Trouver des grands nombres premiers

Cours Turing – Semaine 10

Tester si un nombre est premier: 1^{e} méthode

Définition:

$N \geq 2$ est premier si et seulement s'il admet exactement deux diviseurs distincts (1 et lui-même)

2, 3, 5, 7, 11, 13, 17, 19, ---

Tester si un nombre est premier: 1^{e} méthode

Définition:
$N \geq 2$ est premier si et seulement s'il admet exactement deux diviseurs distincts (1 et lui-même)

Algorithme:
Tester tous les diviseurs A compris entre 2 et ~~$N-1$~~ $\sqrt{N}$

$N = P \cdot Q$

Tester si un nombre est premier: 1e méthode

Définition:
$N \geq 2$ est premier si et seulement s'il admet exactement deux diviseurs distincts (1 et lui-même)

Algorithme:
Tester tous les diviseurs A compris entre 2 et ~~$N-1$~~ $\sqrt{N}$

Question:
Combien cela représente-t-il d'opérations si N est un nombre à 100 chiffres?

$N \sim 10^{100}$ $\quad \sqrt{N} \sim 10^{50}$

Recherche de grands nombres premiers

Crible d’Eratosthéne

Prime numbers

2	3	5	7
11	13	17	19
23	29	31	37
41	43	47	53
59	61	67	71
73	79	83	89
97	101	103	107
109	113		

Le théorème des nombres premiers

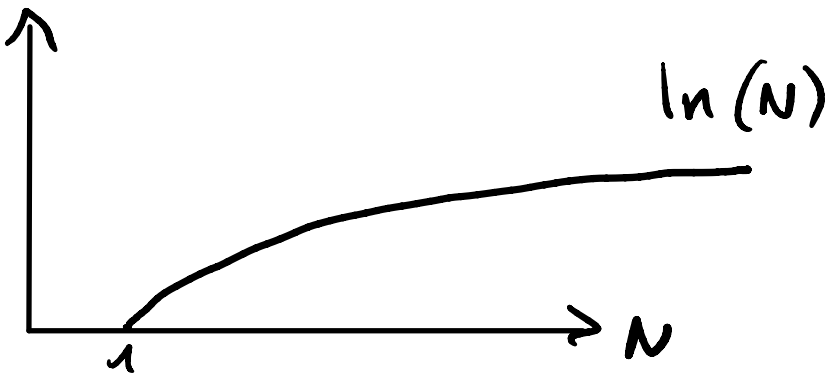
$$\underbrace{\pi(N)}_{\text{= le nombre de nombres premiers} \leq N} \simeq \frac{N}{\ln(N)}$$

$$\left.\begin{matrix}\text{proportion} \\ \text{de nombres} \\ \text{premiers } N\end{matrix}\right\} = \frac{1}{\ln(N)}$$

N est un nombre à n chiffres : $N \sim 10^n$

$\ln(N) \simeq n \cdot 2{,}3$

Si $n = 100$, $\ln(N) \simeq 230$



Arithmétique modulaire

- $A\ (mod\ N) = B$ veut dire que B est le reste de la division de A par N

- B est donc un nombre compris entre 0 et N-1

Opérations en arithmétique modulaire

- Additions: $(41+32) \pmod{11} = 73 \pmod{11} = 7$

$= (41 \pmod{11} + 32 \pmod{11}) \pmod{11}$

$= (8 + 10) \pmod{11} = 18 \pmod{11} = 7$

- Multiplications:

$(41 \cdot 32) \pmod{11} = (1312) \pmod{11} = \dots = 3$

$(41 \pmod{11} \cdot 32 \pmod{11}) \pmod{11} = (8 \cdot 10) \pmod{11}$

$= 80 \pmod{11} = 3$

Le petit théorème de Fermat

Si N est un nombre premier, alors $A^{N-1}(mod\ N) = 1$

pour tout nombre entier A compris entre 1 et N-1

Contraposée

Si A est un nombre entier compris entre 2 et N-1 et $A^{N-1}(mod\ N) \neq 1$,

alors N est *pas* un nombre premier

Le petit théorème de Fermat : extension

Si A est un nombre tiré au hasard entre 2 et N-1 et $A^{N-1}(mod\ N) = 1$,

alors N n'est *pas* un nombre premier avec probabilité <= $\frac{1}{2}$

et donc N *est* un nombre premier avec probabilité >= $\frac{1}{2}$

Si N ≠ premier,

alors $A^{N-1} \pmod N \neq 1$ pour au moins

la moitié des nombres A entre 2 et $N-1$.

<u>Donc</u> : la probabilité que N ≠ premier

et $A^{N-1} \pmod N = 1$ est $\leq 50\%$

↑

tiré au hasard

Le petit théorème de Fermat : extension (suite)

Si $A_1, \ldots, A_k$ sont k nombres tirés au hasard entre 2 et N-1 et

$$A_1{}^{N-1}(mod\ N) = 1, \ldots, A_k{}^{N-1}(mod\ N) = 1$$

alors N n'est *pas* un nombre premier avec probabilité <= $\frac{1}{2^k}$

et donc N *est* un nombre premier avec probabilité >= $1 - \frac{1}{2^k}$

Note: si k=30, alors $\frac{1}{2^k} \cong 0{,}000000001$

Combien d'opérations pour $A^B (mod\ N)$?

Exemple avec A=64, B=6 et N=11:

$$64^6 (mod\ 11) = \text{---} (mod\ 11) = 9$$

$$= (64 (mod\ 11))^6 (mod\ 11) = 9^6 (mod\ 11) = 9^{4+2} (mod 11)$$

$$= ((9^2)^2 \cdot 9^2) (mod\ 11)$$

$$= (((\underbrace{9^2 (mod\ 11)}_{4})^2 (mod\ 11) \cdot (\underbrace{9^2 (mod\ 11)}_{4})) (mod\ 11) = (5 \cdot 4) (mod\ 11)$$

$$= 20 (mod\ 11) = 9$$

Attention!

- Il est vrai que $A^B\,(mod\ N) = (A\ (mod\ N))^B\,(mod\ N)$
- Mais il est *faux* que $A^B\,(mod\ N) = A^{B\ (mod\ N)}\,(mod\ N)$
- Voici un contre-exemple: $64^{17}\,(\text{mod } 11) \neq 64^{6}\,(\text{mod } 11)$

$4 \qquad\qquad 9$

Combien d'opérations pour $A^B \, (mod\ N)$?

pow(A,B,N)

Voyons d'abord comment effectuer A^B efficacement (avec B=43 p.ex.):

$$A^{43} = A^{32+8+2+1} = A^{32} \cdot A^8 \cdot A^2 \cdot A^1$$

$$= (((((A^2)^2)^2)^2)^2) \cdot (((A^2)^2)^2) \cdot A^2 \cdot A^1$$

2·100·10'000

≈ 2'000'000 op.

calculer $A^1, A^2, (A^2)^2, ((A^2)^2)^2, (((A^2)^2)^2)^2, ((((A^2)^2)^2)^2)^2$

Calculer : $B = 43 = 1 + 2 + 0 + 8 + 0 + 32$

- décomposition binaire de B :

$\longrightarrow$ liste = [1, 1, 0, 1, 0, 1]

$\longleftrightarrow$ taille ℓ

- puissances de A :

$\longrightarrow$ liste = [A, $A^2 \% N$, $A^4 \% N$, $A^8 \% N$ ...]

- calculer $A^B \% N = \prod_{i=0}^{\ell-1} \left(B[i] * A^{2^i} \right) \% N$

Combien d'opérations pour $A^B (mod\ N)$?

Voyons d'abord comment effectuer A^B efficacement (avec B=43 p.ex.):

$$B = 43 = 32 + 8 + 2 + 1$$ (= décomposition binaire)

Donc

$$A^B = A^{43} = A^{32} A^8 A^2 A^1$$

On calcule successivement les carrés:

$$A^2,\ \ A^4 = (A^2)^2,\ \ A^8 = (A^4)^2,\ \ A^{16} = (A^8)^2,\ \ A^{32} = (A^{16})^2$$

Puis on effectue la multiplication: $A^{32} A^8 A^2 A^1$

"square-and-multiply"

Combien d'opérations pour $A^B (mod\ N)$?

Pour calculer $A^B (mod\ N)$, on refait *tout* modulo N.

Si A, B, N sont des nombres à 100 chiffres,
combien d'opérations sont-elles nécessaires ?

- La décomposition binaire de B est composée de 100 termes (environ)
- Chaque carré coûte $100^2 = 10'000$ opérations
  (chaque multiplication aussi)
- On effectue au plus 100 carrés, ainsi que 100 multiplications

Combien d'opérations pour $A^B (mod\ N)$?

Pour calculer $A^B (mod\ N)$, on refait *tout* modulo N.

Si A, B, N sont des nombres à 100 chiffres,
combien d'opérations sont-elles nécessaires ?

- La décomposition binaire de B est composée de 100 termes (environ)
- Chaque carré coûte $100^2 = 10'000$ opérations
  (chaque multiplication aussi)
- On effectue au plus 100 carrés, ainsi que 100 multiplications
- Donc au total: $2 \cdot 100 \cdot 10'000 = 2'000'000$ opérations

Décompte final

- Tirer au hasard un nombre à 100 chiffres: nous avons vu la dernière fois que 230 tirages environ sont nécessaires pour tomber sur un nombre premier.

- Pour chaque nombre N, effectuer 30 fois le test $A^{N-1}(mod\ N) = 1$, chaque fois avec un nombre A différent. Par ce qui précède, chaque test coûte environ $2'000'000$ opérations.

- Au total: $230 \cdot 30 \cdot 2'000'000 \cong 10'000'0000'000$ opérations

Factorisation

- Tout nombre entier N s'écrit comme un produit de facteurs premiers.
- Exemple: $\mathrm{N} = 60 = 2 \cdot 2 \cdot 3 \cdot 5$
- (Mal)heureusement, il n'existe pas d'algorithme efficace pour factoriser un grand nombre premier (à 100 chiffres).

Autre problème : factorisation

N = produit de facteurs premiers

A l'heure actuelle, il n'existe pas d'algorithme qui résolve ce problème efficacement !