Bob partagent le secret commun $g^{k_A k_B}$.

$\triangleright$ Un espion, Eve ou Charlie, connaît : G, g, g^{k_A} et g^{k_B}.
Il doit calculer $g^{k_A k_B}$.

Heuristique

Résoudre le problème de Diffie-Hellman i.e. trouver $g^{k_A k_B}$ en connaissant g, g^{k_A} et g^{k_B} est aussi difficile que de résoudre le problème du logarithme discret.

$\rightsquigarrow$ Il y a des pièges à éviter (ex. l'attaque de l'homme du milieu).

$\rightsquigarrow$ Le protocole Diffie-Hellman est à la base de nombreux autres.

Protocole Diffie-Hellman

Alice et Bob se mettent d'accord sur un groupe G et $g \in G$ d'ordre n.

$\rightsquigarrow$ Alice choisit k_A secrètement, calcule $y_A = g^{k_A}$ et l'envoie à Bob.

$\rightsquigarrow$ Bob choisit k_B secrètement, calcule $y_B = g^{k_B}$ et l'envoie à Alice.

$\rightsquigarrow$ Alice reçoit y_B et calcule $(y_B)^{k_A} = g^{k_B k_A}$.

$\rightsquigarrow$ Bob reçoit y_A et calcule $(y_A)^{k_B} = g^{k_A k_B}$.

$\rightsquigarrow$ Alice et Bob partagent le secret commun $g^{k_A k_B}$.

$\triangleright$ Un espion, Eve ou Charlie, connaît : G, g, g^{k_A} et g^{k_B}.
Il doit calculer $g^{k_A k_B}$.

Heuristique

Résoudre le problème de Diffie-Hellman i.e. trouver $g^{k_A k_B}$ en connaissant g, g^{k_A} et g^{k_B} est aussi difficile que de résoudre le problème du logarithme discret.

$\rightsquigarrow$ Il y a des pièges à éviter (ex. l'attaque de l'homme du milieu).

$\rightsquigarrow$ Le protocole Diffie-Hellman est à la base de nombreux autres.

Résoudre le problème du log discret : $y = g^\ell$

▷ Réduction de Pohlig-Hellman :

$\rightsquigarrow$ Si G est d'ordre $n = pq$ avec p et q premiers entre eux.

On écrit $pu + qv = 1$. On résout

$$y^{pu} = \left(g^p\right)^{\ell_1} \quad \leftarrow \text{Pb dans un groupe d'ordre } q.$$
$$y^{qv} = \left(g^q\right)^{\ell_2} \quad \leftarrow \text{Pb dans un groupe d'ordre } p.$$

On a $y = y^{pu+qv} = g^{p\ell_1 + q\ell_2}$!

$\rightsquigarrow$ Si G est d'ordre p^m avec p premier.

Il existe une astuce similaire qui permet de résoudre le pb du log discret dans G en résolvant m pbs du log discret dans un groupe d'ordre p.

Fait

Il faut absolument prendre des groupes d'ordre premier.

Résoudre le problème du log discret : $y = g^\ell$

▷ Réduction de Pohlig-Hellman :

$\rightsquigarrow$ Si G est d'ordre $n = pq$ avec p et q premiers entre eux.

On écrit $pu + qv = 1$. On résout

$$y^{pu} \qquad \left(g^p\right)^{\ell_1} \qquad \leftarrow \text{Pb dans un groupe d'ordre } q.$$

$$y^{qv} \qquad \left(g^q\right)^{\ell_2} \qquad \leftarrow \text{Pb dans un groupe d'ordre } p.$$

On a $y = y^{pu+qv} = g^{p\ell_1 + q\ell_2}$!

$\rightsquigarrow$ Si G est d'ordre p^m avec p premier.

Il existe une astuce similaire qui permet de résoudre le pb du log discret dans G en résolvant m pbs du log discret dans un groupe d'ordre p.

Fail

Il faut absolument prendre des groupes d'ordre premier.

Résoudre le problème du log discret : $y = g^\ell$

▷ Réduction de Pohlig-Hellman :

⇝ Si G est d'ordre $n = pq$ avec p et q premiers entre eux.

On écrit $pu + qv = 1$. On résout

$$y^{pu} = (g^p)^{\ell_1} \quad \leftarrow \text{Pb dans un groupe d'ordre } q.$$

$$y^{qv} = (g^q)^{\ell_2} \quad \leftarrow \text{Pb dans un groupe d'ordre } p.$$

On a $y = y^{pu+qv} = g^{p\ell_1 + q\ell_2}$!

⇝ Si G est d'ordre p^m avec p premier.

Il existe une astuce similaire qui permet de résoudre le pb du log discret dans G en résolvant m pbs du log discret dans un groupe d'ordre p.

Fail

Il faut absolument prendre des groupes d'ordre premier.

Résoudre le problème du log discret : $y = g^\ell$

▷ Réduction de Pohlig-Hellman :

$\rightsquigarrow$ Si G est d'ordre $n = pq$ avec p et q premiers entre eux.

On écrit $pu + qv = 1$. On résout

$$y^{pu} = (g^p)^{\ell_1} \quad \leftarrow \text{Pb dans un groupe d'ordre } q.$$

$$y^{qv} = (g^q)^{\ell_2} \quad \leftarrow \text{Pb dans un groupe d'ordre } p.$$

On a $y = y^{pu+qv} = g^{p\ell_1+q\ell_2}$!

$\rightsquigarrow$ Si G est d'ordre p^m avec p premier.

Il existe une astuce similaire qui permet de résoudre le pb du log discret dans G en résolvant m pbs du log discret dans un groupe d'ordre p.

Fail

Il faut absolument prendre des groupes d'ordre premier.

Résoudre le problème du log discret : $y = g^\ell$

▷ Réduction de Pohlig-Hellman :

$\rightsquigarrow$ Si G est d'ordre $n = pq$ avec p et q premiers entre eux.

On écrit $pu + qv = 1$. On résout

$$\begin{aligned} y^{pu} &= (g^p)^{\ell_1} && \leftarrow \text{Pb dans un groupe d'ordre } q. \\ y^{qv} &= (g^q)^{\ell_2} && \leftarrow \text{Pb dans un groupe d'ordre } p. \end{aligned}$$

On a $y = y^{pu+qv} = g^{p\ell_1+q\ell_2}$!

$\rightsquigarrow$ Si G est d'ordre p^m avec p premier.

Il existe une astuce similaire qui permet de résoudre le pb du log discret dans G en résolvant m pbs du log discret dans un groupe d'ordre p.

Fail

Il faut absolument prendre des groupes d'ordre premier.

Résoudre le problème du log discret : $y = g^\ell$

▷ Réduction de Pohlig-Hellman :

$\rightsquigarrow$ Si G est d'ordre $n = pq$ avec p et q premiers entre eux.

On écrit $pu + qv = 1$. On résout

$$\begin{aligned} y^{pu} &= (g^p)^{\ell_1} && \leftarrow \text{Pb dans un groupe d'ordre } q. \\ y^{qv} &= (g^q)^{\ell_2} && \leftarrow \text{Pb dans un groupe d'ordre } p. \end{aligned}$$

On a $y = y^{pu+qv} = g^{p\ell_1+q\ell_2}$!

$\rightsquigarrow$ Si G est d'ordre p^m avec p premier.

Il existe une astuce similaire qui permet de résoudre le pb du log discret dans G en résolvant m pbs du log discret dans un groupe d'ordre p.

Fail

Il faut absolument prendre des groupes d'ordre premier.

Résoudre le problème du log discret : $y = g^\ell$

▷ Réduction de Pohlig-Hellman :

$\rightsquigarrow$ Si G est d'ordre $n = pq$ avec p et q premiers entre eux.

On écrit $pu + qv = 1$. On résout

$$\begin{aligned} y^{pu} &= (g^p)^{\ell_1} && \leftarrow \text{Pb dans un groupe d'ordre } q. \\ y^{qv} &= (g^q)^{\ell_2} && \leftarrow \text{Pb dans un groupe d'ordre } p. \end{aligned}$$

On a $y = y^{pu+qv} = g^{p\ell_1+q\ell_2}$!

$\rightsquigarrow$ Si G est d'ordre p^m avec p premier.

Il existe une astuce similaire qui permet de résoudre le pb du log discret dans G en résolvant m pbs du log discret dans un groupe d'ordre p.

Fail

Il faut absolument prendre des groupes d'ordre premier.

Résoudre le problème du log discret : $y = g^\ell$

▷ Réduction de Pohlig-Hellman :

$\rightsquigarrow$ Si G est d'ordre $n = pq$ avec p et q premiers entre eux.

On écrit $pu + qv = 1$. On résout

$$\begin{aligned} y^{pu} &= (g^p)^{\ell_1} && \leftarrow \text{Pb dans un groupe d'ordre } q. \\ y^{qv} &= (g^q)^{\ell_2} && \leftarrow \text{Pb dans un groupe d'ordre } p. \end{aligned}$$

On a $y = y^{pu+qv} = g^{p\ell_1+q\ell_2}$!

$\rightsquigarrow$ Si G est d'ordre p^m avec p premier.

Il existe une astuce similaire qui permet de résoudre le pb du log discret dans G en résolvant m pbs du log discret dans un groupe d'ordre p.

Fait

Il faut absolument prendre des groupes d'ordre premier.

Résoudre le problème du log discret : $y = g^\ell$

▷ Réduction de Pohlig-Hellman :

$\rightsquigarrow$ Si G est d'ordre $n = pq$ avec p et q premiers entre eux.

On écrit $pu + qv = 1$. On résout

$$\begin{aligned} y^{pu} &= (g^p)^{\ell_1} && \leftarrow \text{Pb dans un groupe d'ordre } q. \\ y^{qv} &= (g^q)^{\ell_2} && \leftarrow \text{Pb dans un groupe d'ordre } p. \end{aligned}$$

On a $y = y^{pu+qv} = g^{p\ell_1+q\ell_2}$!

$\rightsquigarrow$ Si G est d'ordre p^m avec p premier.

Il existe une astuce similaire qui permet de résoudre le pb du log discret dans G en résolvant m pbs du log discret dans un groupe d'ordre p.

Fait

Il faut absolument prendre des groupes d'ordre premier.

Résoudre le problème du log discret

▷ Algorithme Baby-Steps / Giant Steps :

$\rightsquigarrow$ Si G est d'ordre n. On pose $m = \lceil \sqrt{n} \rceil$. Dans $y = g^\ell$, on écrit $\ell = km + r$ avec $0 \leq k, r < m$. On a $yg^{-r} = (g^m)^k$.

▷ On calcule les pas de géant. $\leftarrow m$ étapes.

▷ On calcule les pas de bébé jusqu'à une collision. $\leftarrow \leq m$ étapes.

On a alors $yg^{-r} = (g^m)^k$ donc $y = g^{km+r}$.

Résoudre le problème du log discret

▷ Algorithme Baby-Steps / Giant Steps :

⇝ Si G est d'ordre n. On pose $m = \lceil \sqrt{n} \rceil$. Dans $y = g^\ell$, on écrit $\ell = km + r$ avec $0 \leq k, r < m$. On a $yg^{-r} = (g^m)^k$.

▷ On calcule les pas de géant. $\leftarrow m$ étapes.

▷ On calcule les pas de bébé jusqu'à une collision. $\leftarrow \leq m$ étapes.

On a alors $yg^{-r} = (g^m)^k$ donc $y = g^{km+r}$.

Résoudre le problème du log discret

▷ Algorithme Baby-Steps / Giant Steps :

⇝ Si G est d'ordre n. On pose $m = \lceil \sqrt{n} \rceil$. Dans $y = g^\ell$, on écrit $\ell = km + r$ avec $0 \leq k, r < m$. On a $yg^{-r} = (g^m)^k$.

▷ On calcule les pas de géant. $\leftarrow m$ étapes.

▷ On calcule les pas de bébé jusqu'à une collision. $\leftarrow \leq m$ étapes.

On a alors $yg^{-r} = (g^m)^k$ donc $y = g^{km+r}$.

Résoudre le problème du log discret

▷ Algorithme Baby-Steps / Giant Steps :

$\rightsquigarrow$ Si G est d'ordre n. On pose $m = \lceil \sqrt{n} \rceil$. Dans $y = g^\ell$, on écrit $\ell = km + r$ avec $0 \leq k, r < m$. On a $yg^{-r} = (g^m)^k$.

▷ On calcule les pas de géant. $\leftarrow m$ étapes.

▷ On calcule les pas de bébé jusqu'à une collision. $\leftarrow \leq m$ étapes.

On a alors $yg^{-r} = (g^m)^k$ donc $y = g^{km+r}$.

Résoudre le problème du log discret

▷ Algorithme Baby-Steps / Giant Steps :

⇝ Si G est d'ordre n. On pose $m = \lceil \sqrt{n} \rceil$. Dans $y = g^\ell$, on écrit $\ell = km + r$ avec $0 \leq k, r < m$. On a $yg^{-r} = (g^m)^k$.

Pas de géant	Pas de bébé
$(g^m)^0$	yg^{-0}
$(g^m)^1$	yg^{-1}
⋮	⋮
$(g^m)^k$	⋮
⋮	yg^{-r}
$(g^m)^{m-1}$	

▷ On calcule les pas de géant. ← m étapes.

▷ On calcule les pas de bébé jusqu'à une collision. ← $\leq m$ étapes.

On a alors $yg^{-r} = (g^m)^k$ donc $y = g^{km+r}$.

Résoudre le problème du log discret

▷ Algorithme Baby-Steps / Giant Steps :

$\rightsquigarrow$ Si G est d'ordre n. On pose $m = \lceil \sqrt{n} \rceil$. Dans $y = g^\ell$, on écrit $\ell = km + r$ avec $0 \leq k, r < m$. On a $yg^{-r} = (g^m)^k$.

Pas de géant	Pas de bébé

▷ On calcule les pas de géant.

▷ On calcule les pas de bébé jusqu'à une collision.

On a alors $yg^{-r} = (g^m)^k$ donc $y = g^{km+r}$.

Résoudre le problème du log discret

▷ Algorithme Baby-Steps / Giant Steps :

$\rightsquigarrow$ Si G est d'ordre n. On pose $m = \lceil \sqrt{n} \rceil$. Dans $y = g^\ell$, on écrit $\ell = km + r$ avec $0 \leq k, r < m$. On a $yg^{-r} = (g^m)^k$.

Pas de géant	Pas de bébé
$(g^m)^0$	yg^{-0}
$(g^m)^1$	yg^{-1}
$\vdots$	$\vdots$
$(g^m)^k$	$\vdots$
$\vdots$	yg^{-r}
$(g^m)^{m-1}$	

▷ On calcule les pas de géant. $\leftarrow m$ étapes.

▷ On calcule les pas de bébé jusqu'à une collision. $\leftarrow \leq m$ étapes.

On a alors $yg^{-r} = (g^m)^k$ donc $y = g^{km+r}$.

Résoudre le problème du log discret

▷ Algorithme Baby-Steps / Giant Steps :

$\rightsquigarrow$ Si G est d'ordre n. On pose $m = \lceil \sqrt{n} \rceil$. Dans $y = g^\ell$, on écrit $\ell = km + r$ avec $0 \leq k, r < m$. On a $yg^{-r} = (g^m)^k$.

Pas de géant	Pas de bébé
$(g^m)^0$	yg^{-0}
$(g^m)^1$	yg^{-1}
$\vdots$	$\vdots$
$(g^m)^k$	$\vdots$
$\vdots$	yg^{-r}
$(g^m)^{m-1}$	

▷ On calcule les pas de géant. $\leftarrow m$ étapes.

▷ On calcule les pas de bébé jusqu'à une collision. $\leftarrow \leq m$ étapes.

On a alors $yg^{-r} = (g^m)^k$ donc $y = g^{km+r}$.

Résoudre le problème du log discret

▷ Algorithme Baby-Steps / Giant Steps :

$\rightsquigarrow$ Si G est d'ordre n. On pose $m = \lceil \sqrt{n} \rceil$. Dans $y = g^\ell$, on écrit $\ell = km + r$ avec $0 \leq k, r < m$. On a $yg^{-r} = (g^m)^k$.

Pas de géant	Pas de bébé
$(g^m)^0$	yg^{-0}
$(g^m)^1$	yg^{-1}
$\vdots$	$\vdots$
$(g^m)^k$	$\vdots$
$\vdots$	yg^{-r}
$(g^m)^{m-1}$	

▷ On calcule les pas de géant. $\leftarrow m$ étapes.

▷ On calcule les pas de bébé jusqu'à une collision. $\leftarrow \leq m$ étapes.

On a alors $yg^{-r} = (g^m)^k$ donc $y = g^{km+r}$.

Résoudre le problème du log discret

▷ Algorithme Baby-Steps / Giant Steps :

$\rightsquigarrow$ Si G est d'ordre n. On pose $m = \lceil \sqrt{n} \rceil$. Dans $y = g^\ell$, on écrit $\ell = km + r$ avec $0 \leq k, r < m$. On a $yg^{-r} = (g^m)^k$.

Pas de géant	Pas de bébé
$(g^m)^0$	yg^{-0}
$(g^m)^1$	yg^{-1}
$\vdots$	$\vdots$
$(g^m)^k$	yg^{-r}
$\vdots$	
$(g^m)^{m-1}$	

▷ On calcule les pas de géant. $\leftarrow m$ étapes.

▷ On calcule les pas de bébé jusqu'à une collision. $\leftarrow \leq m$ étapes.

On a alors $yg^{-r} = (g^m)^k$ donc $y = g^{km+r}$.

Résoudre le problème du log discret

▷ Algorithme Baby-Steps / Giant Steps :

$\rightsquigarrow$ Si G est d'ordre n. On pose $m = \lceil \sqrt{n} \rceil$. Dans $y = g^\ell$, on écrit $\ell = km + r$ avec $0 \leq k, r < m$. On a $yg^{-r} = (g^m)^k$.

Pas de géant	Pas de bébé
$(g^m)^0$	yg^{-0}
$(g^m)^1$	yg^{-1}
$\vdots$	$\vdots$
$(g^m)^k$	yg^{-r}
$\vdots$	
$(g^m)^{m-1}$	

▷ On calcule les pas de géant. $\leftarrow m$ étapes.

▷ On calcule les pas de bébé jusqu'à une collision. $\leftarrow \leq m$ étapes.

On a alors $yg^{-r} = (g^m)^k$ donc $y = g^{km+r}$.

Résoudre le problème du log discret

▷ Algorithme Baby-Steps / Giant Steps :

$\rightsquigarrow$ Si G est d'ordre n. On pose $m = \lceil \sqrt{n} \rceil$. Dans $y = g^\ell$, on écrit $\ell = km + r$ avec $0 \leq k, r < m$. On a $yg^{-r} = (g^m)^k$.

Pas de géant	Pas de bébé
$(g^m)^0$	yg^{-0}
$(g^m)^1$	yg^{-1}
$\vdots$	$\vdots$
$(g^m)^k$	yg^{-r}
$\vdots$	
$(g^m)^{m-1}$	

▷ On calcule les pas de géant. $\leftarrow m$ étapes.

▷ On calcule les pas de bébé jusqu'à une collision. $\leftarrow \leq m$ étapes.

On a alors $yg^{-r} = (g^m)^k$ donc $y = g^{km+r}$.

Résoudre le problème du log discret

▷ Algorithme Baby-Steps / Giant Steps :

$\rightsquigarrow$ Si G est d'ordre n. On pose $m = \lceil \sqrt{n} \rceil$. Dans $y = g^\ell$, on écrit $\ell = km + r$ avec $0 \leq k, r < m$. On a $yg^{-r} = (g^m)^k$.

Pas de géant	Pas de bébé
$(g^m)^0$	yg^{-0}
$(g^m)^1$	yg^{-1}
$\vdots$	$\vdots$
$(g^m)^k$	yg^{-r}
$\vdots$	
$(g^m)^{m-1}$	

▷ On calcule les pas de géant. $\leftarrow m$ étapes.

▷ On calcule les pas de bébé jusqu'à une collision. $\leftarrow \leq m$ étapes.

On a alors $yg^{-r} = (g^m)^k$ donc $y = g^{km+r}$.

Résoudre le problème du log discret

▷ Algorithme Baby-Steps / Giant Steps :

⇝ Si G est d'ordre n. On pose $m = \lceil \sqrt{n} \rceil$. Dans $y = g^\ell$, on écrit $\ell = km + r$ avec $0 \leq k, r < m$. On a $yg^{-r} = (g^m)^k$.

Pas de géant	Pas de bébé
$(g^m)^0$	yg^{-0}
$(g^m)^1$	yg^{-1}
$\vdots$	$\vdots$
$(g^m)^k$	$\vdots$
$\vdots$	yg^{-r}
$(g^m)^{m-1}$	

▷ On calcule les pas de géant. $\leftarrow m$ étapes.

▷ On calcule les pas de bébé jusqu'à une collision. $\leftarrow \leq m$ étapes.

On a alors $yg^{-r} = (g^m)^k$ donc $y = g^{km+r}$.

Résoudre le problème du log discret

▷ Algorithme Baby-Steps / Giant Steps :

$\rightsquigarrow$ Si G est d'ordre n. On pose $m = \lceil \sqrt{n} \rceil$. Dans $y = g^\ell$, on écrit $\ell = km + r$ avec $0 \leq k, r < m$. On a $yg^{-r} = (g^m)^k$.

Pas de géant	Pas de bébé
$(g^m)^0$	yg^{-0}
$(g^m)^1$	yg^{-1}
$\vdots$	$\vdots$
$(g^m)^k$	$\vdots$
$\vdots$	yg^{-r}
$(g^m)^{m-1}$	

▷ On calcule les pas de géant. $\leftarrow m$ étapes.

▷ On calcule les pas de bébé jusqu'à une collision. $\leftarrow \leq m$ étapes.

On a alors $yg^{-r} = (g^m)^k$ donc $y = g^{km+r}$.

Résoudre le problème du log discret

▷ Algorithme Baby-Steps / Giant Steps :

⇝ Si G est d'ordre n. On pose $m = \lceil \sqrt{n} \rceil$. Dans $y = g^\ell$, on écrit $\ell = km + r$ avec $0 \leq k, r < m$. On a $yg^{-r} = (g^m)^k$.

Pas de géant	Pas de bébé
$(g^m)^0$	yg^{-0}
$(g^m)^1$	yg^{-1}
$\vdots$	$\vdots$
$(g^m)^k$	$\vdots$
$\vdots$	yg^{-r}
$(g^m)^{m-1}$	

▷ On calcule les pas de géant. ← m étapes.

▷ On calcule les pas de bébé jusqu'à une collision. ← $\leq m$ étapes.

On a alors $yg^{-r} = (g^m)^k$ donc $y = g^{km+r}$.

Résoudre le problème du log discret

▷ Algorithme Baby-Steps / Giant Steps :

$\rightsquigarrow$ Si G est d'ordre n. On pose $m = \lceil \sqrt{n} \rceil$. Dans $y = g^\ell$, on écrit $\ell = km + r$ avec $0 \leq k, r < m$. On a $yg^{-r} = (g^m)^k$.

Pas de géant	Pas de bébé
$(g^m)^0$	yg^{-0}
$(g^m)^1$	yg^{-1}
$\vdots$	$\vdots$
$(g^m)^k$	$\vdots$
$\vdots$	yg^{-r}
$(g^m)^{m-1}$	

▷ On calcule les pas de géant. $\leftarrow m$ étapes.

▷ On calcule les pas de bébé jusqu'à une collision. $\leftarrow \leq m$ étapes.

On a alors $yg^{-r} = (g^m)^k$ donc $y = g^{km+r}$.

Difficulté du problème du log discret

Théorème de Shoup (1997)

Dans un groupe générique, la résolution du problème du log discret nécessite $O(n^{1/2+\epsilon})$ multiplications.

▷ Dans un groupe générique, le pb du log discret est difficile.

⇝ En principe, la fonction puissance est une fonction à sens unique.

▷ Dans les applications, il n'existe pas de groupe générique.

Difficulté du problème du log discret

Théorème de Shoup (1997)

Dans un groupe générique, la résolution du problème du log discret nécessite $O(n^{1/2+\varepsilon})$ multiplications.

▷ Dans un groupe générique, le pb du log discret est difficile.

⇝ En principe, la fonction puissance est une fonction à sens unique.

▷ Dans les applications, il n'existe pas de groupe générique.

Difficulté du problème du log discret

Théorème de Shoup (1997)

Dans un groupe générique, la résolution du problème du log discret nécessite $O(n^{1/2+\varepsilon})$ multiplications.

▷ Dans un groupe générique, le pb du log discret est difficile.

⇝ En principe, la fonction puissance est une fonction à sens unique.

▷ Dans les applications, il n'existe pas de groupe générique.

Difficulté du problème du log discret

Théorème de Shoup (1997)

Dans un groupe générique, la résolution du problème du log discret nécessite $O(n^{1/2+\varepsilon})$ multiplications.

▷ Dans un groupe générique, le pb du log discret est difficile.

$\rightsquigarrow$ En principe, la fonction puissance est une fonction à sens unique.

▷ Dans les applications, il n'existe pas de groupe générique.

Quels groupes ?

▷ $G = (\mathbb{Z}/p\mathbb{Z}, +)$ et $g = 1$. → Résoudre $y = \ell \cdot 1$: trivial !

▷ $G = ((\mathbb{Z}/p\mathbb{Z})^\times, \cdot)$ et g un générateur (qui existe).

⇝ Il faut que l'ordre du groupe $p - 1$ soit presque premier.

⇝ Il faut trouver un générateur de $(\mathbb{Z}/p\mathbb{Z})^\times$.

⇝ C'est le groupe le plus utilisé. Il faut des grands nombres p car il existe une attaque sous-exponentielle.

Index calculus

On peut résoudre le pb du log discret dans $(\mathbb{Z}/p\mathbb{Z})^\times$ en effectuant $O(e^{k \log(p)^{1/3} (\log \log p)^{2/3}})$ multiplications

▷ Basé sur la factorisation dans $\mathbb{Z}$.

▷ Assez technique à bien mettre en oeuvre.

▷ $G = E(\mathbb{Z}/p\mathbb{Z})$ où E est une courbe elliptique.

Quels groupes ?

▷ $G = (\mathbb{Z}/p\mathbb{Z}, +)$ et $g = 1$. → Résoudre $y = \ell \cdot 1$: trivial !

▷ $G = ((\mathbb{Z}/p\mathbb{Z})^\times, \cdot)$ et g un générateur (qui existe).

⇝ Il faut que l'ordre du groupe $p - 1$ soit presque premier.

⇝ Il faut trouver un générateur de $(\mathbb{Z}/p\mathbb{Z})^\times$.

⇝ C'est le groupe le plus utilisé. Il faut des grands nombres p car il existe une attaque sous-exponentielle.

Index calculus

On peut résoudre le pb du log discret dans $(\mathbb{Z}/p\mathbb{Z})^\times$ en effectuant $O(e^{k \log(p)^{1/3} (\log \log p)^{2/3}})$ multiplications

▷ Basé sur la factorisation dans $\mathbb{Z}$.

▷ Assez technique à bien mettre en oeuvre.

▷ $G = E(\mathbb{Z}/p\mathbb{Z})$ où E est une courbe elliptique.

Quels groupes ?

▷ $G = (\mathbb{Z}/p\mathbb{Z}, +)$ et $g = 1$. → Résoudre $y = \ell \cdot 1$: trivial !

▷ $G = ((\mathbb{Z}/p\mathbb{Z})^\times, \cdot)$ et g un générateur (qui existe).

⇝ Il faut que l'ordre du groupe $p - 1$ soit presque premier.

⇝ Il faut trouver un générateur de $(\mathbb{Z}/p\mathbb{Z})^\times$.

⇝ C'est le groupe le plus utilisé. Il faut des grands nombres p car il existe une attaque sous-exponentielle.

Index calculus

On peut résoudre le pb du log discret dans $(\mathbb{Z}/p\mathbb{Z})^\times$ en effectuant $O(e^{k \log(p)^{1/3} (\log \log p)^{2/3}})$ multiplications

▷ Basé sur la factorisation dans $\mathbb{Z}$.

⇝ ▷ Assez technique à bien mettre en oeuvre.

▷ $G = E(\mathbb{Z}/p\mathbb{Z})$ où E est une courbe elliptique.

Quels groupes ?

▷ $G = (\mathbb{Z}/p\mathbb{Z}, +)$ et $g = 1$. $\rightarrow$ Résoudre $y = \ell \cdot 1$: trivial !

▷ $G = ((\mathbb{Z}/p\mathbb{Z})^\times, \cdot)$ et g un générateur (qui existe).

$\rightsquigarrow$ Il faut que l'ordre du groupe $p - 1$ soit presque premier.

$\rightsquigarrow$ Il faut trouver un générateur de $(\mathbb{Z}/p\mathbb{Z})^\times$.

$\rightsquigarrow$ C'est le groupe le plus utilisé. Il faut des grands nombres p car il existe une attaque sous-exponentielle.

Index calculus

On peut résoudre le pb du log discret dans $(\mathbb{Z}/p\mathbb{Z})^\times$ en effectuant $O(e^{k \log(p)^{1/3} (\log \log p)^{2/3}})$ multiplications

▷ Basé sur la factorisation dans $\mathbb{Z}$.

$\rightsquigarrow$ ▷ Assez technique à bien mettre en oeuvre.

▷ $G = E(\mathbb{Z}/p\mathbb{Z})$ où E est une courbe elliptique.

Quels groupes ?

▷ $G = (\mathbb{Z}/p\mathbb{Z}, +)$ et $g = 1$. $\rightarrow$ Résoudre $y = \ell \cdot 1$: trivial !

▷ $G = ((\mathbb{Z}/p\mathbb{Z})^\times, \cdot)$ et g un générateur (qui existe).

$\rightsquigarrow$ Il faut que l'ordre du groupe $p - 1$ soit presque premier.

$\rightsquigarrow$ Il faut trouver un générateur de $(\mathbb{Z}/p\mathbb{Z})^\times$.

$\rightsquigarrow$ C'est le groupe le plus utilisé. Il faut des grands nombres p car il existe une attaque sous-exponentielle.

Index calculus

On peut résoudre le pb du log discret dans $(\mathbb{Z}/p\mathbb{Z})^\times$ en effectuant $O(e^{k \log(p)^{1/3} (\log \log p)^{2/3}})$ multiplications

▷ Basé sur la factorisation dans $\mathbb{Z}$.

▷ Assez technique à bien mettre en oeuvre.

▷ $G = E(\mathbb{Z}/p\mathbb{Z})$ où E est une courbe elliptique.

Quels groupes ?

▷ $G = (\mathbb{Z}/p\mathbb{Z}, +)$ et $g = 1$. $\rightarrow$ Résoudre $y = \ell \cdot 1$: trivial !

▷ $G = ((\mathbb{Z}/p\mathbb{Z})^\times, \cdot)$ et g un générateur (qui existe).

$\rightsquigarrow$ Il faut que l'ordre du groupe $p - 1$ soit presque premier.

$\rightsquigarrow$ Il faut trouver un générateur de $(\mathbb{Z}/p\mathbb{Z})^\times$.

$\rightsquigarrow$ C'est le groupe le plus utilisé. Il faut des grands nombres p car il existe une attaque sous-exponentielle.

Index calculus

On peut résoudre le pb du log discret dans $(\mathbb{Z}/p\mathbb{Z})^\times$ en effectuant $O(e^{k \log(p)^{1/3} (\log \log p)^{2/3}})$ multiplications

▷ Basé sur la factorisation dans $\mathbb{Z}$.

▷ Assez technique à bien mettre en oeuvre.

▷ $G = E(\mathbb{Z}/p\mathbb{Z})$ où E est une courbe elliptique.

Quels groupes ?

▷ $G = (\mathbb{Z}/p\mathbb{Z}, +)$ et $g = 1$. $\rightarrow$ Résoudre $y = \ell \cdot 1$: trivial !

▷ $G = ((\mathbb{Z}/p\mathbb{Z})^\times, \cdot)$ et g un générateur (qui existe).

$\rightsquigarrow$ Il faut que l'ordre du groupe $p - 1$ soit presque premier.

$\rightsquigarrow$ Il faut trouver un générateur de $(\mathbb{Z}/p\mathbb{Z})^\times$.

$\rightsquigarrow$ C'est le groupe le plus utilisé. Il faut des grands nombres p car il existe une attaque sous-exponentielle.

Index calculus

On peut résoudre le pb du log discret dans $(\mathbb{Z}/p\mathbb{Z})^\times$ en effectuant $O(e^{k \log(p)^{1/3} (\log \log p)^{2/3}})$ multiplications

▷ Basé sur la factorisation dans $\mathbb{Z}$.

▷ Assez technique à bien mettre en oeuvre.

▷ $G = E(\mathbb{Z}/p\mathbb{Z})$ où E est une courbe elliptique.

Quels groupes ?

▷ $G = (\mathbb{Z}/p\mathbb{Z}, +)$ et $g = 1$. $\rightarrow$ Résoudre $y = \ell \cdot 1$: trivial !

▷ $G = ((\mathbb{Z}/p\mathbb{Z})^\times, \cdot)$ et g un générateur (qui existe).

$\rightsquigarrow$ Il faut que l'ordre du groupe $p-1$ soit presque premier.

$\rightsquigarrow$ Il faut trouver un générateur de $(\mathbb{Z}/p\mathbb{Z})^\times$.

$\rightsquigarrow$ C'est le groupe le plus utilisé. Il faut des grands nombres p car il existe une attaque sous-exponentielle.

Index calculus

On peut résoudre le pb du log discret dans $(\mathbb{Z}/p\mathbb{Z})^\times$ en effectuant $O(e^{k \log(p)^{1/3} (\log\log p)^{2/3}})$ multiplications

▷ Basé sur la factorisation dans $\mathbb{Z}$.

▷ Assez technique à bien mettre en oeuvre.

▷ $G = E(\mathbb{Z}/p\mathbb{Z})$ où E est une courbe elliptique.

Quels groupes ?

▷ $G = (\mathbb{Z}/p\mathbb{Z}, +)$ et $g = 1$. $\rightarrow$ Résoudre $y = \ell \cdot 1$: trivial !

▷ $G = ((\mathbb{Z}/p\mathbb{Z})^\times, \cdot)$ et g un générateur (qui existe).

$\rightsquigarrow$ Il faut que l'ordre du groupe $p-1$ soit presque premier.

$\rightsquigarrow$ Il faut trouver un générateur de $(\mathbb{Z}/p\mathbb{Z})^\times$.

$\rightsquigarrow$ C'est le groupe le plus utilisé. Il faut des grands nombres p car il existe une attaque sous-exponentielle.

Index calculus

On peut résoudre le pb du log discret dans $(\mathbb{Z}/p\mathbb{Z})^\times$ en effectuant $O(e^{k \log(p)^{1/3} (\log\log p)^{2/3}})$ multiplications

▷ Basé sur la factorisation dans $\mathbb{Z}$.

▷ Assez technique à bien mettre en oeuvre.

▷ $G = E(\mathbb{Z}/p\mathbb{Z})$ où E est une courbe elliptique.

Problème du logarithme discret dans $(\mathbb{Z}/p\mathbb{Z})^\times$.

▷ Il faut :

$\rightsquigarrow$ Trouver un grand nombre premier p :

$$\#\{p \leq x\} = \pi(x) \sim x/\log(x)$$

Un nombre aléatoire $\ell \approx X$ est premier avec une proba $\sim 1/\log(X)$.

$\rightsquigarrow$ Trouver un grand nombre premier p tel que $p-1$ est presque premier (disons $(p-1)/2 = \ell$ premier).

▷ Idée : soit ℓ un nombre aléatoire, les événements ℓ et $p = 2\ell + 1$ premiers sont (presque) indépendants.

Un nombre aléatoire $\ell \approx X$ vérifie ℓ premier et $2\ell + 1$ premier avec une proba $\sim$

$$\frac{1}{\log(x)^2}$$

$\rightarrow$ Totalement heuristique et conjecturale ! Mais, ça marche !

Problème du logarithme discret dans $(\mathbb{Z}/p\mathbb{Z})^\times$.

▷ Il faut :

⇝ Trouver un grand nombre premier p :

$$\#\{p \leq x\} = \pi(x) \sim x/\log(x)$$

Un nombre aléatoire $\ell \approx X$ est premier avec une proba $\sim 1/\log(X)$.

⇝ Trouver un grand nombre premier p tel que $p-1$ est presque premier (disons $(p-1)/2 = \ell$ premier).

▷ Idée : soit ℓ un nombre aléatoire, les événements ℓ et $p = 2\ell + 1$ premiers sont (presque) indépendants.

Un nombre aléatoire $\ell \approx X$ vérifie ℓ premier et $2\ell + 1$ premier avec une proba $\sim$

$$\frac{1}{\log(x)^2}$$

→ Totalement heuristique et conjecturale ! Mais, ça marche !

Problème du logarithme discret dans $(\mathbb{Z}/p\mathbb{Z})^\times$.

▷ Il faut :

⇝ Trouver un grand nombre premier p :

$$\#\{p \leq x\} = \pi(x) \sim x/\log(x)$$

Un nombre aléatoire $\ell \approx X$ est premier avec une proba $\sim 1/\log(X)$.

⇝ Trouver un grand nombre premier p tel que $p-1$ est presque premier (disons $(p-1)/2 = \ell$ premier).

▷ Idée : soit ℓ un nombre aléatoire, les événements ℓ et $p = 2\ell + 1$ premiers sont (presque) indépendants.

Un nombre aléatoire $\ell \approx X$ vérifie ℓ premier et $2\ell + 1$ premier avec une proba $\sim$

$$\frac{1}{\log(x)^2}$$

→ Totalement heuristique et conjecturale ! Mais, ça marche !

Problème du logarithme discret dans $(\mathbb{Z}/p\mathbb{Z})^\times$.

▷ Il faut :

⇝ Trouver un grand nombre premier p :

$$\#\{p \leq x\} = \pi(x) \sim x/\log(x)$$

Un nombre aléatoire $\ell \approx X$ est premier avec une proba $\sim 1/\log(X)$.

⇝ Trouver un grand nombre premier p tel que $p-1$ est presque premier (disons $(p-1)/2 = \ell$ premier).

▷ Idée : soit ℓ un nombre aléatoire, les événements ℓ et $p = 2\ell + 1$ premiers sont (presque) indépendants.

Un nombre aléatoire $\ell \approx X$ vérifie ℓ premier et $2\ell + 1$ premier avec une proba $\sim$

$$\frac{1}{\log(x)^2}$$

→ Totalement heuristique et conjecturale ! Mais, ça marche !

Problème du logarithme discret dans $(\mathbb{Z}/p\mathbb{Z})^\times$.

▷ Il faut :

⇝ Trouver un grand nombre premier p :

$$\#\{p \leq x\} = \pi(x) \sim x/\log(x)$$

Un nombre aléatoire $\ell \approx X$ est premier avec une proba $\sim 1/\log(X)$.

⇝ Trouver un grand nombre premier p tel que $p-1$ est presque premier (disons $(p-1)/2 = \ell$ premier).

▷ Idée : soit ℓ un nombre aléatoire, les événements ℓ et $p = 2\ell + 1$ premiers sont (presque) indépendants.

Un nombre aléatoire $\ell \approx X$ vérifie ℓ premier et $2\ell+1$ premier avec une proba $\sim$

$$\left(2 \prod_{\substack{q \text{ impair} \\ q \text{ premier}}} \frac{1-\frac{2}{q}}{\left(1-\frac{1}{q}\right)^2}\right) \frac{1}{\log(x)^2} \approx 1.32 \frac{1}{\log(X)^2}.$$

→ Totalement heuristique et conjecturale ! Mais, ça marche !

Problème du logarithme discret dans $(\mathbb{Z}/p\mathbb{Z})^\times$.

▷ Il faut :

⇝ Trouver un grand nombre premier p :

$$\#\{p \leq x\} = \pi(x) \sim x/\log(x)$$

Un nombre aléatoire $\ell \approx X$ est premier avec une proba $\sim 1/\log(X)$.

⇝ Trouver un grand nombre premier p tel que $p-1$ est presque premier (disons $(p-1)/2 = \ell$ premier).

▷ Idée : soit ℓ un nombre aléatoire, les événements ℓ et $p = 2\ell + 1$ premiers sont (presque) indépendants.

Un nombre aléatoire $\ell \approx X$ vérifie ℓ premier et $2\ell+1$ premier avec une proba $\sim$

$$\left(2 \prod_{\substack{q \text{ impair} \\ q \text{ premier}}} \frac{1-\frac{2}{q}}{\left(1-\frac{1}{q}\right)^2}\right) \frac{1}{\log(x)^2} \approx 1.32 \frac{1}{\log(X)^2}.$$

→ Totalement heuristique et conjecturale ! Mais, ça marche !

Problème du logarithme discret dans $(\mathbb{Z}/p\mathbb{Z})^\times$.

▷ Il faut :

$\rightsquigarrow$ Trouver un grand nombre premier p :

$$\#\{p \leq x\} = \pi(x) \sim x/\log(x)$$

Un nombre aléatoire $\ell \approx X$ est premier avec une proba $\sim 1/\log(X)$.

$\rightsquigarrow$ Trouver un grand nombre premier p tel que $p-1$ est presque premier (disons $(p-1)/2 = \ell$ premier).

▷ Idée : soit ℓ un nombre aléatoire, les événements ℓ et $p = 2\ell + 1$ premiers sont (presque) indépendants.

Un nombre aléatoire $\ell \approx X$ vérifie ℓ premier et $2\ell+1$ premier avec une proba $\sim$

$$\left(2 \prod_{\substack{q \text{ impair} \\ q \text{ premier}}} \frac{1 - \frac{2}{q}}{\left(1 - \frac{1}{q}\right)^2} \right) \frac{1}{\log(x)^2} \approx 1.32 \frac{1}{\log(X)^2}.$$

$\rightarrow$ Totalement heuristique et conjecturale ! Mais, ça marche !

Problème du logarithme discret dans $(\mathbb{Z}/p\mathbb{Z})^\times$.

▷ Il faut :

$\rightsquigarrow$ Trouver un grand nombre premier p :

$$\#\{p \leq x\} = \pi(x) \sim x/\log(x)$$

Un nombre aléatoire $\ell \approx X$ est premier avec une proba $\sim 1/\log(X)$.

$\rightsquigarrow$ Trouver un grand nombre premier p tel que $p-1$ est presque premier (disons $(p-1)/2 = \ell$ premier).

▷ Idée : soit ℓ un nombre aléatoire, les événements ℓ et $p = 2\ell + 1$ premiers sont (presque) indépendants.

Un nombre aléatoire $\ell \approx X$ vérifie ℓ premier et $2\ell+1$ premier avec une proba $\sim$

$$\left(2 \prod_{\substack{q \text{ impair} \\ q \text{ premier}}} \frac{1-\frac{2}{q}}{\left(1-\frac{1}{q}\right)^2}\right) \frac{1}{\log(x)^2} \approx 1.32 \frac{1}{\log(X)^2}.$$

$\rightarrow$ Totalement heuristique et conjecturale ! Mais, ça marche !

Problème du logarithme discret dans $(\mathbb{Z}/p\mathbb{Z})^\times$.

Soit p un nombre premier tel que $\frac{p-1}{2}$ est premier.

Pour créer un problème du logarithme discret dans $G = (\mathbb{Z}/p\mathbb{Z})^\times$, il faut un générateur du groupe.

▷ **Théorème :** Si p est premier, $(\mathbb{Z}/p\mathbb{Z})^\times$ est cyclique.

$\rightsquigarrow$ Dans notre cas, g est un générateur de G ssi $g^\ell = -1$ ($\ell = \frac{p-1}{2}$).

$\rightsquigarrow$ On essaie $g = 2$, $g = 3$, $g = 5$, etc.

▷ **Théorème :** Il existe un générateur $g \ll p^{1/4+\varepsilon}$. ($\hookrightarrow$ exponentiel.)

▷ **Théorème :** Sous GRH, il existe un générateur $g \ll \log(p)^6$.

$\rightsquigarrow$ **Dans les faits :** ça marche en quelques essais.

▷ En général, si $p-1$ n'est pas factorisé, il est difficile de trouver un générateur de G.

Problème du logarithme discret dans $(\mathbb{Z}/p\mathbb{Z})^\times$.

Soit p un nombre premier tel que $\frac{p-1}{2}$ est premier.

Pour créer un problème du logarithme discret dans $G = (\mathbb{Z}/p\mathbb{Z})^\times$, il faut un générateur du groupe.

▷ **Théorème :** Si p est premier, $(\mathbb{Z}/p\mathbb{Z})^\times$ est cyclique.

$\rightsquigarrow$ Dans notre cas, g est un générateur de G ssi $g^\ell = -1$ ($\ell = \frac{p-1}{2}$).

$\rightsquigarrow$ On essaie $g = 2$, $g = 3$, $g = 5$, etc.

▷ **Théorème :** Il existe un générateur $g \ll p^{1/4+\varepsilon}$. ($\hookrightarrow$ exponentiel.)

▷ **Théorème :** Sous GRH, il existe un générateur $g \ll \log(p)^6$.

$\rightsquigarrow$ **Dans les faits :** ça marche en quelques essais.

▷ En général, si $p-1$ n'est pas factorisé, il est difficile de trouver un générateur de G.

Problème du logarithme discret dans $(\mathbb{Z}/p\mathbb{Z})^\times$.

Soit p un nombre premier tel que $\frac{p-1}{2}$ est premier.

Pour créer un problème du logarithme discret dans $G = (\mathbb{Z}/p\mathbb{Z})^\times$, il faut un générateur du groupe.

▷ **Théorème :** Si p est premier, $(\mathbb{Z}/p\mathbb{Z})^\times$ est cyclique.

$\rightsquigarrow$ Dans notre cas, g est un générateur de G ssi $g^\ell = -1$ ($\ell = \frac{p-1}{2}$).

$\rightsquigarrow$ On essaie $g = 2$, $g = 3$, $g = 5$, etc.

▷ **Théorème :** Il existe un générateur $g \ll p^{1/4+\varepsilon}$. ($\hookrightarrow$ exponentiel.)

▷ **Théorème :** Sous GRH, il existe un générateur $g \ll \log(p)^6$.

$\rightsquigarrow$ **Dans les faits :** ça marche en quelques essais.

▷ En général, si $p-1$ n'est pas factorisé, il est difficile de trouver un générateur de G.

Problème du logarithme discret dans $(\mathbb{Z}/p\mathbb{Z})^\times$.

Soit p un nombre premier tel que $\frac{p-1}{2}$ est premier.

Pour créer un problème du logarithme discret dans $G = (\mathbb{Z}/p\mathbb{Z})^\times$, il faut un générateur du groupe.

▷ **Théorème :** Si p est premier, $(\mathbb{Z}/p\mathbb{Z})^\times$ est cyclique.

⇝ Dans notre cas, g est un générateur de G ssi $g^\ell = -1$ ($\ell = \frac{p-1}{2}$).

⇝ On essaie $g = 2$, $g = 3$, $g = 5$, etc.

▷ **Théorème :** Il existe un générateur $g \ll p^{1/4+\varepsilon}$. ($\hookrightarrow$ exponentiel.)

▷ **Théorème :** Sous GRH, il existe un générateur $g \ll \log(p)^6$.

⇝ **Dans les faits :** ça marche en quelques essais.

▷ En général, si $p-1$ n'est pas factorisé, il est difficile de trouver un générateur de G.

Problème du logarithme discret dans $(\mathbb{Z}/p\mathbb{Z})^\times$.

Soit p un nombre premier tel que $\frac{p-1}{2}$ est premier.

Pour créer un problème du logarithme discret dans $G = (\mathbb{Z}/p\mathbb{Z})^\times$, il faut un générateur du groupe.

▷ **Théorème :** Si p est premier, $(\mathbb{Z}/p\mathbb{Z})^\times$ est cyclique.

⇝ Dans notre cas, g est un générateur de G ssi $g^\ell = -1$ ($\ell = \frac{p-1}{2}$).

⇝ On essaie $g = 2$, $g = 3$, $g = 5$, etc.

▷ **Théorème :** Il existe un générateur $g \ll p^{1/4+\varepsilon}$. ($\hookrightarrow$ exponentiel.)

▷ **Théorème :** Sous GRH, il existe un générateur $g \ll \log(p)^6$.

⇝ **Dans les faits :** ça marche en quelques essais.

▷ En général, si $p-1$ n'est pas factorisé, il est difficile de trouver un générateur de G.

Problème du logarithme discret dans $(\mathbb{Z}/p\mathbb{Z})^\times$.

Soit p un nombre premier tel que $\frac{p-1}{2}$ est premier.

Pour créer un problème du logarithme discret dans $G = (\mathbb{Z}/p\mathbb{Z})^\times$, il faut un générateur du groupe.

▷ **Théorème :** Si p est premier, $(\mathbb{Z}/p\mathbb{Z})^\times$ est cyclique.

⇝ Dans notre cas, g est un générateur de G ssi $g^\ell = -1$ ($\ell = \frac{p-1}{2}$).

⇝ On essaie $g = 2$, $g = 3$, $g = 5$, etc.

▷ **Théorème :** Il existe un générateur $g \ll p^{1/4+\varepsilon}$. ($\rightarrow$ exponentiel.)

▷ **Théorème :** Sous GRH, il existe un générateur $g \ll \log(p)^6$.

⇝ **Dans les faits :** ça marche en quelques essais.

▷ En général, si $p-1$ n'est pas factorisé, il est difficile de trouver un générateur de G.

Problème du logarithme discret dans $(\mathbb{Z}/p\mathbb{Z})^\times$.

Soit p un nombre premier tel que $\frac{p-1}{2}$ est premier.

Pour créer un problème du logarithme discret dans $G = (\mathbb{Z}/p\mathbb{Z})^\times$, il faut un générateur du groupe.

▷ **Théorème :** Si p est premier, $(\mathbb{Z}/p\mathbb{Z})^\times$ est cyclique.

⇝ Dans notre cas, g est un générateur de G ssi $g^\ell = -1$ ($\ell = \frac{p-1}{2}$).

⇝ On essaie $g = 2$, $g = 3$, $g = 5$, etc.

▷ **Théorème :** Il existe un générateur $g \ll p^{1/4+\varepsilon}$. ($\to$ exponentiel.)

▷ **Théorème :** Sous GRH, il existe un générateur $g \ll \log(p)^6$.

⇝ **Dans les faits :** ça marche en quelques essais.

▷ En général, si $p-1$ n'est pas factorisé, il est difficile de trouver un générateur de G.

Problème du logarithme discret dans $(\mathbb{Z}/p\mathbb{Z})^\times$.

Soit p un nombre premier tel que $\frac{p-1}{2}$ est premier.

Pour créer un problème du logarithme discret dans $G = (\mathbb{Z}/p\mathbb{Z})^\times$, il faut un générateur du groupe.

▷ **Théorème :** Si p est premier, $(\mathbb{Z}/p\mathbb{Z})^\times$ est cyclique.

$\rightsquigarrow$ Dans notre cas, g est un générateur de G ssi $g^\ell = -1$ ($\ell = \frac{p-1}{2}$).

$\rightsquigarrow$ On essaie $g = 2$, $g = 3$, $g = 5$, etc.

▷ **Théorème :** Il existe un générateur $g \ll p^{1/4+\varepsilon}$. ($\rightarrow$ exponentiel.)

▷ **Théorème :** Sous GRH, il existe un générateur $g \ll \log(p)^6$.

$\rightsquigarrow$ **Dans les faits :** ça marche en quelques essais.

▷ En général, si $p-1$ n'est pas factorisé, il est difficile de trouver un générateur de G.

Problème du logarithme discret dans $(\mathbb{Z}/p\mathbb{Z})^\times$.

Soit p un nombre premier tel que $\frac{p-1}{2}$ est premier.

Pour créer un problème du logarithme discret dans $G = (\mathbb{Z}/p\mathbb{Z})^\times$, il faut un générateur du groupe.

▷ **Théorème :** Si p est premier, $(\mathbb{Z}/p\mathbb{Z})^\times$ est cyclique.

$\rightsquigarrow$ Dans notre cas, g est un générateur de G ssi $g^\ell = -1$ ($\ell = \frac{p-1}{2}$).

$\rightsquigarrow$ On essaie $g = 2$, $g = 3$, $g = 5$, etc.

▷ **Théorème :** Il existe un générateur $g \ll p^{1/4+\varepsilon}$. ($\rightarrow$ exponentiel.)

▷ **Théorème :** Sous GRH, il existe un générateur $g \ll \log(p)^6$.

$\rightsquigarrow$ **Dans les faits :** ça marche en quelques essais.

▷ En général, si $p-1$ n'est pas factorisé, il est difficile de trouver un générateur de G.

Courbes elliptiques

Soit k un corps de caractéristique $\neq 2, 3$ (ici $k = \mathbb{Z}/p\mathbb{Z}$ avec p un grand nombre premier ou $k = \mathbb{R}$).

Une courbe elliptique E sur k est une courbe lisse définie par :

$$y^2 = x^3 + ax + b, \quad \text{où} \quad a, b \in k.$$

$\rightsquigarrow$ Lisse : $4a^3 + 27b^2 \neq 0$.

$\rightsquigarrow$ Les points de E :

$$E(k) = \{(x, y) \in k^2 : y^2 = x^3 + ax + b\} \cup \{\mathcal{O}\}.$$

Le point $\mathcal{O}$ s'appelle le point à l'infini.

$\rightsquigarrow$ Fait central : On munit $E(k)$ d'une loi de groupe abélien de nature géométrique :

Si $P, Q \in E(k)$, la droite passant par P et Q recoupe E en un troisième point (x_1, y_1). Par définition : $P + Q = (x_1, -y_1)$.

▷ Le point à l'infini, $\mathcal{O}$, est le neutre de cette addition.

▷ L'opposé de $P = (x_P, y_P)$ est $-P = (x_P, -y_P)$.

Courbes elliptiques

Soit k un corps de caractéristique $\neq 2, 3$ (ici $k = \mathbb{Z}/p\mathbb{Z}$ avec p un grand nombre premier ou $k = \mathbb{R}$).

Une courbe elliptique E sur k est une courbe lisse définie par :

$$y^2 = x^3 + ax + b, \quad \text{où} \quad a, b \in k.$$

$\rightsquigarrow$ Lisse : $4a^3 + 27b^2 \neq 0$.

$\rightsquigarrow$ Les points de E :

$$E(k) = \{(x, y) \in k^2 : y^2 = x^3 + ax + b\} \cup \{\mathcal{O}\}.$$

Le point $\mathcal{O}$ s'appelle le point à l'infini.

$\rightsquigarrow$ Fait central : On munit $E(k)$ d'une loi de groupe abélien de nature géométrique :

Si $P, Q \in E(k)$, la droite passant par P et Q recoupe E en un troisième point (x_1, y_1). Par définition : $P + Q = (x_1, -y_1)$.

▷ Le point à l'infini, $\mathcal{O}$, est le neutre de cette addition.

▷ L'opposé de $P = (x_P, y_P)$ est $-P = (x_P, -y_P)$.

Courbes elliptiques

Soit k un corps de caractéristique $\neq 2, 3$ (ici $k = \mathbb{Z}/p\mathbb{Z}$ avec p un grand nombre premier ou $k = \mathbb{R}$).

Une courbe elliptique E sur k est une courbe lisse définie par :

$$y^2 = x^3 + ax + b, \quad \text{où} \quad a, b \in k.$$

$\rightsquigarrow$ Lisse : $4a^3 + 27b^2 \neq 0$.

$\rightsquigarrow$ Les points de E :

$$E(k) = \{(x, y) \in k^2 : y^2 = x^3 + ax + b\} \cup \{\mathcal{O}\}.$$

Le point $\mathcal{O}$ s'appelle le point à l'infini.

$\rightsquigarrow$ Fait central : On munit $E(k)$ d'une loi de groupe abélien de nature géométrique :

Si $P, Q \in E(k)$, la droite passant par P et Q recoupe E en un troisième point (x_1, y_1). Par définition : $P + Q = (x_1, -y_1)$.

▷ Le point à l'infini, $\mathcal{O}$, est le neutre de cette addition.

▷ L'opposé de $P = (x_P, y_P)$ est $-P = (x_P, -y_P)$.

Courbes elliptiques

Soit k un corps de caractéristique $\neq 2, 3$ (ici $k = \mathbb{Z}/p\mathbb{Z}$ avec p un grand nombre premier ou $k = \mathbb{R}$).

Une courbe elliptique E sur k est une courbe lisse définie par :

$$y^2 = x^3 + ax + b, \quad \text{où} \quad a, b \in k.$$

$\rightsquigarrow$ Lisse : $4a^3 + 27b^2 \neq 0$.

$\rightsquigarrow$ Les points de E :

$$E(k) = \left\{(x, y) \in k^2 : y^2 = x^3 + ax + b\right\} \cup \{\mathcal{O}\}.$$

Le point $\mathcal{O}$ s'appelle le point à l'infini.

$\rightsquigarrow$ Fait central : On munit $E(k)$ d'une loi de groupe abélien de nature géométrique :

Si $P, Q \in E(k)$, la droite passant par P et Q recoupe E en un troisième point (x_1, y_1). Par définition : $P + Q = (x_1, -y_1)$.

▷ Le point à l'infini, $\mathcal{O}$, est le neutre de cette addition.

▷ L'opposé de $P = (x_P, y_P)$ est $-P = (x_P, -y_P)$.

Courbes elliptiques

Soit k un corps de caractéristique $\neq 2, 3$ (ici $k = \mathbb{Z}/p\mathbb{Z}$ avec p un grand nombre premier ou $k = \mathbb{R}$).

Une courbe elliptique E sur k est une courbe lisse définie par :

$$y^2 = x^3 + ax + b, \quad \text{où} \quad a, b \in k.$$

$\rightsquigarrow$ Lisse : $4a^3 + 27b^2 \neq 0$.

$\rightsquigarrow$ Les points de E :

$$E(k) = \left\{(x, y) \in k^2 : y^2 = x^3 + ax + b\right\} \cup \{\mathcal{O}\}.$$

Le point $\mathcal{O}$ s'appelle le point à l'infini.

$\rightsquigarrow$ Fait central : On munit $E(k)$ d'une loi de groupe abélien de nature géométrique :

Si $P, Q \in E(k)$, la droite passant par P et Q recoupe E en un troisième point (x_1, y_1). Par définition : $P + Q = (x_1, -y_1)$.

▷ Le point à l'infini, $\mathcal{O}$, est le neutre de cette addition.

▷ L'opposé de $P = (x_P, y_P)$ est $-P = (x_P, -y_P)$.

Courbes elliptiques

Soit k un corps de caractéristique $\neq 2, 3$ (ici $k = \mathbb{Z}/p\mathbb{Z}$ avec p un grand nombre premier ou $k = \mathbb{R}$).

Une courbe elliptique E sur k est une courbe lisse définie par :

$$y^2 = x^3 + ax + b, \quad \text{où} \quad a, b \in k.$$

$\rightsquigarrow$ Lisse : $4a^3 + 27b^2 \neq 0$.

$\rightsquigarrow$ Les points de E :

$$E(k) = \{(x, y) \in k^2 : y^2 = x^3 + ax + b\} \cup \{\mathcal{O}\}.$$

Le point $\mathcal{O}$ s'appelle le point à l'infini.

$\rightsquigarrow$ Fait central : On munit $E(k)$ d'une loi de groupe abélien de nature géométrique :

Si $P, Q \in E(k)$, la droite passant par P et Q recoupe E en un troisième point (x_1, y_1). Par définition : $P + Q = (x_1, -y_1)$.

▷ Le point à l'infini, $\mathcal{O}$, est le neutre de cette addition.

▷ L'opposé de $P = (x_P, y_P)$ est $-P = (x_P, -y_P)$.

Courbes elliptiques

Soit k un corps de caractéristique $\neq 2, 3$ (ici $k = \mathbb{Z}/p\mathbb{Z}$ avec p un grand nombre premier ou $k = \mathbb{R}$).

Une courbe elliptique E sur k est une courbe lisse définie par :

$$y^2 = x^3 + ax + b, \quad \text{où} \quad a, b \in k.$$

$\rightsquigarrow$ Lisse : $4a^3 + 27b^2 \neq 0$.

$\rightsquigarrow$ Les points de E :

$$E(k) = \{(x, y) \in k^2 : y^2 = x^3 + ax + b\} \cup \{\mathcal{O}\}.$$

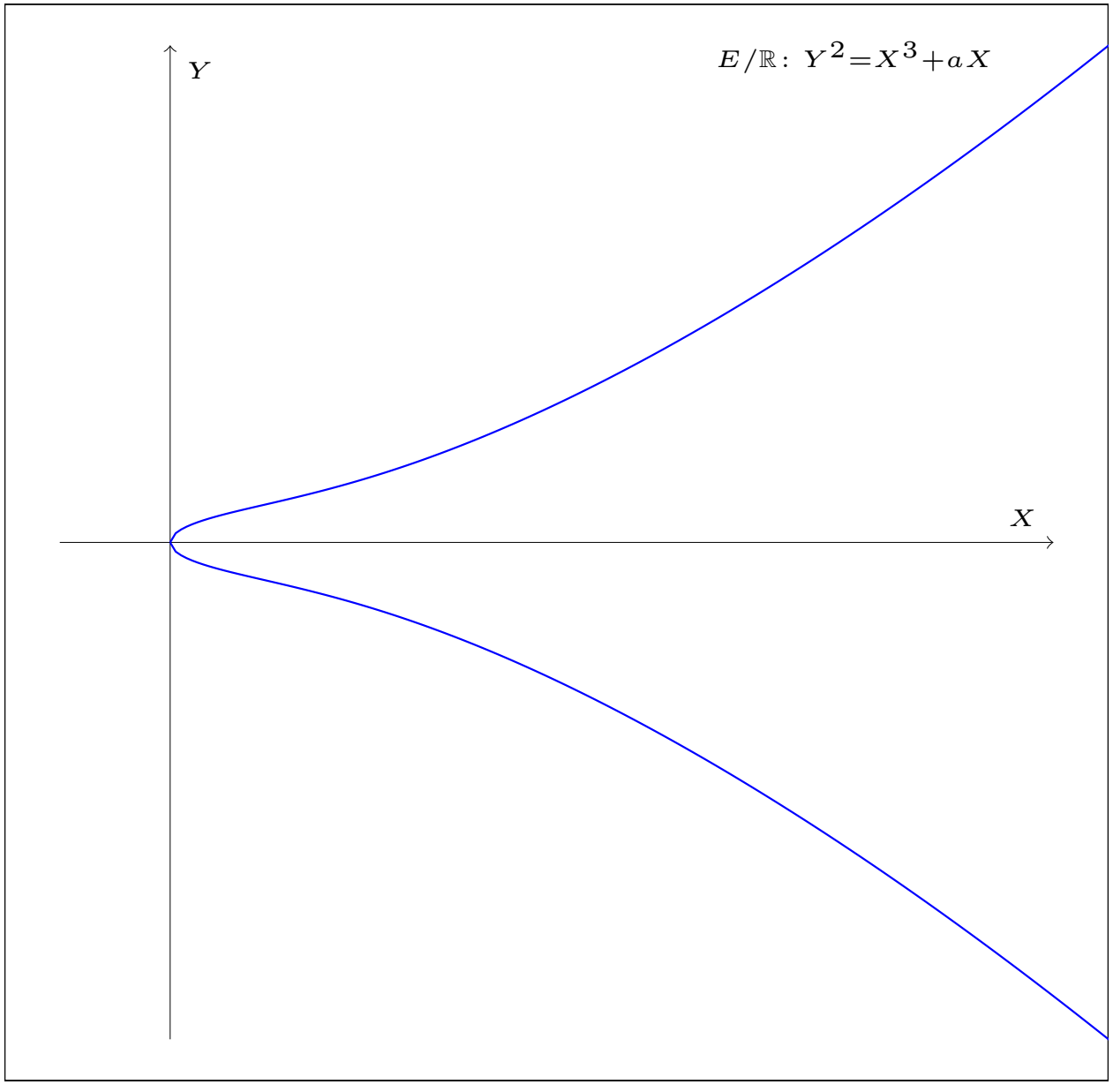
Le point $\mathcal{O}$ s'appelle le point à l'infini.

$\rightsquigarrow$ Fait central : On munit $E(k)$ d'une loi de groupe abélien de nature géométrique :

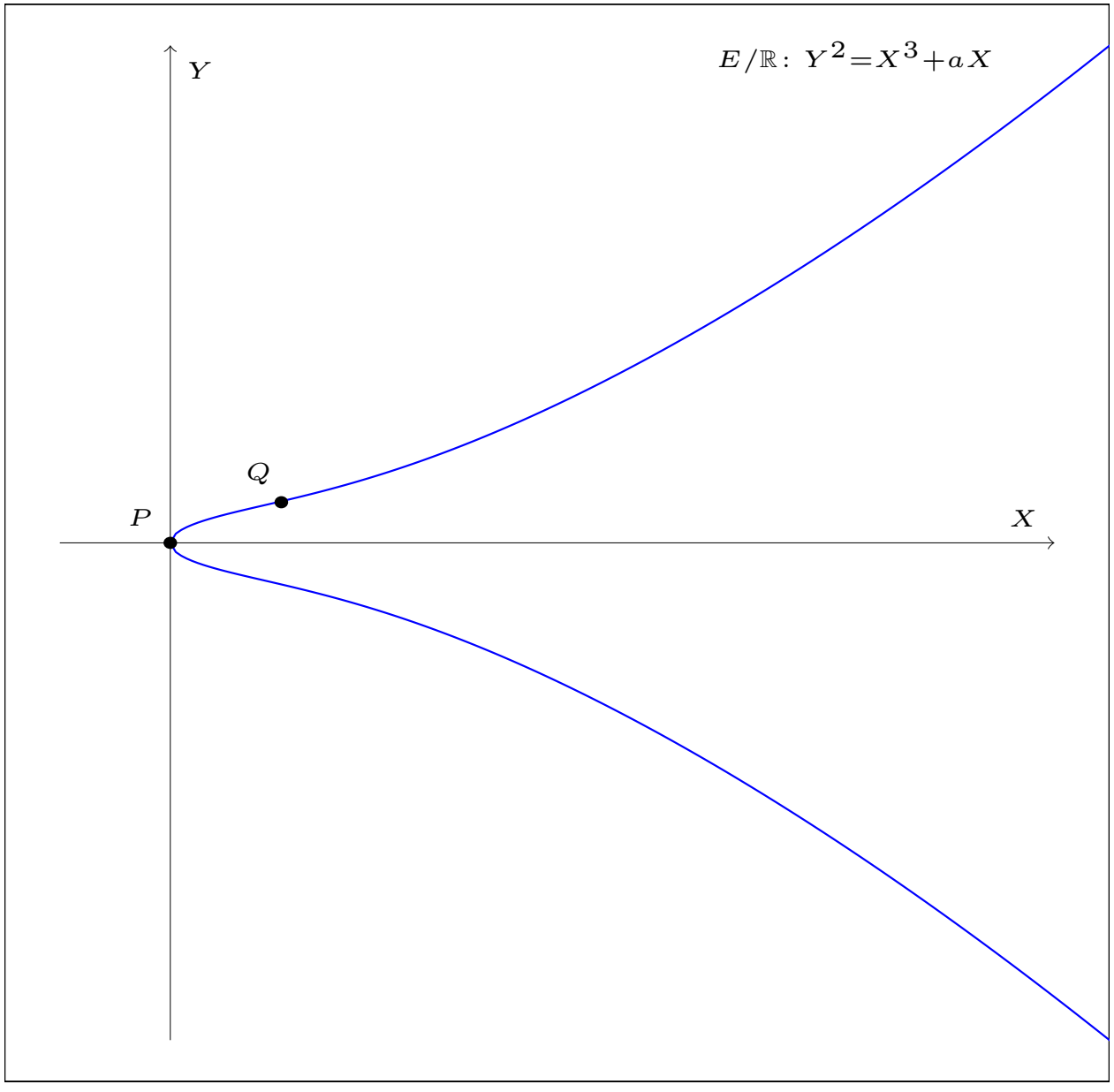
Si $P, Q \in E(k)$, la droite passant par P et Q recoupe E en un troisième point (x_1, y_1). Par définition : $P + Q = (x_1, -y_1)$.

$\triangleright$ Le point à l'infini, $\mathcal{O}$, est le neutre de cette addition.
$\triangleright$ L'opposé de $P = (x_P, y_P)$ est $-P = (x_P, -y_P)$.

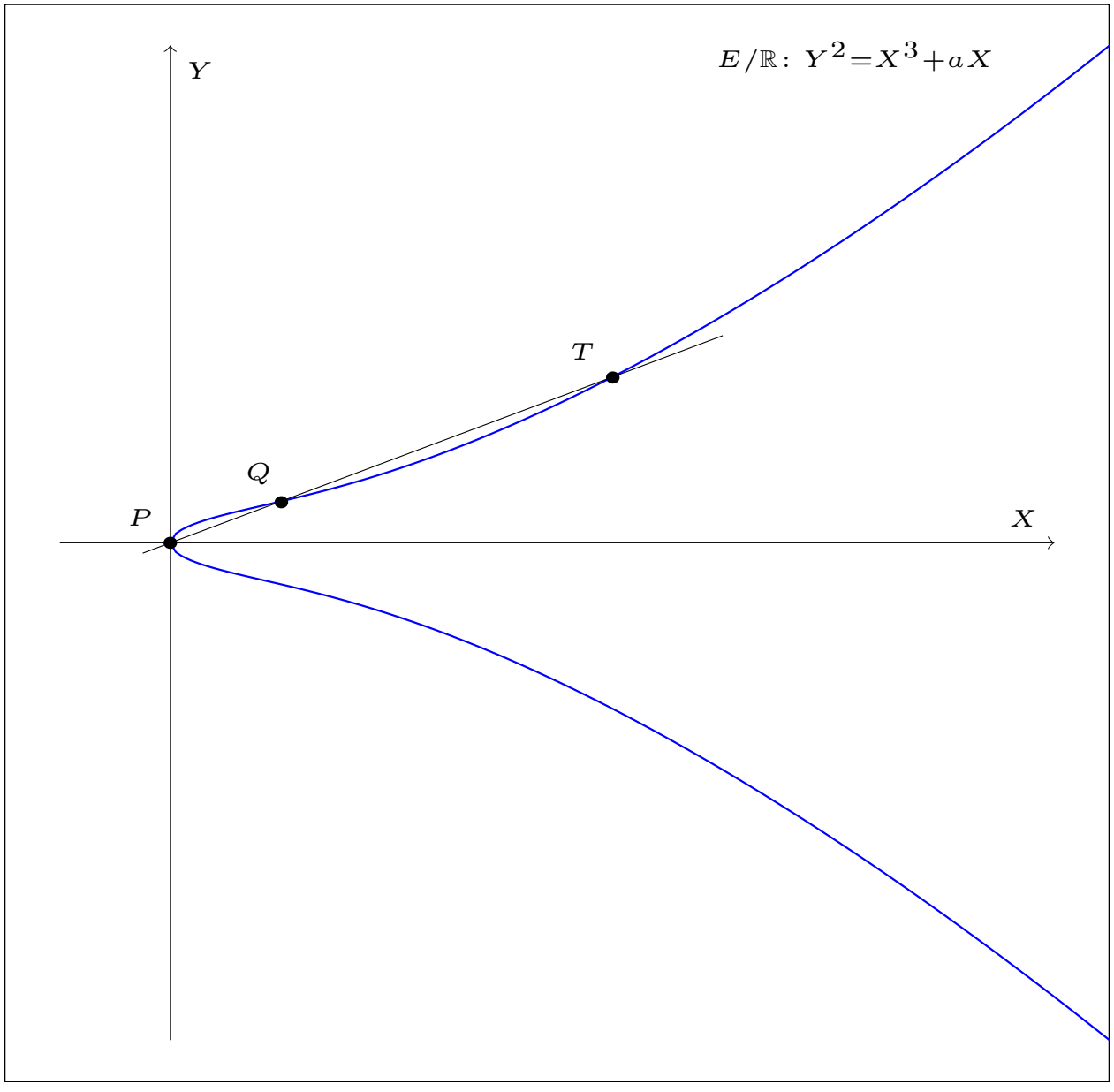
Courbes elliptiques : loi de groupe



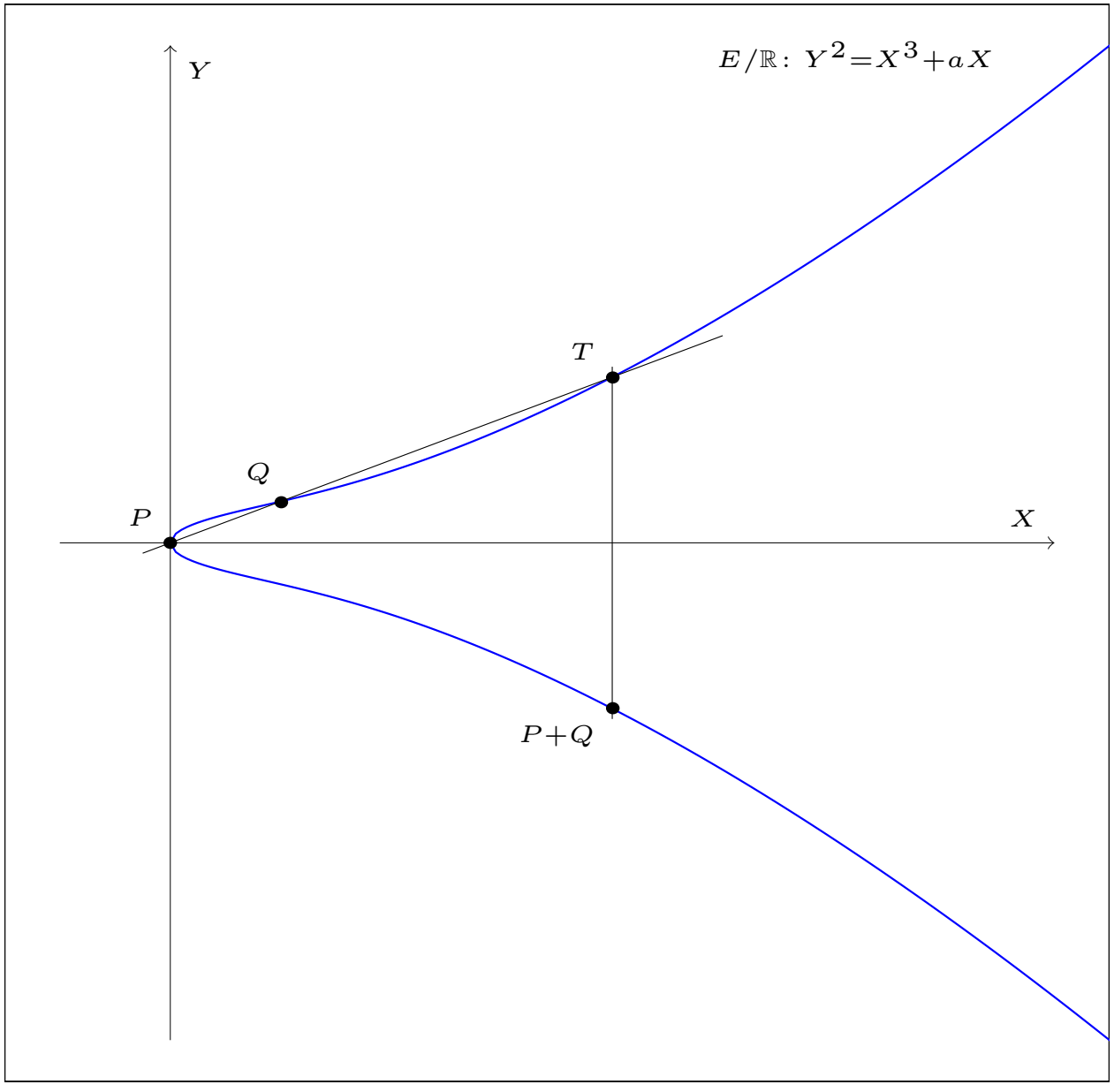
Courbes elliptiques : loi de groupe



Courbes elliptiques : loi de groupe



Courbes elliptiques : loi de groupe



Courbes elliptiques sur $\mathbb{Z}/p\mathbb{Z}$

▷ On prend $k = \mathbb{F}_7$. Soit $E : y^2 = x^3 + x + 4$. On a

$$E(\mathbb{F}_7) =< (0,2) >= \{\mathcal{O}, (0,2), 2 \cdot (0,2), \cdots, 9 \cdot (0,2)\}$$

On a donc $(2,0) = \ell \cdot (0,2)$ pour un certain ℓ. Que vaut ℓ ?.

$\rightsquigarrow$ C'est le problème du logarithme discret dans $E(\mathbb{F}_7)$! (ici $\ell = 5$).

$\rightsquigarrow$ $E(\mathbb{F}_7)$ est cyclique (c'est fréquent).

Si E est définie sur $\mathbb{F}_p$ avec p grand, le problème du log discret semble compliqué : utile pour la cryptographie.

Il faut pouvoir le mettre en place :

$\rightsquigarrow$ Trouver E et p tels que $\sharp E(\mathbb{F}_p)$ est (presque) premier ;

$\rightsquigarrow$ Il faut savoir calculer $\sharp E(\mathbb{F}_p)$;

$\rightsquigarrow$ Il faut savoir trouver un point $G \in E(\mathbb{F}_p)$;

$\rightsquigarrow$ Il faut que E ne soit pas "faible"...

Courbes elliptiques sur $\mathbb{Z}/p\mathbb{Z}$

▷ On prend $k = \mathbb{F}_7$. Soit $E : y^2 = x^3 + x + 4$. On a

$$E(\mathbb{F}_7) =< (0,2) >= \{\mathcal{O}, (0,2), 2 \cdot (0,2), \cdots, 9 \cdot (0,2)\}$$

On a donc $(2,0) = \ell \cdot (0,2)$ pour un certain ℓ. Que vaut ℓ ?.

$\rightsquigarrow$ C'est le problème du logarithme discret dans $E(\mathbb{F}_7)$! (ici $\ell = 5$).

$\rightsquigarrow$ $E(\mathbb{F}_7)$ est cyclique (c'est fréquent).

Si E est définie sur $\mathbb{F}_p$ avec p grand, le problème du log discret semble compliqué : utile pour la cryptographie.

Il faut pouvoir le mettre en place :

$\rightsquigarrow$ Trouver E et p tels que $\sharp E(\mathbb{F}_p)$ est (presque) premier ;

$\rightsquigarrow$ Il faut savoir calculer $\sharp E(\mathbb{F}_p)$;

$\rightsquigarrow$ Il faut savoir trouver un point $G \in E(\mathbb{F}_p)$;

$\rightsquigarrow$ Il faut que E ne soit pas "faible"...

Courbes elliptiques sur $\mathbb{Z}/p\mathbb{Z}$

▷ On prend $k = \mathbb{F}_7$. Soit $E : y^2 = x^3 + x + 4$. On a

$$E(\mathbb{F}_7) =< (0,2) >= \{\mathcal{O}, (0,2), 2 \cdot (0,2), \cdots, 9 \cdot (0,2)\}$$

On a donc $(2,0) = \ell \cdot (0,2)$ pour un certain ℓ. Que vaut ℓ ?.

$\rightsquigarrow$ C'est le problème du logarithme discret dans $E(\mathbb{F}_7)$! (ici $\ell = 5$).

$\rightsquigarrow$ $E(\mathbb{F}_7)$ est cyclique (c'est fréquent).

Si E est définie sur $\mathbb{F}_p$ avec p grand, le problème du log discret semble compliqué : utile pour la cryptographie.

Il faut pouvoir le mettre en place :

$\rightsquigarrow$ Trouver E et p tels que $\sharp E(\mathbb{F}_p)$ est (presque) premier ;

$\rightsquigarrow$ Il faut savoir calculer $\sharp E(\mathbb{F}_p)$;

$\rightsquigarrow$ Il faut savoir trouver un point $G \in E(\mathbb{F}_p)$;

$\rightsquigarrow$ Il faut que E ne soit pas "faible"...

Courbes elliptiques sur $\mathbb{Z}/p\mathbb{Z}$

▷ On prend $k = \mathbb{F}_7$. Soit $E : y^2 = x^3 + x + 4$. On a

$$E(\mathbb{F}_7) =< (0,2) >= \{\mathcal{O}, (0,2), 2 \cdot (0,2), \cdots, 9 \cdot (0,2)\}$$

On a donc $(2,0) = \ell \cdot (0,2)$ pour un certain ℓ. Que vaut ℓ ?.

$\rightsquigarrow$ C'est le problème du logarithme discret dans $E(\mathbb{F}_7)$! (ici $\ell = 5$).

$\rightsquigarrow$ $E(\mathbb{F}_7)$ est cyclique (c'est fréquent).

Si E est définie sur $\mathbb{F}_p$ avec p grand, le problème du log discret semble compliqué : utile pour la cryptographie.

Il faut pouvoir le mettre en place :

$\rightsquigarrow$ Trouver E et p tels que $\sharp E(\mathbb{F}_p)$ est (presque) premier ;

$\rightsquigarrow$ Il faut savoir calculer $\sharp E(\mathbb{F}_p)$;

$\rightsquigarrow$ Il faut savoir trouver un point $G \in E(\mathbb{F}_p)$;

$\rightsquigarrow$ Il faut que E ne soit pas "faible"...

Courbes elliptiques sur $\mathbb{Z}/p\mathbb{Z}$

▷ On prend $k = \mathbb{F}_7$. Soit $E : y^2 = x^3 + x + 4$. On a

$$E(\mathbb{F}_7) =< (0,2) >= \{\mathcal{O}, (0,2), 2 \cdot (0,2), \cdots, 9 \cdot (0,2)\}$$

On a donc $(2,0) = \ell \cdot (0,2)$ pour un certain ℓ. Que vaut ℓ ?.

$\rightsquigarrow$ C'est le problème du logarithme discret dans $E(\mathbb{F}_7)$! (ici $\ell = 5$).

$\rightsquigarrow$ $E(\mathbb{F}_7)$ est cyclique (c'est fréquent).

Si E est définie sur $\mathbb{F}_p$ avec p grand, le problème du log discret semble compliqué : utile pour la cryptographie.

Il faut pouvoir le mettre en place :

$\rightsquigarrow$ Trouver E et p tels que $\sharp E(\mathbb{F}_p)$ est (presque) premier ;

$\rightsquigarrow$ Il faut savoir calculer $\sharp E(\mathbb{F}_p)$;

$\rightsquigarrow$ Il faut savoir trouver un point $G \in E(\mathbb{F}_p)$;

$\rightsquigarrow$ Il faut que E ne soit pas "faible"...

Courbes elliptiques sur $\mathbb{Z}/p\mathbb{Z}$

▷ On prend $k = \mathbb{F}_7$. Soit $E : y^2 = x^3 + x + 4$. On a

$$E(\mathbb{F}_7) =< (0,2) >= \{\mathcal{O}, (0,2), 2 \cdot (0,2), \cdots, 9 \cdot (0,2)\}$$

On a donc $(2,0) = \ell \cdot (0,2)$ pour un certain ℓ. Que vaut ℓ ?.

$\rightsquigarrow$ C'est le problème du logarithme discret dans $E(\mathbb{F}_7)$! (ici $\ell = 5$).

$\rightsquigarrow$ $E(\mathbb{F}_7)$ est cyclique (c'est fréquent).

Si E est définie sur $\mathbb{F}_p$ avec p grand, le problème du log discret semble compliqué : utile pour la cryptographie.

Il faut pouvoir le mettre en place :

$\rightsquigarrow$ Trouver E et p tels que $\sharp E(\mathbb{F}_p)$ est (presque) premier ;

$\rightsquigarrow$ Il faut savoir calculer $\sharp E(\mathbb{F}_p)$;

$\rightsquigarrow$ Il faut savoir trouver un point $G \in E(\mathbb{F}_p)$;

$\rightsquigarrow$ Il faut que E ne soit pas "faible"...

Courbes elliptiques sur $\mathbb{Z}/p\mathbb{Z}$

▷ On prend $k = \mathbb{F}_7$. Soit $E : y^2 = x^3 + x + 4$. On a

$$E(\mathbb{F}_7) =< (0,2) >= \{\mathcal{O}, (0,2), 2 \cdot (0,2), \cdots, 9 \cdot (0,2)\}$$

On a donc $(2,0) = \ell \cdot (0,2)$ pour un certain ℓ. Que vaut ℓ ?.

$\rightsquigarrow$ C'est le problème du logarithme discret dans $E(\mathbb{F}_7)$! (ici $\ell = 5$).

$\rightsquigarrow$ $E(\mathbb{F}_7)$ est cyclique (c'est fréquent).

Si E est définie sur $\mathbb{F}_p$ avec p grand, le problème du log discret semble compliqué : utile pour la cryptographie.

Il faut pouvoir le mettre en place :

$\rightsquigarrow$ Trouver E et p tels que $\sharp E(\mathbb{F}_p)$ est (presque) premier ;

$\rightsquigarrow$ Il faut savoir calculer $\sharp E(\mathbb{F}_p)$;

$\rightsquigarrow$ Il faut savoir trouver un point $G \in E(\mathbb{F}_p)$;

$\rightsquigarrow$ Il faut que E ne soit pas "faible"...

Courbes elliptiques sur $\mathbb{Z}/p\mathbb{Z}$

▷ On prend $k = \mathbb{F}_7$. Soit $E : y^2 = x^3 + x + 4$. On a

$$E(\mathbb{F}_7) =< (0,2) >= \{\mathcal{O}, (0,2), 2 \cdot (0,2), \cdots, 9 \cdot (0,2)\}$$

On a donc $(2,0) = \ell \cdot (0,2)$ pour un certain ℓ. Que vaut ℓ ?.

$\rightsquigarrow$ C'est le problème du logarithme discret dans $E(\mathbb{F}_7)$! (ici $\ell = 5$).

$\rightsquigarrow$ $E(\mathbb{F}_7)$ est cyclique (c'est fréquent).

Si E est définie sur $\mathbb{F}_p$ avec p grand, le problème du log discret semble compliqué : utile pour la cryptographie.

Il faut pouvoir le mettre en place :

$\rightsquigarrow$ Trouver E et p tels que $\sharp E(\mathbb{F}_p)$ est (presque) premier ;

$\rightsquigarrow$ Il faut savoir calculer $\sharp E(\mathbb{F}_p)$;

$\rightsquigarrow$ Il faut savoir trouver un point $G \in E(\mathbb{F}_p)$;

$\rightsquigarrow$ Il faut que E ne soit pas "faible"...

Courbes elliptiques sur $\mathbb{Z}/p\mathbb{Z}$

▷ On prend $k = \mathbb{F}_7$. Soit $E : y^2 = x^3 + x + 4$. On a

$$E(\mathbb{F}_7) =< (0,2) >= \{\mathcal{O}, (0,2), 2 \cdot (0,2), \cdots, 9 \cdot (0,2)\}$$

On a donc $(2,0) = \ell \cdot (0,2)$ pour un certain ℓ. Que vaut ℓ ?.

$\rightsquigarrow$ C'est le problème du logarithme discret dans $E(\mathbb{F}_7)$! (ici $\ell = 5$).

$\rightsquigarrow$ $E(\mathbb{F}_7)$ est cyclique (c'est fréquent).

Si E est définie sur $\mathbb{F}_p$ avec p grand, le problème du log discret semble compliqué : utile pour la cryptographie.

Il faut pouvoir le mettre en place :

$\rightsquigarrow$ Trouver E et p tels que $\sharp E(\mathbb{F}_p)$ est (presque) premier ;

$\rightsquigarrow$ Il faut savoir calculer $\sharp E(\mathbb{F}_p)$;

$\rightsquigarrow$ Il faut savoir trouver un point $G \in E(\mathbb{F}_p)$;

$\rightsquigarrow$ Il faut que E ne soit pas "faible"...

Courbes elliptiques sur $\mathbb{Z}/p\mathbb{Z}$

▷ On prend $k = \mathbb{F}_7$. Soit $E : y^2 = x^3 + x + 4$. On a

$$E(\mathbb{F}_7) =< (0,2) >= \{\mathcal{O}, (0,2), 2 \cdot (0,2), \cdots, 9 \cdot (0,2)\}$$

On a donc $(2,0) = \ell \cdot (0,2)$ pour un certain ℓ. Que vaut ℓ ?.

$\rightsquigarrow$ C'est le problème du logarithme discret dans $E(\mathbb{F}_7)$! (ici $\ell = 5$).

$\rightsquigarrow$ $E(\mathbb{F}_7)$ est cyclique (c'est fréquent).

Si E est définie sur $\mathbb{F}_p$ avec p grand, le problème du log discret semble compliqué : utile pour la cryptographie.

Il faut pouvoir le mettre en place :

$\rightsquigarrow$ Trouver E et p tels que $\sharp E(\mathbb{F}_p)$ est (presque) premier ;

$\rightsquigarrow$ Il faut savoir calculer $\sharp E(\mathbb{F}_p)$;

$\rightsquigarrow$ Il faut savoir trouver un point $G \in E(\mathbb{F}_p)$;

$\rightsquigarrow$ Il faut que E ne soit pas "faible"...

Courbes elliptiques sur $\mathbb{Z}/p\mathbb{Z}$

▷ On prend $k = \mathbb{F}_7$. Soit $E : y^2 = x^3 + x + 4$. On a

$$E(\mathbb{F}_7) =< (0,2) >= \{\mathcal{O}, (0,2), 2 \cdot (0,2), \cdots, 9 \cdot (0,2)\}$$

On a donc $(2,0) = \ell \cdot (0,2)$ pour un certain ℓ. Que vaut ℓ ?.

$\rightsquigarrow$ C'est le problème du logarithme discret dans $E(\mathbb{F}_7)$! (ici $\ell = 5$).

$\rightsquigarrow$ $E(\mathbb{F}_7)$ est cyclique (c'est fréquent).

Si E est définie sur $\mathbb{F}_p$ avec p grand, le problème du log discret semble compliqué : utile pour la cryptographie.

Il faut pouvoir le mettre en place :

$\rightsquigarrow$ Trouver E et p tels que $\sharp E(\mathbb{F}_p)$ est (presque) premier ;

$\rightsquigarrow$ Il faut savoir calculer $\sharp E(\mathbb{F}_p)$;

$\rightsquigarrow$ Il faut savoir trouver un point $G \in E(\mathbb{F}_p)$;

$\rightsquigarrow$ Il faut que E ne soit pas "faible"...

Courbes elliptiques sur $\mathbb{Z}/p\mathbb{Z}$

▷ On prend $k = \mathbb{F}_p$, p grand et $E : y^2 = x^3 + ax + b$. On écrit

$$\sharp E(\mathbb{F}_p) = p + 1 - t$$

Le nombre t s'appelle la trace du Frobenius.

Théorème

- On a $|t| \leq 2\sqrt{p}$ (th. Hasse-Weil).
- Le nombre t se calcule en temps polynomial.
- $E(\mathbb{F}_p) \simeq \mathbb{Z}/d\mathbb{Z}$ ou $\mathbb{Z}/2\mathbb{Z} \times \mathbb{Z}/2d\mathbb{Z}$ où $d = \frac{p+1-t}{2}$.

Conséquences :

$\rightsquigarrow \sharp E(\mathbb{F}_p) \approx p$.

$\rightsquigarrow \sharp E(\mathbb{F}_p)$ se calcule facilement.

$\rightsquigarrow E(\mathbb{F}_p)$ est cyclique ou presque.

Courbes elliptiques sur $\mathbb{Z}/p\mathbb{Z}$

▷ On prend $k = \mathbb{F}_p$, p grand et $E : y^2 = x^3 + ax + b$. On écrit

$$\sharp E(\mathbb{F}_p) = p + 1 - t$$

Le nombre t s'appelle la trace du Frobenius.

Théorème

- On a $|t| \leq 2\sqrt{p}$ (th. Hasse-Weil).
- Le nombre t se calcule en temps polynomial.
- $E(\mathbb{F}_p) \simeq \mathbb{Z}/d_2\mathbb{Z}$ ou $\simeq \mathbb{Z}/d_1\mathbb{Z} \oplus \mathbb{Z}/d_2\mathbb{Z}$ avec $d_1 \mid d_2$ et $d_1 \mid p - 1$.

Conséquences :

$\rightsquigarrow \sharp E(\mathbb{F}_p) \approx p$.

$\rightsquigarrow \sharp E(\mathbb{F}_p)$ se calcule facilement.

$\rightsquigarrow E(\mathbb{F}_p)$ est cyclique ou presque.

Courbes elliptiques sur $\mathbb{Z}/p\mathbb{Z}$

▷ On prend $k = \mathbb{F}_p$, p grand et $E : y^2 = x^3 + ax + b$. On écrit

$$\sharp E(\mathbb{F}_p) = p + 1 - t$$

Le nombre t s'appelle la trace du Frobenius.

Théorème

- On a $|t| \leq 2\sqrt{p}$ (th. Hasse-Weil).
- Le nombre t se calcule en temps polynomial.
- $E(\mathbb{F}_p) \simeq \mathbb{Z}/d_2\mathbb{Z}$ ou $\simeq \mathbb{Z}/d_1\mathbb{Z} \oplus \mathbb{Z}/d_2\mathbb{Z}$ avec $d_1 \mid d_2$ et $d_1 \mid p - 1$.

Conséquences :

$\rightsquigarrow \sharp E(\mathbb{F}_p) \approx p$.

$\rightsquigarrow \sharp E(\mathbb{F}_p)$ se calcule facilement.

$\rightsquigarrow E(\mathbb{F}_p)$ est cyclique ou presque.

Courbes elliptiques sur $\mathbb{Z}/p\mathbb{Z}$

▷ On prend $k = \mathbb{F}_p$, p grand et $E : y^2 = x^3 + ax + b$. On écrit

$$\sharp E(\mathbb{F}_p) = p + 1 - t$$

Le nombre t s'appelle la trace du Frobenius.

Théorème

- On a $|t| \leq 2\sqrt{p}$ (th. Hasse-Weil).
- Le nombre t se calcule en temps polynomial.
- $E(\mathbb{F}_p) \simeq \mathbb{Z}/d_2\mathbb{Z}$ ou $\simeq \mathbb{Z}/d_1\mathbb{Z} \oplus \mathbb{Z}/d_2\mathbb{Z}$ avec $d_1 \mid d_2$ et $d_1 \mid p - 1$.

Conséquences :

$\rightsquigarrow \sharp E(\mathbb{F}_p) \approx p$.

$\rightsquigarrow \sharp E(\mathbb{F}_p)$ se calcule facilement.

$\rightsquigarrow E(\mathbb{F}_p)$ est cyclique ou presque.

Courbes elliptiques sur $\mathbb{Z}/p\mathbb{Z}$

▷ On prend $k = \mathbb{F}_p$, p grand et $E : y^2 = x^3 + ax + b$. On écrit

$$\sharp E(\mathbb{F}_p) = p + 1 - t$$

Le nombre t s'appelle la trace du Frobenius.

Théorème

- On a $|t| \leq 2\sqrt{p}$ (th. Hasse-Weil).
- Le nombre t se calcule en temps polynomial.
- $E(\mathbb{F}_p) \simeq \mathbb{Z}/d_2\mathbb{Z}$ ou $\simeq \mathbb{Z}/d_1\mathbb{Z} \oplus \mathbb{Z}/d_2\mathbb{Z}$ avec $d_1 \mid d_2$ et $d_1 \mid p - 1$.

Conséquences :

$\rightsquigarrow \sharp E(\mathbb{F}_p) \approx p$.

$\rightsquigarrow \sharp E(\mathbb{F}_p)$ se calcule facilement.

$\rightsquigarrow E(\mathbb{F}_p)$ est cyclique ou presque.

Courbes elliptiques sur $\mathbb{Z}/p\mathbb{Z}$

▷ On prend $k = \mathbb{F}_p$, p grand et $E : y^2 = x^3 + ax + b$. On écrit

$$\sharp E(\mathbb{F}_p) = p + 1 - t$$

Le nombre t s'appelle la trace du Frobenius.

Théorème

- On a $|t| \leq 2\sqrt{p}$ (th. Hasse-Weil).
- Le nombre t se calcule en temps polynomial.
- $E(\mathbb{F}_p) \simeq \mathbb{Z}/d_2\mathbb{Z}$ ou $\simeq \mathbb{Z}/d_1\mathbb{Z} \oplus \mathbb{Z}/d_2\mathbb{Z}$ avec $d_1 \mid d_2$ et $d_1 \mid p - 1$.

Conséquences :

$\rightsquigarrow \sharp E(\mathbb{F}_p) \approx p$.

$\rightsquigarrow \sharp E(\mathbb{F}_p)$ se calcule facilement.

$\rightsquigarrow E(\mathbb{F}_p)$ est cyclique ou presque.

Courbes elliptiques sur $\mathbb{Z}/p\mathbb{Z}$

▷ On prend $k = \mathbb{F}_p$, p grand et $E : y^2 = x^3 + ax + b$. On écrit

$$\sharp E(\mathbb{F}_p) = p + 1 - t$$

Le nombre t s'appelle la trace du Frobenius.

Théorème

- On a $|t| \leq 2\sqrt{p}$ (th. Hasse-Weil).
- Le nombre t se calcule en temps polynomial.
- $E(\mathbb{F}_p) \simeq \mathbb{Z}/d_2\mathbb{Z}$ ou $\simeq \mathbb{Z}/d_1\mathbb{Z} \oplus \mathbb{Z}/d_2\mathbb{Z}$ avec $d_1 \mid d_2$ et $d_1 \mid p - 1$.

Conséquences :

$\rightsquigarrow \sharp E(\mathbb{F}_p) \approx p$.

$\rightsquigarrow \sharp E(\mathbb{F}_p)$ se calcule facilement.

$\rightsquigarrow E(\mathbb{F}_p)$ est cyclique ou presque.

Courbes elliptiques sur $\mathbb{Z}/p\mathbb{Z}$

▷ On prend $k = \mathbb{F}_p$, p grand et $E : y^2 = x^3 + ax + b$. On écrit

$$\sharp E(\mathbb{F}_p) = p + 1 - t$$

Le nombre t s'appelle la trace du Frobenius.

Théorème

- On a $|t| \leq 2\sqrt{p}$ (th. Hasse-Weil).
- Le nombre t se calcule en temps polynomial.
- $E(\mathbb{F}_p) \simeq \mathbb{Z}/d_2\mathbb{Z}$ ou $\simeq \mathbb{Z}/d_1\mathbb{Z} \oplus \mathbb{Z}/d_2\mathbb{Z}$ avec $d_1 \mid d_2$ et $d_1 \mid p - 1$.

Conséquences :

$\rightsquigarrow \sharp E(\mathbb{F}_p) \approx p$.

$\rightsquigarrow \sharp E(\mathbb{F}_p)$ se calcule facilement.

$\rightsquigarrow E(\mathbb{F}_p)$ est cyclique ou presque.

Courbes elliptiques sur $\mathbb{Z}/p\mathbb{Z}$

▷ On prend $k = \mathbb{F}_p$, p grand et $E : y^2 = x^3 + ax + b$. On écrit

$$\sharp E(\mathbb{F}_p) = p + 1 - t$$

Le nombre t s'appelle la trace du Frobenius.

Théorème

- On a $|t| \leq 2\sqrt{p}$ (th. Hasse-Weil).
- Le nombre t se calcule en temps polynomial.
- $E(\mathbb{F}_p) \simeq \mathbb{Z}/d_2\mathbb{Z}$ ou $\simeq \mathbb{Z}/d_1\mathbb{Z} \oplus \mathbb{Z}/d_2\mathbb{Z}$ avec $d_1 \mid d_2$ et $d_1 \mid p - 1$.

Conséquences :

$\rightsquigarrow \sharp E(\mathbb{F}_p) \approx p$.

$\rightsquigarrow \sharp E(\mathbb{F}_p)$ se calcule facilement.

$\rightsquigarrow E(\mathbb{F}_p)$ est cyclique ou presque.

Applications cryptographiques

Étape 1 : On choisit p un grand nombre premier.

Étape 2 : On choisit une courbe elliptique E définie sur $\mathbb{F}_p$.

Étape 3 : On calcule t et $\sharp E(\mathbb{F}_p)$.
Si $\sharp E(\mathbb{F}_p)$ n'est pas de la forme $c \cdot q$ avec $c = 1, 2$ ou 3 et q premier alors on retourne à l'étape 1 ou 2.

Étape 4 : On cherche un point $G \in E(\mathbb{F}_p)$ d'ordre q.

Exemple :

▷ $p = 2^{192} - 2^{32} - 2^{12} - 2^8 - 2^7 - 2^6 - 2^3 - 1$.

▷ $E : y^2 = x^3 + 3$.

▷ $t = 146402144145231529258894028971$.

▷ $p + 1 - t$ est un nombre premier $\rightsquigarrow E(\mathbb{F}_p)$ est cyclique d'ordre premier.

▷ $G = (1, 2)$ est générateur de $E(\mathbb{F}_p)$.

L'exemple est la courbe $\mathrm{secp192k1}$ de Certicom. La sécurité est annoncée équivalente à RSA/DSA 1536 (i.e. qu'un problème du log discret posé sur un $\mathbb{F}_p^\times$ avec $p \approx 2^{1536}$).

Applications cryptographiques

Étape 1 : On choisit p un grand nombre premier.

Étape 2 : On choisit une courbe elliptique E définie sur $\mathbb{F}_p$.

Étape 3 : On calcule t et $\sharp E(\mathbb{F}_p)$.
Si $\sharp E(\mathbb{F}_p)$ n'est pas de la forme $c \cdot q$ avec $c = 1, 2$ ou 3 et q premier alors on retourne à l'étape 1 ou 2.

Étape 4 : On cherche un point $G \in E(\mathbb{F}_p)$ d'ordre q.

Exemple :

▷ $p = 2^{192} - 2^{32} - 2^{12} - 2^8 - 2^7 - 2^6 - 2^3 - 1$.

▷ $E : y^2 = x^3 + 3$.

▷ $t = 146402144145231529258894028971$.

▷ $p + 1 - t$ est un nombre premier $\rightsquigarrow E(\mathbb{F}_p)$ est cyclique d'ordre premier.

▷ $G = (1, 2)$ est générateur de $E(\mathbb{F}_p)$.

L'exemple est la courbe $\mathrm{secp192k1}$ de Certicom. La sécurité est annoncée équivalente à RSA/DSA 1536 (i.e. qu'un problème du log discret posé sur un $\mathbb{F}_p^\times$ avec $p \approx 2^{1536}$).

Applications cryptographiques

Étape 1 : On choisit p un grand nombre premier.

Étape 2 : On choisit une courbe elliptique E définie sur $\mathbb{F}_p$.

Étape 3 : On calcule t et $\sharp E(\mathbb{F}_p)$.
Si $\sharp E(\mathbb{F}_p)$ n'est pas de la forme $c \cdot q$ avec $c = 1, 2$ ou 3 et q premier alors on retourne à l'étape 1 ou 2.

Étape 4 : On cherche un point $G \in E(\mathbb{F}_p)$ d'ordre q.

Exemple :

- $p = 2^{192} - 2^{32} - 2^{12} - 2^8 - 2^7 - 2^6 - 2^3 - 1$.
- $E : y^2 = x^3 + 3$.
- $t = 146402144145231529258894028971$.
- $p + 1 - t$ est un nombre premier $\rightsquigarrow E(\mathbb{F}_p)$ est cyclique d'ordre premier.
- $G = (1, 2)$ est générateur de $E(\mathbb{F}_p)$.

L'exemple est la courbe $\mathtt{secp192k1}$ de Certicom. La sécurité est annoncée équivalente à RSA/DSA 1536 (i.e. qu'un problème du log discret posé sur un $\mathbb{F}_p^\times$ avec $p \approx 2^{1536}$).

Applications cryptographiques

Étape 1 : On choisit p un grand nombre premier.

Étape 2 : On choisit une courbe elliptique E définie sur $\mathbb{F}_p$.

Étape 3 : On calcule t et $\sharp E(\mathbb{F}_p)$.
Si $\sharp E(\mathbb{F}_p)$ n'est pas de la forme $c \cdot q$ avec $c = 1, 2$ ou 3 et q premier alors on retourne à l'étape 1 ou 2.

Étape 4 : On cherche un point $G \in E(\mathbb{F}_p)$ d'ordre q.

Exemple :

- $p = 2^{192} - 2^{32} - 2^{12} - 2^8 - 2^7 - 2^6 - 2^3 - 1$.
- $E : y^2 = x^3 + 3$.
- $t = 146402144145231529258894028971$.
- $p + 1 - t$ est un nombre premier $\rightsquigarrow E(\mathbb{F}_p)$ est cyclique d'ordre premier.
- $G = (1, 2)$ est générateur de $E(\mathbb{F}_p)$.

L'exemple est la courbe $\mathtt{secp192k1}$ de Certicom. La sécurité est annoncée équivalente à RSA/DSA 1536 (i.e. qu'un problème du log discret posé sur un $\mathbb{F}_p^\times$ avec $p \approx 2^{1536}$).

Applications cryptographiques

Étape 1 : On choisit p un grand nombre premier.

Étape 2 : On choisit une courbe elliptique E définie sur $\mathbb{F}_p$.

Étape 3 : On calcule t et $\sharp E(\mathbb{F}_p)$.
Si $\sharp E(\mathbb{F}_p)$ n'est pas de la forme $c \cdot q$ avec $c = 1, 2$ ou 3 et q premier alors on retourne à l'étape 1 ou 2.

Étape 4 : On cherche un point $G \in E(\mathbb{F}_p)$ d'ordre q.

Exemple :

- $p = 2^{192} - 2^{32} - 2^{12} - 2^8 - 2^7 - 2^6 - 2^3 - 1$.
- $E : y^2 = x^3 + 3$.
- $t = 146402144145231529258894028971$.
- $p + 1 - t$ est un nombre premier $\rightsquigarrow E(\mathbb{F}_p)$ est cyclique d'ordre premier.
- $G = (1, 2)$ est générateur de $E(\mathbb{F}_p)$.

L'exemple est la courbe $\mathtt{secp192k1}$ de Certicom. La sécurité est annoncée équivalente à RSA/DSA 1536 (i.e. qu'un problème du log discret posé sur un $\mathbb{F}_p^\times$ avec $p \approx 2^{1536}$).

Applications cryptographiques

Étape 1 : On choisit p un grand nombre premier.

Étape 2 : On choisit une courbe elliptique E définie sur $\mathbb{F}_p$.

Étape 3 : On calcule t et $\sharp E(\mathbb{F}_p)$.
Si $\sharp E(\mathbb{F}_p)$ n'est pas de la forme $c \cdot q$ avec $c = 1, 2$ ou 3 et q premier alors on retourne à l'étape 1 ou 2.

Étape 4 : On cherche un point $G \in E(\mathbb{F}_p)$ d'ordre q.

Exemple :

▷ $p = 2^{192} - 2^{32} - 2^{12} - 2^8 - 2^7 - 2^6 - 2^3 - 1$.

▷ $E : y^2 = x^3 + 3$.

▷ $t = 146402144145231529258894028971$.

▷ $p + 1 - t$ est un nombre premier $\rightsquigarrow E(\mathbb{F}_p)$ est cyclique d'ordre premier.

▷ $G = (1, 2)$ est générateur de $E(\mathbb{F}_p)$.

L'exemple est la courbe $\mathtt{secp192k1}$ de Certicom. La sécurité est annoncée équivalente à RSA/DSA 1536 (i.e. qu'un problème du log discret posé sur un $\mathbb{F}_p^\times$ avec $p \approx 2^{1536}$).

Applications cryptographiques

Étape 1 : On choisit p un grand nombre premier.

Étape 2 : On choisit une courbe elliptique E définie sur $\mathbb{F}_p$.

Étape 3 : On calcule t et $\sharp E(\mathbb{F}_p)$.
Si $\sharp E(\mathbb{F}_p)$ n'est pas de la forme $c \cdot q$ avec $c = 1, 2$ ou 3 et q premier alors on retourne à l'étape 1 ou 2.

Étape 4 : On cherche un point $G \in E(\mathbb{F}_p)$ d'ordre q.

Exemple :

▷ $p = 2^{192} - 2^{32} - 2^{12} - 2^8 - 2^7 - 2^6 - 2^3 - 1$.

▷ $E\colon y^2 = x^3 + 3$.

▷ $t = 146402144145231529258894028971$.

▷ $p + 1 - t$ est un nombre premier $\rightsquigarrow E(\mathbb{F}_p)$ est cyclique d'ordre premier.

▷ $G = (1, 2)$ est générateur de $E(\mathbb{F}_p)$.

L'exemple est la courbe $\mathtt{secp192k1}$ de Certicom. La sécurité est annoncée équivalente à RSA/DSA 1536 (i.e. qu'un problème du log discret posé sur un $\mathbb{F}_p^\times$ avec $p \approx 2^{1536}$).

Applications cryptographiques

Étape 1 : On choisit p un grand nombre premier.

Étape 2 : On choisit une courbe elliptique E définie sur $\mathbb{F}_p$.

Étape 3 : On calcule t et $\sharp E(\mathbb{F}_p)$.
Si $\sharp E(\mathbb{F}_p)$ n'est pas de la forme $c \cdot q$ avec $c = 1, 2$ ou 3 et q premier alors on retourne à l'étape 1 ou 2.

Étape 4 : On cherche un point $G \in E(\mathbb{F}_p)$ d'ordre q.

Exemple :

$\triangleright$ $p = 2^{192} - 2^{32} - 2^{12} - 2^8 - 2^7 - 2^6 - 2^3 - 1$.

$\triangleright$ $E\colon y^2 = x^3 + 3$.

$\triangleright$ $t = 146402144145231529258894028971$.

$\triangleright$ $p + 1 - t$ est un nombre premier $\rightsquigarrow$ $E(\mathbb{F}_p)$ est cyclique d'ordre premier.

$\triangleright$ $G = (1, 2)$ est générateur de $E(\mathbb{F}_p)$.

L'exemple est la courbe `secp192k1` de Certicom. La sécurité est annoncée équivalente à RSA/DSA 1536 (i.e. qu'un problème du log discret posé sur un $\mathbb{F}_p^\times$ avec $p \approx 2^{1536}$).

Applications cryptographiques

Étape 1 : On choisit p un grand nombre premier.

Étape 2 : On choisit une courbe elliptique E définie sur $\mathbb{F}_p$.

Étape 3 : On calcule t et $\sharp E(\mathbb{F}_p)$.
Si $\sharp E(\mathbb{F}_p)$ n'est pas de la forme $c \cdot q$ avec $c = 1, 2$ ou 3 et q premier alors on retourne à l'étape 1 ou 2.

Étape 4 : On cherche un point $G \in E(\mathbb{F}_p)$ d'ordre q.

Exemple :

▷ $p = 2^{192} - 2^{32} - 2^{12} - 2^8 - 2^7 - 2^6 - 2^3 - 1$.

▷ $E\colon y^2 = x^3 + 3$.

▷ $t = 146402144145231529258894028971$.

▷ $p + 1 - t$ est un nombre premier $\rightsquigarrow E(\mathbb{F}_p)$ est cyclique d'ordre premier.

▷ $G = (1, 2)$ est générateur de $E(\mathbb{F}_p)$.

L'exemple est la courbe `secp192k1` de Certicom. La sécurité est annoncée équivalente à RSA/DSA 1536 (i.e. qu'un problème du log discret posé sur un $\mathbb{F}_p^\times$ avec $p \approx 2^{1536}$).

Applications cryptographiques

Étape 1 : On choisit p un grand nombre premier.

Étape 2 : On choisit une courbe elliptique E définie sur $\mathbb{F}_p$.

Étape 3 : On calcule t et $\sharp E(\mathbb{F}_p)$.
Si $\sharp E(\mathbb{F}_p)$ n'est pas de la forme $c \cdot q$ avec $c = 1, 2$ ou 3 et q premier alors on retourne à l'étape 1 ou 2.

Étape 4 : On cherche un point $G \in E(\mathbb{F}_p)$ d'ordre q.

Exemple :

$\triangleright$ $p = 2^{192} - 2^{32} - 2^{12} - 2^8 - 2^7 - 2^6 - 2^3 - 1$.

$\triangleright$ $E\colon y^2 = x^3 + 3$.

$\triangleright$ $t = 1464021441452315292588940289 71$.

$\triangleright$ $p + 1 - t$ est un nombre premier $\rightsquigarrow$ $E(\mathbb{F}_p)$ est cyclique d'ordre premier.

$\triangleright$ $G = (1, 2)$ est générateur de $E(\mathbb{F}_p)$.

L'exemple est la courbe `secp192k1` de Certicom. La sécurité est annoncée équivalente à RSA/DSA 1536 (i.e. qu'un problème du log discret posé sur un $\mathbb{F}_p^\times$ avec $p \approx 2^{1536}$).

Applications cryptographiques

▷ Le système est donné par $(\mathbb{F}_p, E, G, n)$ où E est une courbe elliptique définie sur $\mathbb{F}_p$, $G \in E(\mathbb{F}_p)$ est d'ordre n premier.

Attention :

- Si $t = 1$ alors $\sharp E(\mathbb{F}_p) = p + 1 - t = p$ est premier. ← Attaque de Smart (linéaire) pour le log discret.
- Si $t = 0$ alors le problème du log discret sur $E(\mathbb{F}_p)$ se ramène à un problème du log discret sur $\mathbb{F}_q^\times$ ← attaque sous-exponentielle (attaque MOV).

...

Applications cryptographiques

▷ Le système est donné par $(\mathbb{F}_p, E, G, n)$ où E est une courbe elliptique définie sur $\mathbb{F}_p$, $G \in E(\mathbb{F}_p)$ est d'ordre n premier.

Attention :

- Si $t = 1$ alors $\sharp E(\mathbb{F}_p) = p + 1 - t = p$ est premier. ← Attaque de Smart (linéaire) pour le log discret.
- Si $t = 0$ alors le problème du log discret sur $E(\mathbb{F}_p)$ se ramène à un problème du log discret sur $\mathbb{F}_q^\times$ ← attaque sous-exponentielle (attaque MOV).

...

Applications cryptographiques

▷ Le système est donné par $(\mathbb{F}_p, E, G, n)$ où E est une courbe elliptique définie sur $\mathbb{F}_p$, $G \in E(\mathbb{F}_p)$ est d'ordre n premier.

Attention :

• Si $t = 1$ alors $\sharp E(\mathbb{F}_p) = p + 1 - t = p$ est premier. ← Attaque de Smart (linéaire) pour le log discret.

• Si $t = 0$ alors le problème du log discret sur $E(\mathbb{F}_p)$ se ramène à un problème du log discret sur $\mathbb{F}_q^\times$ ← attaque sous-exponentielle (attaque MOV).

...

Applications cryptographiques

$\triangleright$ Le système est donné par $(\mathbb{F}_p, E, G, n)$ où E est une courbe elliptique définie sur $\mathbb{F}_p$, $G \in E(\mathbb{F}_p)$ est d'ordre n premier.

Attention :

• Si $t = 1$ alors $\sharp E(\mathbb{F}_p) = p + 1 - t = p$ est premier. $\leftarrow$ Attaque de Smart (linéaire) pour le log discret.

• Si $t = 0$ alors le problème du log discret sur $E(\mathbb{F}_p)$ se ramène à un problème du log discret sur $\mathbb{F}_q^\times$ $\leftarrow$ attaque sous-exponentielle (attaque MOV).

...

Conclusion

▷ Les courbes elliptiques permettent de réduire la taille des nombres dans les cryptosystèmes :

⇝ Calculs plus rapides ;

⇝ Besoin de moins de mémoire.

▷ Système pensé dans les années 80 et de plus en plus utilisé depuis une dizaine d'années.

▷ Les courbes elliptiques sont aussi utilisée en cryptographie pour :

⇝ Factoriser ;

⇝ Certifier la primalité des nombres premiers.

▷ Il y a des généralisations...

Conclusion

▷ Les courbes elliptiques permettent de réduire la taille des nombres dans les cryptosystèmes :

- ⇝ Calculs plus rapides ;
- ⇝ Besoin de moins de mémoire.

▷ Système pensé dans les années 80 et de plus en plus utilisé depuis une dizaine d'années.

▷ Les courbes elliptiques sont aussi utilisée en cryptographie pour :

- ⇝ Factoriser ;
- ⇝ Certifier la primalité des nombres premiers.

▷ Il y a des généralisations...

Conclusion

▷ Les courbes elliptiques permettent de réduire la taille des nombres dans les cryptosystèmes :

- ⇝ Calculs plus rapides ;
- ⇝ Besoin de moins de mémoire.

▷ Système pensé dans les années 80 et de plus en plus utilisé depuis une dizaine d'années.

▷ Les courbes elliptiques sont aussi utilisée en cryptographie pour :

- ⇝ Factoriser ;
- ⇝ Certifier la primalité des nombres premiers.

▷ Il y a des généralisations...

Conclusion

▷ Les courbes elliptiques permettent de réduire la taille des nombres dans les cryptosystèmes :

- ⇝ Calculs plus rapides ;
- ⇝ Besoin de moins de mémoire.

▷ Système pensé dans les années 80 et de plus en plus utilisé depuis une dizaine d'années.

▷ Les courbes elliptiques sont aussi utilisée en cryptographie pour :

- ⇝ Factoriser ;
- ⇝ Certifier la primalité des nombres premiers.

▷ Il y a des généralisations...

Conclusion

▷ Les courbes elliptiques permettent de réduire la taille des nombres dans les cryptosystèmes :

- ⇝ Calculs plus rapides ;
- ⇝ Besoin de moins de mémoire.

▷ Système pensé dans les années 80 et de plus en plus utilisé depuis une dizaine d'années.

▷ Les courbes elliptiques sont aussi utilisée en cryptographie pour :

- ⇝ Factoriser ;
- ⇝ Certifier la primalité des nombres premiers.

▷ Il y a des généralisations...

Jouer à pile ou face

▷ Symbole de Legendre : soit p un nombre premier (impair) et $a \in \mathbb{Z}$:

$$\left(\frac{a}{p}\right) = \begin{cases} 1 & \text{si } a \text{est un carré non-nul modulo } p \\ 0 & \text{si } a \text{est nul modulo } p \\ -1 & \text{si } a \text{n'est pas un carré modulo } p \end{cases}.$$

▷ Quelques règles pour le calcul du symbole :

- $\left(\frac{a}{p}\right) = a^{\frac{p-1}{2}} \mod p$; $\left(\frac{ab}{p}\right) = \left(\frac{b}{p}\right)\left(\frac{b}{p}\right)$;
- $\left(\frac{-1}{p}\right) = (-1)^{\frac{p-1}{2}}$; $\left(\frac{2}{p}\right) = (-1)^{\frac{p^2-1}{8}}$;
- si q est nombre premier impair, $\left(\frac{q}{p}\right) = (-1)^{\frac{(p-1)(q-1)}{4}}\left(\frac{p}{q}\right)$.

Exemple :

$$\left(\tfrac{43}{103}\right) = (-1)^{51\times 21}\left(\tfrac{103}{43}\right) = -\left(\tfrac{17}{43}\right) = -(-1)^{16\times 21}\left(\tfrac{43}{17}\right) = -\left(\tfrac{9}{17}\right) = -1$$

⇝ Méthode la plus rapide connue pour calculer le symbole.

▷ Résoudre $x^2 \equiv a \mod p$ est plus délicat (cf. gén. de $(\mathbb{Z}/p\mathbb{Z})^\times$).

Jouer à pile ou face

▷ Symbole de Legendre : soit p un nombre premier (impair) et $a \in \mathbb{Z}$:

$$\left(\frac{a}{p}\right) = \begin{cases} 1 & \text{si } a\text{est un carré non-nul modulo } p \\ 0 & \text{si } a\text{est nul modulo } p \\ -1 & \text{si } a\text{n'est pas un carré modulo } p \end{cases}.$$

▷ Quelques règles pour le calcul du symbole :

- $\left(\frac{a}{p}\right) = a^{\frac{p-1}{2}} \mod p$; $\left(\frac{ab}{p}\right) = \left(\frac{b}{p}\right)\left(\frac{b}{p}\right)$;
- $\left(\frac{-1}{p}\right) = (-1)^{\frac{p-1}{2}}$; $\left(\frac{2}{p}\right) = (-1)^{\frac{p^2-1}{8}}$;
- si q est nombre premier impair, $\left(\frac{q}{p}\right) = (-1)^{\frac{(p-1)(q-1)}{4}}\left(\frac{p}{q}\right)$.

Exemple :

$$\left(\tfrac{43}{103}\right) = (-1)^{51\times 21}\left(\tfrac{103}{43}\right) = -\left(\tfrac{17}{43}\right) = -(-1)^{16\times 21}\left(\tfrac{43}{17}\right) = -\left(\tfrac{9}{17}\right) = -1$$

⇝ Méthode la plus rapide connue pour calculer le symbole.

▷ Résoudre $x^2 \equiv a \mod p$ est plus délicat (cf. gén. de $(\mathbb{Z}/p\mathbb{Z})^\times$).

Jouer à pile ou face

▷ Symbole de Legendre : soit p un nombre premier (impair) et $a \in \mathbb{Z}$:

$$\left(\frac{a}{p}\right) = \begin{cases} 1 & \text{si } a\text{est un carré non-nul modulo } p \\ 0 & \text{si } a\text{est nul modulo } p \\ -1 & \text{si } a\text{n'est pas un carré modulo } p \end{cases}.$$

▷ Quelques règles pour le calcul du symbole :

- $\left(\frac{a}{p}\right) = a^{\frac{p-1}{2}} \mod p$; $\left(\frac{ab}{p}\right) = \left(\frac{b}{p}\right)\left(\frac{b}{p}\right)$;
- $\left(\frac{-1}{p}\right) = (-1)^{\frac{p-1}{2}}$; $\left(\frac{2}{p}\right) = (-1)^{\frac{p^2-1}{8}}$;
- si q est nombre premier impair, $\left(\frac{q}{p}\right) = (-1)^{\frac{(p-1)(q-1)}{4}}\left(\frac{p}{q}\right)$.

Exemple :

$$\left(\tfrac{43}{103}\right) = (-1)^{51\times 21}\left(\tfrac{103}{43}\right) = -\left(\tfrac{17}{43}\right) = -(-1)^{16\times 21}\left(\tfrac{43}{17}\right) = -\left(\tfrac{9}{17}\right) = -1$$

⇝ Méthode la plus rapide connue pour calculer le symbole.

▷ Résoudre $x^2 \equiv a \mod p$ est plus délicat (cf. gén. de $(\mathbb{Z}/p\mathbb{Z})^\times$).

Jouer à pile ou face

▷ Symbole de Legendre : soit p un nombre premier (impair) et $a \in \mathbb{Z}$:

$$\left(\frac{a}{p}\right) = \begin{cases} 1 & \text{si } a\text{est un carré non-nul modulo } p \\ 0 & \text{si } a\text{est nul modulo } p \\ -1 & \text{si } a\text{n'est pas un carré modulo } p \end{cases}.$$

▷ Quelques règles pour le calcul du symbole :

- $\left(\frac{a}{p}\right) = a^{\frac{p-1}{2}} \mod p$; $\left(\frac{ab}{p}\right) = \left(\frac{b}{p}\right)\left(\frac{b}{p}\right)$;
- $\left(\frac{-1}{p}\right) = (-1)^{\frac{p-1}{2}}$; $\left(\frac{2}{p}\right) = (-1)^{\frac{p^2-1}{8}}$;
- si q est nombre premier impair, $\left(\frac{q}{p}\right) = (-1)^{\frac{(p-1)(q-1)}{4}}\left(\frac{p}{q}\right)$.

Exemple :

$$\left(\tfrac{43}{103}\right)=(-1)^{51\times 21}\left(\tfrac{103}{43}\right)=-\left(\tfrac{17}{43}\right)=-(-1)^{16\times 21}\left(\tfrac{43}{17}\right)=-\left(\tfrac{9}{17}\right)=-1$$

⇝ Méthode la plus rapide connue pour calculer le symbole.

▷ Résoudre $x^2 \equiv a \mod p$ est plus délicat (cf. gén. de $(\mathbb{Z}/p\mathbb{Z})^\times$).

Jouer à pile ou face

▷ Symbole de Legendre : soit p un nombre premier (impair) et $a \in \mathbb{Z}$:

$$\left(\frac{a}{p}\right) = \begin{cases} 1 & \text{si } a\text{est un carré non-nul modulo } p \\ 0 & \text{si } a\text{est nul modulo } p \\ -1 & \text{si } a\text{n'est pas un carré modulo } p \end{cases}.$$

▷ Quelques règles pour le calcul du symbole :

- $\left(\frac{a}{p}\right) = a^{\frac{p-1}{2}} \mod p$; $\left(\frac{ab}{p}\right) = \left(\frac{b}{p}\right)\left(\frac{b}{p}\right)$;
- $\left(\frac{-1}{p}\right) = (-1)^{\frac{p-1}{2}}$; $\left(\frac{2}{p}\right) = (-1)^{\frac{p^2-1}{8}}$;
- si q est nombre premier impair, $\left(\frac{q}{p}\right) = (-1)^{\frac{(p-1)(q-1)}{4}}\left(\frac{p}{q}\right)$.

Exemple :

$$\left(\tfrac{43}{103}\right)=(-1)^{51\times 21}\left(\tfrac{103}{43}\right)=-\left(\tfrac{17}{43}\right)=-(-1)^{16\times 21}\left(\tfrac{43}{17}\right)=-\left(\tfrac{9}{17}\right)=-1$$

⇝ Méthode la plus rapide connue pour calculer le symbole.

▷ Résoudre $x^2 \equiv a \mod p$ est plus délicat (cf. gén. de $(\mathbb{Z}/p\mathbb{Z})^\times$).

Jouer à pile ou face

▷ Symbole de Legendre : soit p un nombre premier (impair) et $a \in \mathbb{Z}$:

$$\left(\frac{a}{p}\right) = \begin{cases} 1 & \text{si } a\text{est un carré non-nul modulo } p \\ 0 & \text{si } a\text{est nul modulo } p \\ -1 & \text{si } a\text{n'est pas un carré modulo } p \end{cases} .$$

▷ Quelques règles pour le calcul du symbole :

- $\left(\frac{a}{p}\right) = a^{\frac{p-1}{2}} \mod p$; $\left(\frac{ab}{p}\right) = \left(\frac{b}{p}\right)\left(\frac{b}{p}\right)$;
- $\left(\frac{-1}{p}\right) = (-1)^{\frac{p-1}{2}}$; $\left(\frac{2}{p}\right) = (-1)^{\frac{p^2-1}{8}}$;
- si q est nombre premier impair, $\left(\frac{q}{p}\right) = (-1)^{\frac{(p-1)(q-1)}{4}}\left(\frac{p}{q}\right)$.

Exemple :

$$\left(\tfrac{43}{103}\right)=(-1)^{51\times 21}\left(\tfrac{103}{43}\right)=-\left(\tfrac{17}{43}\right)=-(-1)^{16\times 21}\left(\tfrac{43}{17}\right)=-\left(\tfrac{9}{17}\right)=-1$$

⇝ Méthode la plus rapide connue pour calculer le symbole.

▷ Résoudre $x^2 \equiv a \mod p$ est plus délicat (cf. gén. de $(\mathbb{Z}/p\mathbb{Z})^\times$).

Jouer à pile ou face

▷ On étend le symbole de Legendre.

⇝ Symbole de Kronecker-Jacobi : si $N = \prod_{p_i | N} p_i^{\alpha_i}$:

$$\left(\frac{a}{N}\right) = \prod_{p_i | N} \left(\frac{a}{p_i}\right)^{\alpha_i}$$

▷ Les mêmes règles de calculs sont vraies !

⇝ Permet de calculer $\left(\frac{a}{b}\right)$ sans factoriser b.

▷ Alice et Bob veulent jouer à pile ou face au téléphone.

⇝ Alice choisit 2 nombres premiers, $p < q$, calcule $N = pq$ et choisit un nombre a t.q. $\left(\frac{a}{N}\right) = -1$.

⇝ Alice envoie N et a à Bob.

⇝ Bob vérifie que $\left(\frac{a}{N}\right) = -1$ et mise sur petit ou grand selon

$$\left(\frac{a}{p}\right) = -1 \text{ ou } \left(\frac{a}{q}\right) = -1.$$

Jouer à pile ou face

▷ On étend le symbole de Legendre.

⇝ Symbole de Kronecker-Jacobi : si $N = \prod_{p_i | N} p_i^{\alpha_i}$:

$$\left(\frac{a}{N}\right) = \prod_{p_i | N} \left(\frac{a}{p_i}\right)^{\alpha_i}$$

▷ Les mêmes règles de calculs sont vraies !

⇝ Permet de calculer $\left(\frac{a}{b}\right)$ sans factoriser b.

▷ Alice et Bob veulent jouer à pile ou face au téléphone.

⇝ Alice choisit 2 nombres premiers, $p < q$, calcule $N = pq$ et choisit un nombre a t.q. $\left(\frac{a}{N}\right) = -1$.

⇝ Alice envoie N et a à Bob.

⇝ Bob vérifie que $\left(\frac{a}{N}\right) = -1$ et mise sur petit ou grand selon

$$\left(\frac{a}{p}\right) = -1 \text{ ou } \left(\frac{a}{q}\right) = -1.$$

Jouer à pile ou face

▷ On étend le symbole de Legendre.

⇝ Symbole de Kronecker-Jacobi : si $N = \prod_{p_i | N} p_i^{\alpha_i}$:

$$\left(\frac{a}{N}\right) = \prod_{p_i | N} \left(\frac{a}{p_i}\right)^{\alpha_i}$$

▷ Les mêmes règles de calculs sont vraies !

⇝ Permet de calculer $\left(\frac{a}{b}\right)$ sans factoriser b.

▷ Alice et Bob veulent jouer à pile ou face au téléphone.

⇝ Alice choisit 2 nombres premiers, $p < q$, calcule $N = pq$ et choisit un nombre a t.q. $\left(\frac{a}{N}\right) = -1$.

⇝ Alice envoie N et a à Bob.

⇝ Bob vérifie que $\left(\frac{a}{N}\right) = -1$ et mise sur petit ou grand selon

$$\left(\frac{a}{p}\right) = -1 \text{ ou } \left(\frac{a}{q}\right) = -1.$$

Jouer à pile ou face

▷ On étend le symbole de Legendre.

⇝ Symbole de Kronecker-Jacobi : si $N = \prod_{p_i | N} p_i^{\alpha_i}$:

$$\left(\frac{a}{N}\right) = \prod_{p_i | N} \left(\frac{a}{p_i}\right)^{\alpha_i}$$

▷ Les mêmes règles de calculs sont vraies !

⇝ Permet de calculer $\left(\frac{a}{b}\right)$ sans factoriser b.

▷ Alice et Bob veulent jouer à pile ou face au téléphone.

⇝ Alice choisit 2 nombres premiers, $p < q$, calcule $N = pq$ et choisit un nombre a t.q. $\left(\frac{a}{N}\right) = -1$.

⇝ Alice envoie N et a à Bob.

⇝ Bob vérifie que $\left(\frac{a}{N}\right) = -1$ et mise sur petit ou grand selon

$$\left(\frac{a}{p}\right) = -1 \text{ ou } \left(\frac{a}{q}\right) = -1.$$

Jouer à pile ou face

▷ On étend le symbole de Legendre.

⇝ Symbole de Kronecker-Jacobi : si $N = \prod_{p_i | N} p_i^{\alpha_i}$:

$$\left(\frac{a}{N}\right) = \prod_{p_i | N} \left(\frac{a}{p_i}\right)^{\alpha_i}$$

▷ Les mêmes règles de calculs sont vraies !

⇝ Permet de calculer $\left(\frac{a}{b}\right)$ sans factoriser b.

▷ Alice et Bob veulent jouer à pile ou face au téléphone.

⇝ Alice choisit 2 nombres premiers, $p < q$, calcule $N = pq$ et choisit un nombre a t.q. $\left(\frac{a}{N}\right) = -1$.

⇝ Alice envoie N et a à Bob.

⇝ Bob vérifie que $\left(\frac{a}{N}\right) = -1$ et mise sur petit ou grand selon

$$\left(\frac{a}{p}\right) = -1 \quad \text{ou} \quad \left(\frac{a}{q}\right) = -1.$$

Jouer à pile ou face

▷ On étend le symbole de Legendre.

⇝ Symbole de Kronecker-Jacobi : si $N = \prod_{p_i | N} p_i^{\alpha_i}$:

$$\left(\frac{a}{N}\right) = \prod_{p_i | N} \left(\frac{a}{p_i}\right)^{\alpha_i}$$

▷ Les mêmes règles de calculs sont vraies !

⇝ Permet de calculer $\left(\frac{a}{b}\right)$ sans factoriser b.

▷ Alice et Bob veulent jouer à pile ou face au téléphone.

⇝ Alice choisit 2 nombres premiers, $p < q$, calcule $N = pq$ et choisit un nombre a t.q. $\left(\frac{a}{N}\right) = -1$.

⇝ Alice envoie N et a à Bob.

⇝ Bob vérifie que $\left(\frac{a}{N}\right) = -1$ et mise sur petit ou grand selon

$$\left(\frac{a}{p}\right) = -1 \quad \text{ou} \quad \left(\frac{a}{q}\right) = -1.$$

Jouer à pile ou face

▷ On étend le symbole de Legendre.

⇝ Symbole de Kronecker-Jacobi : si $N = \prod_{p_i | N} p_i^{\alpha_i}$:

$$\left(\frac{a}{N}\right) = \prod_{p_i | N} \left(\frac{a}{p_i}\right)^{\alpha_i}$$

▷ Les mêmes règles de calculs sont vraies !

⇝ Permet de calculer $\left(\frac{a}{b}\right)$ sans factoriser b.

▷ Alice et Bob veulent jouer à pile ou face au téléphone.

⇝ Alice choisit 2 nombres premiers, $p < q$, calcule $N = pq$ et choisit un nombre a t.q. $\left(\frac{a}{N}\right) = -1$.

⇝ Alice envoie N et a à Bob.

⇝ Bob vérifie que $\left(\frac{a}{N}\right) = -1$ et mise sur petit ou grand selon

$$\left(\frac{a}{p}\right) = -1 \text{ ou } \left(\frac{a}{q}\right) = -1.$$

Jouer à pile ou face

▷ On étend le symbole de Legendre.

⇝ Symbole de Kronecker-Jacobi : si $N = \prod_{p_i | N} p_i^{\alpha_i}$:

$$\left(\frac{a}{N}\right) = \prod_{p_i | N} \left(\frac{a}{p_i}\right)^{\alpha_i}$$

▷ Les mêmes règles de calculs sont vraies !

⇝ Permet de calculer $\left(\frac{a}{b}\right)$ sans factoriser b.

▷ Alice et Bob veulent jouer à pile ou face au téléphone.

⇝ Alice choisit 2 nombres premiers, $p < q$, calcule $N = pq$ et choisit un nombre a t.q. $\left(\frac{a}{N}\right) = -1$.

⇝ Alice envoie N et a à Bob.

⇝ Bob vérifie que $\left(\frac{a}{N}\right) = -1$ et mise sur petit ou grand selon

$$\left(\frac{a}{p}\right) = -1 \text{ ou } \left(\frac{a}{q}\right) = -1.$$

Jouer à pile ou face

▷ On étend le symbole de Legendre.

⇝ Symbole de Kronecker-Jacobi : si $N = \prod_{p_i | N} p_i^{\alpha_i}$:

$$\left(\frac{a}{N}\right) = \prod_{p_i | N} \left(\frac{a}{p_i}\right)^{\alpha_i}$$

▷ Les mêmes règles de calculs sont vraies !

⇝ Permet de calculer $\left(\frac{a}{b}\right)$ sans factoriser b.

▷ Alice et Bob veulent jouer à pile ou face au téléphone.

⇝ Alice choisit 2 nombres premiers, $p < q$, calcule $N = pq$ et choisit un nombre a t.q. $\left(\frac{a}{N}\right) = -1$.

⇝ Alice envoie N et a à Bob.

⇝ Bob vérifie que $\left(\frac{a}{N}\right) = -1$ et mise sur petit ou grand selon

$$\left(\frac{a}{p}\right) = -1 \text{ ou } \left(\frac{a}{q}\right) = -1.$$

Jouer à pile ou face

▷ On étend le symbole de Legendre.

⇝ Symbole de Kronecker-Jacobi : si $N = \prod_{p_i | N} p_i^{\alpha_i}$:

$$\left(\frac{a}{N}\right) = \prod_{p_i | N} \left(\frac{a}{p_i}\right)^{\alpha_i}$$

▷ Les mêmes règles de calculs sont vraies !

⇝ Permet de calculer $\left(\frac{a}{b}\right)$ sans factoriser b.

▷ Alice et Bob veulent jouer à pile ou face au téléphone.

⇝ Alice choisit 2 nombres premiers, $p < q$, calcule $N = pq$ et choisit un nombre a t.q. $\left(\frac{a}{N}\right) = -1$.

⇝ Alice envoie N et a à Bob.

⇝ Bob vérifie que $\left(\frac{a}{N}\right) = -1$ et mise sur petit ou grand selon

$$\left(\frac{a}{p}\right) = -1 \quad \text{ou} \quad \left(\frac{a}{q}\right) = -1.$$