アルゴリズム論特講 （塩田）

2006 年 7 月 13 日の課題

課題

(1) 以下のファイルをダウンロードせよ：

- 関数定義部 crypto060706.py
  （前回のものに、ヤコビ記号に関するコメントを追加した。）
- ヤコビ記号 $\left(\frac{a}{n}\right)$ の計算を、相互法則による方法、素因数分解＋オイラーの規準による方法の２通りで比較するプログラム jacobi.py

(2) 法のビット数を取りかえて jacobi.py を実行し、２通りの方法の計算量の違いを実感せよ。

提出期限：7 月 27 日 （ 512 号室ポストまで ）

実行例

```
法のビット数を指定してください（ 0 で終了 ）: 50

##### 法が素数の場合 #####

  素数生成中 ...
  p = 907310303311043 （ 50 bits の素数）
  a = 433085787969859
  平方剰余記号 jacobi(a,p) を計算します。

  1. 相互法則を用いた計算
     jacobi(a,p) = 1
     計算時間 = 0.0

  2. オイラーの規準を用いた計算
     jacobi(a,p) = 1
     計算時間 = 0.00100016593933

##### 法が一般の奇数の場合 ####

  n = 565563366267739 （ 50 bits の奇数）
  a = 263647939985897
  ヤコビ記号 jacobi(a,n) を計算します。

  1. 相互法則を用いた計算
     jacobi(a,n) = -1
     計算時間 = 0.0

  2. 素因数分解＋オイラーの規準を用いた計算
     jacobi(a,n) = -1
     計算時間 = 6.00099992752
```

ヤコビ記号

3以上の奇数 n と整数 a を変数とする次の関数 $\left(\frac{a}{n}\right)$ をヤコビ記号と言う:

$$\left(\frac{a}{n}\right) = \left(\frac{a}{p_1}\right)^{e_1} \times \left(\frac{a}{p_2}\right)^{e_2} \times \cdots \times \left(\frac{a}{p_s}\right)^{e_s}$$

ただし n の素因数分解を

$$n = p_1^{e_1} \times p_2^{e_2} \times \cdots \times p_s^{e_s}$$

とし、右辺は平方剰余記号 $\left(\frac{a}{p_i}\right)$ たちの積として定める。

ヤコビ記号は平方剰余記号の拡張

n が奇素数ならば $\left(\frac{a}{n}\right)$ は平方剰余記号と同じものである。

ヤコビ記号の定義どおりの計算例

$$\left(\frac{5}{99}\right) = \left(\frac{5}{3^2 \times 11}\right) = \left(\frac{5}{3}\right)^2 \times \left(\frac{5}{11}\right)^1 = (-1)^2 \times (1)^1 = 1$$

注

この方法は n の素因数分解の計算量が大きくて使いものにならない。ところが次の定理の「相互法則」のおかげで $\log^3$ オーダーでヤコビ記号を計算することができる。

定理 (1) $\gcd(a,n) \neq 1 \implies \left(\frac{a}{n}\right) = 0$

特に $\left(\frac{0}{n}\right) = 0$

(2) $\gcd(a,n) = 1 \implies \left(\frac{a^2}{n}\right) = 1$

特に $\left(\frac{1}{n}\right) = 1$

(3) $a \equiv b \bmod n \implies \left(\frac{a}{n}\right) = \left(\frac{b}{n}\right)$

(4) $\left(\frac{ab}{n}\right) = \left(\frac{a}{n}\right)\left(\frac{b}{n}\right)$

(5)（第1補充法則）

$$\left(\frac{-1}{n}\right) = \begin{cases} 1 & n \equiv 1 \bmod 4 \text{ のとき} \\ -1 & n \equiv 3 \bmod 4 \text{ のとき} \end{cases}$$

(6)（第2補充法則）

$$\left(\frac{2}{n}\right) = \begin{cases} 1 & n \equiv 1 \text{ または } 7 \bmod 8 \text{ のとき} \\ -1 & n \equiv 3 \text{ または } 5 \bmod 8 \text{ のとき} \end{cases}$$

(7)（相互法則）m，n が共に 3 以上の奇数のとき、

$$\left(\frac{m}{n}\right) = \begin{cases} -\left(\dfrac{n}{m}\right) & m \equiv n \equiv 3 \bmod 4 \text{ のとき} \\ \left(\dfrac{n}{m}\right) & \text{それ以外のとき} \end{cases}$$

相互法則を用いたヤコビ記号の計算例（数字の小さくなり方に注目）

$$\begin{aligned}
\left(\frac{159}{551}\right) &= -\left(\frac{551}{159}\right) && \text{相互法則} \\
&= -\left(\frac{74}{159}\right) && (3) \\
&= -\left(\frac{2}{159}\right)\left(\frac{37}{159}\right) && (4) \text{ で } a=2 \text{ の場合} \\
&= -(1)\left(\frac{37}{159}\right) && \text{第 2 補充法則} \\
&= -\left(\frac{159}{37}\right) && \text{相互法則} \\
&= -\left(\frac{11}{37}\right) && (3) \\
&= -\left(\frac{37}{11}\right) && \text{相互法則} \\
&= -\left(\frac{4}{11}\right) && (3) \\
&= -\left(\frac{2}{11}\right)\left(\frac{2}{11}\right)\left(\frac{1}{11}\right) && (4) \text{ で } a=2 \text{ の場合} \\
&= -(-1)(-1)(1) && \text{第 2 補充法則} \\
&= -1
\end{aligned}$$

ヤコビ記号 $\left(\dfrac{a}{n}\right)$ を求めるアルゴリズム

1° $a < 0$ なら「(4) で $a = -1$ の場合」と第 1 補充法則を用いる。

2° $a = 0$ なら 0 を出力し終了。

3° a が偶数なら 「(4) で $a = 2$ の場合」と第 2 補充法則を用いて再び 3° へ。

4° $a = 1$ なら終了。

5° $a \geqq 3$ なら相互法則と (3) を用いて 2° へ。

平方剰余問題

奇数　$n \geqq 3$ と、n と互いに素な a が与えられたとき、

(1) a が $\bmod n$ の平方剰余である必要十分条件は

$$n \text{ の全ての素因数 } p \text{ について } \left(\frac{a}{p}\right) = 1$$

(2) このとき a の $\bmod n$ での平方根 x は次のアルゴリズムで計算できる：

1° n を素因数分解する：

$$n = p_1^{e_1} \times p_2^{e_2} \times \cdots \times p_s^{e_s}$$

2° 先週のアルゴリズムで $\alpha_i^2 \equiv a \bmod p_i^{e_i}$ を満たす α_i を求める。

3° 中国剰余アルゴリズムを用いて連立合同式

$$x \equiv \alpha_i \bmod p_i^{e_i} \qquad (i = 1, 2, \cdots, s)$$

を満たす x を求める。

暗号へのキーポイント

$$\begin{cases} \text{ヤコビ記号の計算} : \text{素因数分解は不要} \Rightarrow \text{高速} \\ \text{平方剰余問題} \quad\ : \text{素因数分解が必要} \Rightarrow \text{計算量膨大} \end{cases}$$

このギャップが暗号に使える。（詳しくは次回。）