Test de Primalité de Solovay-Strassen :

Le test de primalité de Solovay-Strassen permet de vérifier qu'un nombre n est bien premier. Il a été développé par Robert Solovay et Volker Strassen en 1977 et publié pour la première fois dans l'article *A FAST MONTE-CARLO TEST FOR PRIMALITY* dans le magazine SIAM Journal of Computing, et corrigé dans l'article *ERRATUM: A FAST MONTE-CARLO TEST FOR PRIMALITY* dans le même magazine en 1978. Ce test a ensuite été supplanté par le test de Miller-Rabin à partir de 1980, mais a été très important dans la preuve de la faisabilité pratique du RSA.

L'algorithme de ce test est le suivant :

Algorithme 8 : `Test de Solovay-Strassen`

Données : Un entier n, dont on souhaite tester la primalité et le nombre k de répétitions

Résultat : `composé` si n est composé, sinon `probablement premier`

pour $i = 1, \ldots, k$ **faire**

  Choisir aléatoirement $a \in \{2, \ldots, n-1\}$

  $r \leftarrow \left(\frac{a}{n}\right)$

  si $r = 0$ *ou* $a^{\frac{n-1}{2}} \neq r \mod n$ **alors**

    renvoyer `composé`

renvoyer `premier`

Cet algorithme repose sur le critère d'Euler et le symbole de Jacobi.

Symbole de Jacobi :

Le symbole de Jacobi a été développé par Charles Gustave Jacob Jacobi en 1837. Il s'agit d'une généralisation du symbole de Legendre.

Le symbole de Legendre est une fonction à deux variables a et p caractérisant les résidus quadratiques entre ces variables (ie si a est ou non un carré modulo p). Cette fonction nécessite que p soit premier pour être calculable : le symbole de Jacobi la généralise aux nombres composés impairs.

Jacobi pose que :

$$\left(\frac{a}{b}\right) = \left(\frac{a}{p_1}\right)^{e_1} \cdot \left(\frac{a}{p_2}\right)^{e_2} \cdot \ldots \cdot \left(\frac{a}{p_n}\right)^{e_n}$$

c'est-à-dire que le symbole de Jacobi $\frac{a}{p}$ est égal au produit du symbole de Jacobi de $\frac{a}{p_k}$, p_k étant un diviseur premier de b.

On a suivi l'algorithme suivant pour calculer le symbole de Jacobi, trouvé dans la source 3 :

Partant de a et b avec b impair positif, on utilise ε la variable de stockage, initialisée à 1 :

- Si b = 1, on renvoie ε.
- Réduction 1 : si a = bq + r est la division euclidienne de a par b, on a simplement à calculer($\frac{r}{b}$)
    - (si r = 0, on termine l'algorithme en renvoyant 0),
    - on remplace donc a par r, de sorte que a < b.
- Réduction 2 : Si $a = 2^k \cdot a'$ avec a′ impair, on multiplie ε par $(-1)^{\frac{b^2-1}{8}}$ à la puissance k puis on remplace a par a′, de sorte que a soit impair.
- On multiplie ε par $(-1)^{\frac{(a-1)(b-1)}{4}}$ (autrement dit ε = 1 sauf si a et b sont congrus à 3 modulo 4), et on échange les variables a et b, autrement dit on calcule $(\frac{b}{a})$. On recommence à la première étape.

Critère d'Euler

Le critère d'Euler permet de savoir si un entier p est premier : si c'est le cas, alors pour tout a ∈ $\mathbb{Z}p/\mathbb{Z}$, on a:

$$(\frac{a}{p}) = a^{\frac{p-1}{2}}$$

Test probabiliste

Le test de Solovay-Strassen consiste donc, pour un nombre p, à générer un entier a aléatoire, à calculer son symbole de Jacobi a/p et à calculer le critère d'Euler. Cette simple opération n'est cependant pas suffisante pour être sûr que p soit premier : le test a $\frac{1}{2}$ chances de générer un nombre a premier avec p même si p n'est pas premier.
La solution pour diminuer cette chance d'erreur est de répéter le test pour k itérations.
La probabilité d'erreur est alors $\frac{1}{2^k}$. Avec un k suffisamment grand, on considère que le test est fiable.

Complexité

Ce test est de complexité $O(k \cdot log^3(p))$ si on utilise l'algorithme d'exponentiation rapide.

Remarques

Ce test permet de différencier les nombres de Carmichael des nombres premiers, contrairement au test de Fermat.

La probabilité d'erreur étant de $\frac{1}{2^k}$, ce test a été remplacé par le test de Miller-Rabin dont la probabilité d'erreur est $\frac{1}{4^k}$, et qui nécessite donc deux fois moins d'opérations pour arriver à la même probabilité d'erreur.

Nous avons échoué dans notre implémentation de la fonction symbole_jacobi supposée calculer le symbole de jacobi entre deux entiers a et n. Nous avons rendu notre tentative d'implémentation mais avons utilisé la fonction native mpz_jacobi dans l'implémentation du test de Solovay-Strassen.

Sources :

Test de Solovay-Strassen :

Chapitre 15 - Primalité

https://epubs.siam.org/doi/10.1137/0206006

https://epubs.siam.org/doi/10.1137/0207009

https://www-fourier.ujf-grenoble.fr/~lefourns/contenu/cours/Agreg2018/CourssymboleLegendre.pdf