Euclidの互除法、RSA暗号、素因数分解の教材化に関する一考察

林　　雄一郎
北海道情報大学

A Study on Development of Teaching Contents about Euclid's Algorithm,RSA Public Cryptosystem and Factoring Integers

Yuuichirou HAYASHI
Hokkaido Information University

平成26年11月

北海道情報大学紀要　第26巻　第１号別刷

〈論　文〉

Euclidの互除法、RSA暗号、素因数分解の教材化に関する一考察

林　　雄一郎

A Study on Development of Teaching Contents about Euclid's Algorithm,RSA Public Cryptosystem and Factoring Integers

Yuuichirou　HAYASHI

要　　旨

高校「数学A」には「整数の性質」の単元に、約数と倍数、Euclidの互除法、整数の活用に関する教材がある。これらの学習により、素因数を用いて公約数や公倍数に関連する整数に関連した事象を論理的に考察し表現することや、Euclidの互除法の仕組みと最大公約数、二元一次不定方程式の解の意味とその求め方、二進数の仕組みや分数が有限小数または無限小数で表される仕組みを理解し、整数の性質を事象の考察に活用できるようになる。

本稿では、以上の観点を踏まえて、Euclid互除法、RSA暗号、素因数分解に関する発展的な教材を提示し、幾つかのアルゴリズムをC++言語などで記した算譜を与える。情報科学を学ぶ学生への教材としても有効と考える。

Abstract

In unit of the integer in mathematics A,there are teaching contents about divisor and multiple,Euclid algorithm,application of integer. By using these contents ,students can consider logically and express the matter related to integers ,and come to understand the mechanism of Euclid algorithm and the meaning of the solutions of indefinite equation and how to get solutions.Also,they understand the mechanism of diadic system and fraction expressing by finite /infinite decimal,and apply the properties of integer to consideration of matter.In this paper,based on the above,the author presented the progressive teaching contents about Euclid's algorithm , RSA cryptosystem, Factoring and gave C++ programs to execute these algorithms.The author conceive that these contents are useful for students studying informatics,also.

キーワード

発展的な教材（progressive teaching contents）、ユークリッドの互除法（Euclid's algorthm）、RSA暗号系（RSA public cryptosystem）、素因数分解（Factoring integer）

*北海道情報大学情報メディア学部情報メディア学科特任教授　Specially appointed Professor , Department of　Information　Media , faculty of Information Media

1 はじめに

久しぶりに高校数学「数学A」に導入されたEuclidの互除法は人類最古のアルゴリズムであり、B.C.300年頃に記されたEuclidの原論第7巻命題1に記載されている。ギリシャ数学はバビロニア数学のように代数が発達していなかったから図形的な計算で処理されている。この互除法は連分数の考え方と同じであり、また数の連分数展開から無理数への理解が一層深まるのである。

また、今日、インターネット上の商取引や重要文書の通信セキュリティの確保にPKIシステムが整備され、RSA暗号方式がそのベースになっている。これは、1977年MITのRivest、Shamir、Adlemanのグループが考案したもので彼らの頭文字をとってRSA暗号としている。彼らはこの功績でTuring賞（2002年）を受賞した。この暗号には初等整数論のいろいろな性質や巨大素数からなる巨大合成数が活用されている。もしこの数が意図的に因数分解できればたちまち暗号は解読されセキュリティは破綻する。

本稿では、Euclidの互除法と不定方程式、連分数と関連する幾つかのトピックスをはじめ、RSA暗号体系や素因数分解問題に関する素材を取り上げ、高校数学における整数論の教材を考察するものである。

2 Euclidの互除法について

公約数の考えは小学校5年生から学ぶ。例えば、横24ｃｍ、縦18ｃｍの長方形の厚紙がある。これを余り屑が出ないように、しかも出来るだけ大きな正方形に分けるとき、何センチメートルの正方形にすればいいか？という問題を考えるとき最大公約数の考えに自然に導かれるだろう。

ところで、例えば527と1147の最大公約数を求めるとき、小学5年生はまず2つの数の約数を調べるため素因数分解をしようとするだろう。奇素数$3,5,7,11,13,17$ で次々に割ってゆき、$527=31\times17$ を見つける。$\sqrt{527}=22.956\cdots$ だから19 までで見つかるはずである。

次に、1147 を17 または31で割って $1147=31\times37$ を得る。さらに、37 と17に公約数があるか否かチェックする。

$\sqrt{17}=4.123\cdots$ だから奇素数3で$17,37$ を割り、公約数は1 しかないことを知り、31が最大公約数であると判断するだろう。

以上の操作で必要な割り算の回数は、合計$6+2+2=10$回 となる。

一方、Euclidの互除法を使った場合は

$$1147=2\times527+93$$
$$527=5\times93+62$$
$$93=1\times62+31$$
$$62=2\times31$$

計4回の割り算で済み、互除法の効率の良さを納得することになる。

整数a,bの最大公約数$\gcd(a,b)$ を求めるEuclidの互除法は次のようなアルゴリズムの形式で表される（Knuth,1972）。

二つの整数$a,b(a>b)$に対して

E1.aをbで割り、余り$r(0\le r<b)$とする

E2.$r=0$ならば終了し、$b=\gcd(a,b)$

$r\neq0$ ならばE3.に行く

E3.代入操作$a\leftarrow b,b\leftarrow r$　E1.に戻る

また、割り算を用いない次のような方法も

ある。t,s を変数として

F1.$t \leftarrow a, s \leftarrow b$

F2.$t = s$ なら s を $\gcd(a,b)$ として終了

F3.$t > s$ ならば $t \leftarrow t - s$ とし 、$t < s$ なら $s \leftarrow s - t$ として F2 へ飛ぶ

この VC++の算譜は次のようになる。

```
#include "stdafx.h"
#include <iostream>
#include <stdio.h>
#include <stdlib.h>

int main(void)
{  int a,b,x,y,i;
for (i=1;i=10;i++)
 {
puts("数を入力してください");
    scanf("%d",&a);
puts("数をもう一つ入力してください¥n");
scanf("%d",&b);
  x=a;  y=b;
    do {  if (x>y)  x = x-y;
          else      y = y-x;
       }
    while (x != y);   {
printf("最大公約数?＝?%d¥n",y);
   }
        return 0;
}
```

527 と 1147 の最大公約数を求めると次の通りである。

数字を入力してください
527
数字をもう一つ入力してください
1147
最大公約数＝31

また、LIST 処理言語 GNUCLISP を用いて記述すると次のような再帰的関数呼び出し(recursive call)を用いた美しい算譜となる。

```
[1]> (defun   gcdivisor  ( x   y )
      (cond (    (=   x   y )  y
                 )
                ( (<   x   y)
                  (gcdivisor   x   (-   y   x))
                 )
                ( t   (gcdivisor   (-   x   y )   y )
                  )))
```

91 と 104 の最大公約数を求める実行結果は次の通りである。

```
GCDIVISOR
[2]> (gcdivisor   91   104   )
13
```

2-1　E1～E3 の各操作では常に最大公約数が保存される。

$$\gcd(a,b) = \gcd(b,r)$$

（証）$\gcd(a,b) = d$ とおき、

$a = da', b = db'$　a', b' は互いに素とする。このとき a,b の任意の公約数 $s(\neq 1)$ は d の約数になる。もし d の約数でなければ a', b'

の公約数となり矛盾する。E1の操作で商を q とすれば、d は $b, a-bq=r$ の約数だから $\gcd(b,r)$ の約数である。

一方、$\gcd(b,r)$ は b,r の約数だから、$bq+r=a, b$ の公約数となり、d の約数である。

具体的な数で互除法を実行してみる。

例1　$1147, 1071$ の最大公約数を求める。

互除法の操作は次の5回になる。

$$1147=1\times1071+76$$
$$1071=14\times76+7$$
$$76=10\times7+6$$
$$7=1\times6+1$$
$$6=6\times1$$

$$\therefore \gcd(1147,1071)=1$$

このとき次式が成り立つ。

$$\gcd(1147,1071)=\gcd(1071,76)$$
$$=\gcd(76,7)=\gcd(7,6)=1$$

以上を一般的に表現する。整数 a,b を考え、$\frac{a}{b}$ は有理数とする。a を b で割り、商を k_0 余りを x_2 とおく。　$a=x_0, b=x_1$ とする。以下、互除法の操作E1～E3を続ける。

$x_1>x_2>\cdots\geq0$ だから、この操作は m 回で終了し、アルゴリズムは停止する。

$$a=x_0=k_0x_1+x_2 \qquad 0<x_2<x_1$$
$$b=x_1=k_1x_2+x_3 \qquad 0<x_3<x_2$$
$$x_2=k_2x_3+x_4 \qquad 0<x_4<x_3$$

・・・・・・・・・

$$x_{m-2}=k_{m-2}x_{m-1}+x_m \qquad 0<x_m<x_{m-1}$$
$$x_{m-1}=k_{m-1}x_m \qquad x_{m+1}=0$$

$$\gcd(x_{i-1},x_i)=\gcd(x_i,x_{i+1})$$

$$i=1,2,\cdots,m-1 \quad x_m=\gcd(a,b)$$

以上を例1の場合で確認してみる。

割る数、割られる数の系列 $\{x_i\}$ は

$$[x_0,x_1,x_2,x_3,x_4,x_5]=[1147,1071,76,7,6,1]$$

商となる数の系列 $\{k_j\}$ は

$$[k_0,k_1,k_2,k_3,k_4]=[1,14,10,1,6]$$

これらの数の生成にはどういうカラクリが潜んでいるだろうか？という疑問が湧いてくるだろう。それを考察してみる。

2-2　まず a,b を $x_i\ (i=2,3,\cdots m)$ で表わすことを考える。

$$a=x_0$$
$$=k_0x_1+x_2$$
$$=k_0(k_1x_2+x_3)+x_2=(1+k_0k_1)x_2+k_0x_3$$
$$=(1+k_0k_1)(k_2x_3+x_4)+k_0x_3$$
$$=(k_0+k_2+k_0k_1k_2)x_3+(1+k_0k_1)x_4=\cdots$$

各段階の x_n の係数を p_n とおく。

$$p_0=1, p_1=k_0$$
$$p_2=1+k_0k_1=p_1k_1+p_0$$

$$p_3=k_0+k_2+k_0k_1k_2$$
$$=(1+k_0k_1)k_2+k_0=p_2k_2+p_1$$

・・・・

$\{p_n\}$ は次の漸化式を満たす。

$$p_n = p_{n-1}k_{n-1} + p_{n-2}$$

これを用いると

$$a = x_0 = p_n x_n + p_{n-1}x_{n+1} \quad \cdots ①$$

と予想できる。

また、

$$b = x_1 = k_1 x_2 + x_3$$
$$= k_1\left(k_2 x_3 + x_4\right) + x_3$$
$$= \left(1 + k_1 k_2\right)x_3 + k_1 x_4 = \cdots$$

各段階の x_n の係数を q_n とおく。

$$q_0 = 0, q_1 = 1,$$
$$q_2 = k_1 = q_1 k_1 + q_0$$
$$q_3 = k_1 k_2 + 1 = q_2 k_2 + q_1$$
$$\cdots\cdots$$

$\{q_n\}$ は次の漸化式を満たす。

$$q_n = q_{n-1}k_{n-1} + q_{n-2}$$

これを用いると

$$b = x_1 = q_n x_n + q_{n-1}x_{n+1} \quad \cdots ②$$

と予想できる。

①、②を数学的帰納法で証明する。

$n = 1$ のとき

$$\begin{pmatrix} a \\ b \end{pmatrix} = \begin{pmatrix} k_0 x_1 + x_2 \\ k_1 x_2 + x_3 \end{pmatrix} = \begin{pmatrix} p_1 x_1 + p_0 x_2 \\ q_2 x_2 + q_1 x_3 \end{pmatrix}$$ となり

成り立つ。

$n-1$ まで仮定して

$$\begin{pmatrix} a \\ b \end{pmatrix} = \begin{pmatrix} p_{n-1} & p_{n-2} \\ q_{n-1} & q_{n-2} \end{pmatrix}\begin{pmatrix} x_{n-1} \\ x_n \end{pmatrix}$$
$$= \begin{pmatrix} p_{n-1} & p_{n-2} \\ q_{n-1} & q_{n-2} \end{pmatrix}\begin{pmatrix} k_{n-1}x_n + x_{n+1} \\ x_n \end{pmatrix}$$
$$= \begin{pmatrix} \left(p_{n-1}k_{n-1} + p_{n-2}\right)x_n + p_{n-1}x_{n+1} \\ \left(q_{n-1}k_{n-1} + q_{n-2}\right)x_n + q_{n-1}x_{n+1} \end{pmatrix}$$
$$= \begin{pmatrix} p_n x_n + p_{n-1}x_{n+1} \\ q_n x_n + q_{n-1}x_{n+1} \end{pmatrix}$$

よって、成り立つ。

なお、$n = m$ のとき $x_{m+1} = 0$ だから

$$a = p_m x_m, b = q_m x_m$$

例えば $a = 1147, b = 1071$ の場合、

$$k_0 = 1, k_1 = 14, k_2 = 10, k_3 = 1, k_4 = 6$$
$$m = 5$$

$$p_0 = 1, p_1 = k_0 = 1$$
$$p_2 = p_1 k_1 + p_0 = 1\times 14 + 1 = 15$$
$$p_3 = p_2 k_2 + p_1 = 15\times 10 + 1 = 151$$
$$p_4 = p_3 k_3 + p_2 = 151\times 1 + 15 = 166$$
$$p_5 = p_4 k_4 + p_3 = 166\times 6 + 151 = 1147$$

$$q_0 = 0, q_1 = 1, q_2 = k_1 = 14$$
$$q_3 = q_2 k_2 + q_1 = 14\times 10 + 1 = 141$$
$$q_4 = q_3 k_3 + q_2 = 141\times 1 + 14 = 155$$
$$q_5 = q_4 k_4 + q_3 = 155\times 6 + 141 = 1071$$

$$1147 = x_0 = k_0 x_1 + x_2 = 1\times 1071 + 76$$
$$1071 = x_1 = k_1 x_2 + x_3 = 14\times 76 + 7$$
$$76 = x_2 = k_2 x_3 + x_4 = 10\times 7 + 6$$
$$7 = x_3 = k_3 x_4 + x_5 = 1\times 6 + 1$$
$$6 = x_4 = k_4 x_5 = 6\times 1$$

$$\gcd(x_0,x_1)=\gcd(1147,1071)$$
$$\gcd(x_1,x_2)=\gcd(1071,76)$$
$$\gcd(x_2,x_3)=\gcd(76,7)$$
$$\gcd(x_3,x_4)=\gcd(7,6)$$
$$\gcd(x_4,x_5)=\gcd(6,1)=1=x_5$$

2-3 $\{p_n\},\{q_n\}$ は次式を満たす。

$$\begin{vmatrix} p_n & p_{n-1} \\ q_n & q_{n-1} \end{vmatrix}=(-1)^n \quad \cdots ③$$

（証）

$n=1$ のとき、$p_1=k_0, p_0=q_1=1, q_0=0$

$$\therefore \begin{vmatrix} p_1 & p_0 \\ q_1 & q_0 \end{vmatrix}=\begin{vmatrix} k_0 & 1 \\ 1 & 0 \end{vmatrix}=-1$$

$n-1$ のとき成り立つと仮定する。

$$\begin{vmatrix} p_n & p_{n-1} \\ q_n & q_{n-1} \end{vmatrix}=\begin{vmatrix} p_{n-1}k_{n-1}+p_{n-2} & p_{n-1} \\ q_{n-1}k_{n-1}+q_{n-2} & q_{n-1} \end{vmatrix}$$
$$=k_{n-1}\begin{vmatrix} p_{n-1} & p_{n-1} \\ q_{n-1} & q_{n-1} \end{vmatrix}-\begin{vmatrix} p_{n-1} & p_{n-2} \\ q_{n-1} & q_{n-2} \end{vmatrix}=(-1)^n$$

2-4 x_n を a,b についての 1 次式で表す。

$n=1$ のとき

$-x_1=0\cdot x_0-1\cdot x_1$ から

$-x_1=q_0x_0-p_0x_1=q_0a-p_0b$

$n=2$ のとき、

$x_2=x_0-k_0x_1=q_1x_0-p_1x_1=q_1a-p_1b$

以上から、次式が予想される。

$$(-1)^n x_n=q_{n-1}a-p_{n-1}b$$

（証）

①、②を行列で表す。

$$\begin{pmatrix} a \\ b \end{pmatrix}=\begin{pmatrix} p_n & p_{n-1} \\ q_n & q_{n-1} \end{pmatrix}\begin{pmatrix} x_n \\ x_{n-1} \end{pmatrix}$$

逆行列をとって③を用いる。

$$\begin{pmatrix} x_n \\ x_{n+1} \end{pmatrix}=\begin{pmatrix} p_n & p_{n-1} \\ q_n & q_{n-1} \end{pmatrix}^{-1}\begin{pmatrix} a \\ b \end{pmatrix}$$
$$=\frac{1}{(-1)^n}\begin{pmatrix} q_{n-1} & -p_{n-1} \\ -q_n & p_n \end{pmatrix}\begin{pmatrix} a \\ b \end{pmatrix}$$
$$=(-1)^n\begin{pmatrix} q_{n-1}a-p_{n-1}b \\ -q_na+p_nb \end{pmatrix}$$

$$\therefore x_n=(-1)^n q_{n-1}a-(-1)^n p_{n-1}b \quad \cdots ④$$

2-5 Euclid の互除法の除算の回数は小さい方の数の 10 進数表示の桁数の 5 倍を超えない（*Lamė* の定理）。

これは数列 $x_0,x_1,\cdots x_m$ を Fibonacci 数列と比較することで証明できる。そのために補題を一つ必要とする。

補題 Fibonacci 数列 $\{f_k\}$ において

$f_{5p+1}>10^p$ が成り立つ。

例えば、

$$f_6=f_5+f_4=f_4+2f_3+f_2$$
$$=3f_3+2f_2=5f_2+3f_1$$
$$=8f_1+5f_0=13>10^1$$

（証）Fibonacci 数列の一般項は

$$f_k=\frac{1}{\sqrt{5}}\{\alpha^{k+1}-\beta^{k+1}\}$$

$$\alpha=\left(1+\sqrt{5}\right)/2, \beta=\left(1-\sqrt{5}\right)/2$$

これに $k=5p+1$ を代入

$$f_{5p+1}=\frac{3+\sqrt{5}}{2\sqrt{5}}\left\{\alpha^5\right\}^p-\frac{3-\sqrt{5}}{2\sqrt{5}}\left\{\beta^5\right\}^p$$

$\alpha^5\approx 11.090\cdots>11$　　$\beta^5\approx -0.090\cdots$

よって、$f_{5p+1}\geq\frac{3+\sqrt{5}}{2\sqrt{5}}\cdot 11^p-\frac{3-\sqrt{5}}{2\sqrt{5}}$

このとき右辺は10^p 以上となることが分かる。

まず、$p=0$ のときは明らか。

$p\geq 0$ のとき成り立つと仮定する。

$$\frac{3+\sqrt{5}}{2\sqrt{5}}\cdot 11^{p+1}-\frac{3-\sqrt{5}}{2\sqrt{5}}$$
$$=11\left(\frac{3+\sqrt{5}}{2\sqrt{5}}\cdot 11^p-\frac{3-\sqrt{5}}{2\sqrt{5}}\right)+10\cdot\frac{3-\sqrt{5}}{2\sqrt{5}}$$
$$\geq 11\cdot 10^p\geq 10^{p+1}$$

そこで、$\left\{x_i\right\}$ $i=0,1,2,\cdots m$ を逆順に

$\left\{l_j\right\}$ と定め、$\left\{f_j\right\}$ と比較する。

$l_0=x_m\geq f_1=1$

$x_{m-1}>x_m, k_{m-1}>1$ より

$l_1=x_{m-1}=k_{m-1}x_m\geq l_0+1\geq f_0+f_1=f_2$

$l_2=x_{m-2}=k_{m-2}x_{m-1}+x_m\geq l_1+l_0$

$\geq f_2+f_1=f_3$

そこで、$l_j\geq f_{j+1}$ と仮定する。数学的帰納法で示す。

$l_{j+1}=x_{m-j-1}=k_{m-j-1}x_{m-j}+x_{m-j+1}$

$\geq l_j+l_{j-1}\geq f_{j+1}+f_j=f_{j+2}$

したがって、２つの整数$a,b\left(a>b\right)$ で

Euclid の互除法の回数m が$m\geq 5q+1$ とする。このとき、

$b=x_1=l_{m-1}\geq f_m\geq f_{5q+1}>10^q$

$\therefore \log_{10}b>q$ が成り立つ。

この対偶は$\log_{10}b\leq q$ ならば、$m<5q$ となる。よって、互除法の回数は$5\lceil\log_{10}b\rceil$ でおさえられる。

ただし、$\lceil x\rceil$ はx を超える最小の整数であり、$x=\log_{10}b$ ならばb の 桁数を表す。

例　$\lceil\log_{10}1234\rceil=\lceil\log_{10}1.234+3\rceil=4$

３ Euclid 互除法と不定方程式

不定方程式の解法への応用を考える。

3-1　不定方程式　$ax+by=d$　・・⑤

の一般解を求める。

⑤が解をもつためにはd が$\gcd\left(a,b\right)$

の倍数となることが必要十分である。

いま、$d=\gcd\left(a,b\right)$ とする。

Euclid の互除法を用いて **2-4** の④式が求められたとする。

$x_m=d=\gcd\left(a,b\right)$

$=\left(-1\right)^m q_{m-1}a+\left(-1\right)^{m-1}p_{m-1}b$

$a=a'd, b=b'd$ とおく。

この式と⑤から

$a'\left(x-\left(-1\right)^m q_{m-1}\right)$

$=b'\left(-y+\left(-1\right)^{m-1}p_{m-1}\right)$

a', b' は互いに素だから

$x=\left(-1\right)^m q_{m-1}+b't$

$$y=(-1)^{m-1}p_{m-1}-a't \quad \cdots ⑥$$

（t：整数）

これが⑤の一般解である。

例２　不定方程式　$1147x+1071y=1$　の一般解を求める。

$m=5, p_4=166, q_4=155$
$a'=1147, b'=1071$
を⑥に代入すれば
一般解は
$x=-155+1071t, y=166-1147t$

3-2　なお、次のように行列を使って特殊解を求める方法がある（岩堀、1983）。

$a=x_0=k_0x_1+x_2$　から

$$\begin{pmatrix}a\\b\end{pmatrix}=\begin{pmatrix}x_0\\x_1\end{pmatrix}=\begin{pmatrix}k_0&1\\1&0\end{pmatrix}\begin{pmatrix}x_1\\x_2\end{pmatrix}$$

同様に　$\begin{pmatrix}x_1\\x_2\end{pmatrix}=\begin{pmatrix}k_1&1\\1&0\end{pmatrix}\begin{pmatrix}x_2\\x_3\end{pmatrix}$

$$\begin{pmatrix}x_2\\x_3\end{pmatrix}=\begin{pmatrix}k_2&1\\1&0\end{pmatrix}\begin{pmatrix}x_3\\x_4\end{pmatrix}、\quad ・・・、$$

$$\begin{pmatrix}x_{m-1}\\x_m\end{pmatrix}=\begin{pmatrix}k_{m-1}&1\\1&0\end{pmatrix}\begin{pmatrix}x_m\\0\end{pmatrix}$$

が成り立つ。

これから

$$\begin{pmatrix}a\\b\end{pmatrix}=\begin{pmatrix}x_0\\x_1\end{pmatrix}$$
$$=\begin{pmatrix}k_0&1\\1&0\end{pmatrix}\begin{pmatrix}k_1&1\\1&0\end{pmatrix}\cdots\begin{pmatrix}k_{m-1}&1\\1&0\end{pmatrix}\begin{pmatrix}x_m\\0\end{pmatrix}$$
$$=\begin{pmatrix}\alpha&\beta\\\gamma&\delta\end{pmatrix}\begin{pmatrix}x_m\\0\end{pmatrix}=\begin{pmatrix}\alpha&\beta\\\gamma&\delta\end{pmatrix}\begin{pmatrix}d\\0\end{pmatrix}$$

$$\begin{pmatrix}a\\b\end{pmatrix}=\begin{pmatrix}\alpha&\beta\\\gamma&\delta\end{pmatrix}\begin{pmatrix}d\\0\end{pmatrix}$$

$$\begin{pmatrix}d\\0\end{pmatrix}=\begin{pmatrix}\alpha&\beta\\\gamma&\delta\end{pmatrix}^{-1}\begin{pmatrix}a\\b\end{pmatrix}=\begin{pmatrix}x&y\\z&w\end{pmatrix}\begin{pmatrix}a\\b\end{pmatrix}$$

$\therefore d=ax+by$

$$\begin{pmatrix}x&y\\z&w\end{pmatrix}$$
$$=\begin{pmatrix}k_{m-1}&1\\1&0\end{pmatrix}^{-1}\begin{pmatrix}k_{m-2}&1\\1&0\end{pmatrix}^{-1}\cdots\begin{pmatrix}k_0&1\\1&0\end{pmatrix}^{-1}$$
$$=\begin{pmatrix}0&1\\1&-k_{m-1}\end{pmatrix}\begin{pmatrix}0&1\\1&-k_{m-2}\end{pmatrix}\cdots\begin{pmatrix}0&1\\1&-k_0\end{pmatrix}$$

・・⑦

$$\begin{pmatrix}k_i&1\\1&0\end{pmatrix}^{-1}=-\begin{pmatrix}0&-1\\-1&k_i\end{pmatrix}=\begin{pmatrix}0&1\\1&-k_i\end{pmatrix}$$

⑦を用いて**2-1**の例１の特殊解を求める。

$k_0=1, k_1=14, k_2=10, k_3=1, k_4=6$
だから　⑦は

$$\begin{pmatrix}x&y\\*&*\end{pmatrix}=\begin{pmatrix}0&1\\1&-6\end{pmatrix}\begin{pmatrix}0&1\\1&-1\end{pmatrix}\begin{pmatrix}0&1\\1&-10\end{pmatrix}$$
$$\times\begin{pmatrix}0&1\\1&-14\end{pmatrix}\begin{pmatrix}0&1\\1&-1\end{pmatrix}$$

$$\begin{pmatrix}0 & 1\\ 1 & -6\end{pmatrix}\begin{pmatrix}1 & -10\\ -1 & 1\end{pmatrix}\begin{pmatrix}1 & -1\\ -14 & 15\end{pmatrix}$$
$$=\begin{pmatrix}0 & 1\\ 1 & -6\end{pmatrix}\begin{pmatrix}141 & -151\\ -155 & 166\end{pmatrix}$$
$$=\begin{pmatrix}-155 & 166\\ * & *\end{pmatrix}$$

特殊解は $x=-155, y=166$ である。

4　Euclid 互除法と連分数

Euclid 互除法の式から連分数の展開式が出てくる。

4-1　$1147,1071$ の互除法の各式変形を再掲する。

$$1147=1\cdot 1071+76, 1071=14\cdot 76+7$$
$$76=10\cdot 7+6, 7=1\cdot 6+1$$

これを連分数に展開すると

$$\frac{1147}{1071}=1+\frac{76}{1071}=1+\frac{1}{14+\frac{7}{76}}$$

$$=1+\frac{1}{14+\frac{1}{10+\frac{6}{7}}}=1+\frac{1}{14+\frac{1}{10+\frac{1}{1+\frac{1}{6}}}}$$

一般の a,b の場合は

$$\frac{a}{b}=\frac{x_0}{x_1}=k_0+\frac{x_2}{x_1}=k_0+\frac{1}{k_1+\frac{x_3}{x_2}}$$

$$=k_0+\frac{1}{k_1+\frac{1}{k_2+\frac{x_4}{x_3}}}=\cdots$$

$$=k_0+\frac{1}{k_1+\frac{1}{k_2+\cdots}}$$
$$+\frac{1}{k_{m-2}+\frac{1}{k_{m-1}}}$$

$=\left[k_0,k_1,\cdot\cdot,k_{m-1}\right]$ と書く。

これは **2-1** で扱った互除法の商系列の表現と同じである。

上の例では　$\frac{1147}{1071}=[1,14,10,1,6]$

なお、有理数は有限の連分数展開で表される。逆も成り立つ。

無理数の連分数展開はどうなるか？

例えば、$\sqrt{2}$ を連分数展開は

$$\sqrt{2}=1+\left(\sqrt{2}-1\right)=1+\frac{1}{1+\sqrt{2}}$$
$$=1+\frac{1}{2+\left(\sqrt{2}-1\right)}$$
$$=1+\frac{1}{2+\frac{1}{1+\sqrt{2}}}=[1,2,2,\cdots]$$
$$\cdots$$

となり、無限の連分数展開となる。

実は、任意の無理数は無限連分数に展開される。逆も成り立つ。

$\sqrt{2}=[1,2,2,\cdots]=\left[1,\dot{2}\right]$ ということは

$\sqrt{2}$ が無理数であることの証左である。

4-2　$\tau=\frac{1+\sqrt{5}}{2}$ の連分数展開を考える。

この数は黄金比と呼ばれ、$\tau^2-\tau-1=0$ を満たす正の解である。式変形すると

$$\tau = 1+\frac{1}{\tau} = 1+\frac{1}{1+\frac{1}{\tau}} = 1+\frac{1}{1+\frac{1}{1+\frac{1}{\tau}}}$$

$$= [1,1,\cdots]$$

黄金比はFibonacci数列の漸化式 $p_0 = p_1 = 1 \quad p_n = p_{n-1} + p_{n-2}$ を満たす数列$1,1,2,3,5,\cdots$に関連している。

辺の長さ1の正五角形ABCDE（図1）において、$\angle AEB = \frac{\pi}{5}$　点Aから線分BEに垂線を下した点をFとすれば

$$EF = \cos\frac{\pi}{5} \qquad BE = 2\cos\frac{\pi}{5}$$

$$AB = AE = 1$$

$$\frac{BE}{AB} = \frac{AB}{BP} \quad \text{より} \quad BP = \frac{1}{BE}$$

$$BE = BP + PE = BP + AE = BP + 1$$

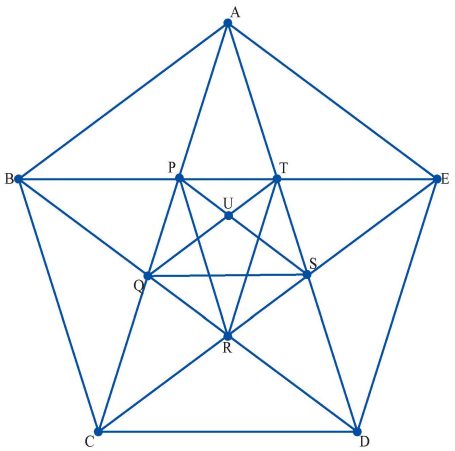

図1　五角形の対角線

$$\therefore BE = 1+\frac{1}{BE}$$

ここで$BE = \tau$とおけば　$\tau = 1+\frac{1}{\tau}$　となりBEの長さは黄金比になる。

$$BE = \frac{1+\sqrt{5}}{2}$$

正五角形の作図はこの比を使ってできる。

次に、線分同士の互除法を考える。

$$BE = AE\cdot 1 + BP$$
$$AE = BT = BP\cdot 1 + PT$$
$$BP = QT = PT\cdot 1 + UT$$
$$\cdots\cdots$$

これは以下のような無限連分数になる。

$$\tau = \frac{BE}{AE} = BE = 1+\frac{BP}{AE} = 1+\frac{1}{\frac{AE}{BP}}$$

$$= 1+\frac{1}{1+\frac{1}{\frac{BP}{PT}}} = 1+\frac{1}{1+\frac{1}{\frac{PT}{UT}}} = \cdots$$

よって、τは無理数となる。

5　RSA暗号系について

大きな2つの素数p,qを選びその積Nを定めておく。また、$(p-1)(q-1)$と互いに素なある整数r（閉める鍵）を選んでおく。N,rは公開鍵として公開される。いま、数値化された通信文（平文）をaとする。

5-1　$a^r \equiv b \pmod N$　となるbを求め、それを暗号文とする。受信側は暗号文bから元の通信文aを次の手順で解読する。

①　まず、公開鍵Nを素因数分解し、p,q

を求める。この素因数分解を適切な時間（多項式時間）と妥当な記憶容量内で処理するのは大変困難が伴う。これが RSA 暗号の安全性を保障している。

② $rs \equiv 1\left(mod\left(p-1\right)\left(q-1\right)\right)$ を満たす s は開ける鍵である。これから p,q,r から求め、暗号文 b を s 乗し、法 N に関する剰余を求める。

$$b^s \equiv \left(a^r\right)^s = a^{rs} \equiv a^1 = a\left(mod\ N\right)$$

こうして、元の通信文 a を得る。

RSA 暗号系に用いられる初等整数論の知識を **5-2**、**5-3** にまとめておく。

5-2 不定方程式 $ax+by=c$ が解をもつためには、$gcd\left(a,b\right)\left|c\right.$ が成り立つことが必要十分条件である。

（証）必要条件は明らかである。十分条件は、

$n = Min\left\{ax+by\left|ax+by>0\right.\right\}$ とおく。

$ax+by=m$ となる数 m は n の倍数になる。

何故なら、$m=nq+r \quad 0\le r<n$

$as+bt=n \qquad ax+by=m$

$$r=m-nq=\left(ax+by\right)-\left(as+bt\right)q$$
$$=a\left(x-sq\right)+b\left(y-tq\right)\ge 0$$

n の最小性から $r=0$ となる。

$a=a\cdot 1+b\cdot 0 \quad b=a\cdot 0+b\cdot 1$ よって、a,b は n の倍数。つまり、n は a,b の公約数であるから、$d=gcd\left(a,b\right)$ の約数となる。

$n\le d$ 他方、n は a,b の 1 次結合 $as+bt$ と表されているから、d の倍数となり $d\le n$

よって $n=d$

したがって、$d=as+bt$ であり d の倍数 c も a,b の 1 次結合で表される。

5-3 次に、Euler の定理、Fermat の定理について述べる。

1 から N までの自然数で N と互いに素な数の個数を Euler 関数 $\varphi\left(N\right)$ という。

p が素数ならば $\varphi\left(p\right)=p-1$ は明らかである。q も素数で $N=pq$ ならば次式が成り立つ。 $\varphi\left(N\right)=\left(p-1\right)\left(q-1\right)$

何故なら、1から $\left(N-1\right)$ までの数で p,q の約数はそれぞれ $\left(q-1\right),\left(p-1\right)$ 個ずつあるから

$$\varphi\left(N\right)=pq-1-\left(p-1\right)-\left(q-1\right)$$
$$=pq-p-q+1$$
$$=\left(p-1\right)\left(q-1\right)$$

（Euler の定理）

自然数 m、整数 a で $gcd\left(a,m\right)=1$ ならば

$$a^{\varphi(m)} \equiv 1\left(mod\ m\right)$$

（証） $\left\{x_1,x_2,\cdots,x_l\right\}$ を 法 m に関する既約剰余系とする。$l=\varphi\left(m\right)$ このとき、ax_i は m と互いに素だから

$\left\{ax_1,ax_2,\cdots,ax_l\right\}$ はまた既約剰余系となる。

$$(ax_1)(ax_2)\cdots(ax_l)\equiv x_1x_2\cdots x_l\,(mod\ m)$$

$$a^l x_1x_2\cdots x_l \equiv x_1x_2\cdots x_l\,(mod\ m)$$

$$(x_1x_2\cdots x_l, m)=1$$

$$\therefore\ a^l = a^{\varphi(m)}\equiv 1(mod\ m)$$

（Fermat の定理）

Euler の定理で $N=p$（素数）とすれば $\varphi(p)=p-1$ だから、$gcd(a,p)=1$ のとき、

$a^{p-1}\equiv 1(mod\ p)$ が成り立つ。

なお、$x\equiv 1(mod\ p-1)$ ならば

$x=1+(p-1)t$ となり

$$a^x = a^{(p-1)t+1} = \left(a^{p-1}\right)^t\cdot a\equiv 1^t\cdot a$$
$$= a(mod\ p)$$

$\therefore a^x\equiv a(mod\ p)$ となることに留意しておく。

5-4 暗号解読の計算②を考察する。

$$gcd(r, p-1)=1, gcd(r, q-1)=1$$

だから次の１次不定方程式は解をもつ。

$$ru+(p-1)v=1$$
$$rx+(q-1)y=1 \quad \cdots ①$$

この式から

$$(u-x)r=-v(p-1)+y(q-1)$$

$gcd(p-1,q-1)=d$ とすれば、

$d\mid(u-x)$ となる。よって、次の不定方程式は解を持つ。

$$(p-1)w+(q-1)t=u-x \quad \cdots ②$$

この特殊解を w, t とする。

$u-(p-1)w=x+(q-1)t=s$ とおけば、s が求めるものである。

公開健 r に対して

$$rs=r(u-(p-1)w)$$
$$=ru-(p-1)rw$$
$$=1-v(p-1)-(p-1)rw$$
$$\equiv 1(mod\ p-1)$$

$$rs=r(x+(q-1)t)$$
$$=rx+rt(q-1)$$
$$=1-y(q-1)+rt(q-1)$$
$$\equiv 1(mod\ q-1)$$

したがって、

$$rs\equiv 1(mod\ p-1)$$
$$rs\equiv 1(mod\ q-1)$$

r,s は既約剰余類群 $(Z/\varphi(N)Z)^\times$ における乗法に関する逆元同士である。

5-3 で留意したことから

$$a^{rs}\equiv a(mod\ p) \qquad a^{rs}\equiv a(mod\ q)$$

$a^{rs}\equiv a(mod\ pq)=a(mod\ N)$ となり解読される（一松、1980）。

例３

公開健はそれぞれ $N=143, r=7$ とする。

通信文（平文）を$a=5$ とする。
このとき、暗号文は

$$a^7=5^7=78125$$
$$=546\times143+47\equiv47(mod\,143)$$

これを解読してみよう。
まず、公開健Nを素因数分解する。$N=11\times13$　よって、$p=11,q=13$と確定する。
そこで、不定方程式①が求まる。

$$7u+10v=1,7x+12y=1$$

これをEuclidの互除法を用いて解く。
$10=1\times7+3,7=2\times3+1$から

$$1=7-2\times(10-1\times7)=3\times7-2\times10$$

$$\therefore\ u=3,v=-2$$

$12=1\times7+5,7=1\times5+2,5=2\times2+1$から

$$1=5-2\times(7-1\times5)=3\times5-2\times7$$
$$=3\times(12-1\times7)-2\times7$$
$$=3\times12-5\times7$$

$$\therefore x=-5,y=3$$

これから、次の不定方程式②を得る。

$$10w+12t=u-x=8$$

この方程式の特殊解は　$w=-4,t=4$
よって、開ける鍵は

$$s=u-w\times10=3+4\times10=43$$

47^{43} は10進72桁の数となる。

$47^{43}(mod\,143)$はMathematicaで計算すると Mod[47^43,143]=5 となる。

6　素数の判定

5で述べたとおりRSA暗号には適切な素数が必要となる。そのような数を構成するためには、ある数が素数か否か見極めることが必要となってくる。こうして素数の判定が重要となる。

6-1　数nに対する素数判定法して、良く知られたEratosthenesの篩法は、$\sqrt{n}$以下のすべての素数での割り算を試すので効率が悪い。**2-5** で述べたように、$1\le k\le n$、$gcd(k,n)$を求める計算回数はkの10進桁数の5倍以下となる。例えば$n=10^{50}$ $m=\left[\sqrt{n}\right]$とすると計算総数は次式を超えることはない。

$$5\sum_{k=1}^{m}log_{10}k=5log_{10}m!$$
$$\sim5log_{10}\left(\sqrt{2\pi}\left(m^{m+1/2}\right)e^{-m}\right)$$

$m=10^{25}$のときの値は1.27172×10^{27}となり、1回の互除法の平均計算が100億分の1秒かかるとしても40億年かかる。

次に、Fermatの小定理を用いた方法がある（Fermat-test）。**5-3** で述べたように、Fermatの定理「nが素数ならば、
$a^{n-1}\equiv1(\mathrm{mod}\,p)$ が成り立つ」の対偶
「$a^{n-1}\equiv1(\mathrm{mod}\,n)$ が成り立たないaがあれば、nは素数ではない」を使う。

$a(1<a\le n-1,gcd(a,n)=1)$をランダムに選び、$a^{n-1}\equiv1(\mathrm{mod}\,n)$が成り立つかどうか調べ、成りたたなければ$n$が合成数となる。このアルゴリズムを効率化したのが、次のような方法（Miller-Rabin法）である。

6-2　与えられたnが奇数とする。$n-1$を2

で割り切って、$n-1=2^s\cdot d$（dは奇数）となるs,dを求める。また、判定回数kを指定しておく。

MR1. $2\le a\le n-1$である整数a（底）をランダムに選ぶ

MR2. $a^d\equiv 1\pmod n$であるか、または

$a^{2^r\cdot d}\equiv -1\pmod n$となる$r\left(0\le r\le s-1\right)$があるか調べる。あれば$n$は「素数」と見做す。なければ「合成数」と見做す。

MR3. k回行ってl回「素数」と出れば、nが合成数である確率は2^{-2l}以下であると判定する。

このアルゴリズムの論拠を考えてみる。nは奇素数で$gcd\left(a,n\right)=1$とする。$n-1$を2で割り切って、$n-1=2^s\cdot d$　dは奇数となる。このとき、次のことが成り立つ。

補題　nが奇素数ならば上の操作で、次の(1),(2)のいずれかが必ず成り立つ。

(1)　$a^d\equiv 1\pmod n$

(2)　$a^{2^s\cdot d}\equiv -1\pmod n$

（証明）　nは素数だから

Fermatの定理より、$a^{n-1}\equiv 1\pmod n$

$a^{n-1}=a^{2^s\cdot d}=\left(a^{2^{s-1}\cdot d}\right)^2\equiv 1$より$a^{2^{s-1}\cdot d}\equiv 1$または$a^{2^{s-1}\cdot d}\equiv -1$である。何故なら、$Z/nZ$は整域だから（$\bar{a}\cdot\bar{b}\equiv\bar{0}\Leftrightarrow\bar{a}\equiv\bar{0}$または$\bar{b}\equiv\bar{0}$）、$\bar{x}^2=\bar{1}$ならば、$\bar{x}=1$または$\bar{x}=\overline{-1}=-\bar{1}$が成り立つからである。もし、$a^{2^{s-1}\cdot d}\equiv -1$なら(2)が成り立つ。また、$a^{2^{s-1}\cdot d}\equiv 1$ならば、同じ議論を続けていく。このとき、$2^{s-1}>2^{s-2}>2^{s-3}>\cdots=2^0=1$となる。最後は、$a^{2^0\cdot d}=a^d\equiv 1\pmod n$となり、(1)が成り立つ。

補題の対偶は、「ある$a\in\left(Z/nZ\right)^\times$が存在し、(1),(2)の両方が成り立たないならば、nは合成数である」ことになる。これを精密に調べ、判定するのがMiller-Rabin法である。

上のアルゴリズムにおいて、nが合成数であるにもかかわらず、ある底に対して「素数」であると誤った判定を下す確率は高々2^{-2}であることが分かっている（深川、2002）。したがって、試行回数を多くすることで判定の妥当性をあげることができる。

6-3　MR1.～MR3.のアルゴリズムをVC++で書くと次のような算譜となる。

```
// 素数判定プログラム.cpp : //

#include "stdafx.h"
#include "stdio.h"
#include "math.h"
#include "cstdlib"
#include <ctime>

int main(void)
{ int n,k,s,x,t,r,dd ; long double
a,z,y,p,w,yy,ww,probability;
   srand((unsigned)time(NULL));
```

```
    printf("5以上の奇数を入力してください
'¥n");
    scanf("%d",&n);
    printf("回数を入力してください'¥n");
    scanf("%d",&k);
    x=n-1;
    dd=s=r=0;
    while (r==0)
      { dd=x/2;
       s=s+1;
       x=dd;
       r=x%2;
      }
    a=probability=0;
    for (int j=1;j<=k;j++)
     {
         a=2+rand()%n;
        yy=powl(a,dd)-1.0;
         y=fmodl(yy,n);
         if (y==0)
          { probability=probability+1.0;
               continue;
              }
         else
         {    p=s-1;
         while (p >= 0.0)
           {
         z=dd*powl(2.0,p);
        ww=powl(a,z)+1;
          w=fmodl(ww,n);
          if (w==0.0)
            { probability=probability+1.0;
              break;
             }
          else
             p=p-1.0;
                }
              }
            }
          }
         if (probability > 0.0)
          printf("数 n は素数です。この判定が
          誤る確率は高々%f です。
          ¥n",1/powl(4,probability));
         else
          printf("数 n は合成数です。¥n");
        return (0);
      }
```

以下は $n=29$ 、$k=4$ のとき実行結果である。

```
5 以上の奇数を入力してください
29
回数を入力してください
4
a=16.000000
y=0.000000
a=7.000000
y=0.000000
a=13.000000
y=27.000000
w=2.000000
w=0.000000
a=2.000000
y=11.000000
w=0.000000
probability=4.000000
数 n は素数です。この判定が誤る確率は
高々0.003906 です。
```

この算譜では k 回で判定できない場合は合成数としている。合成数は原理的に判定可能であり、素数の判定も確率的に可能なはず

である。しかし、そうならないのは底の選び方から巨大な冪計算が必要な場合があり、有効桁数の限界が立ちはだかっているからである。

なお、近年、関数電卓の機能向上は著しく、C++で出来ない計算が TI-nspireCX で出来たことを付け加えておく。

7　巨大数の素因数分解

多項式時間のアルゴリズムとは、入力サイズｎに対して、処理時間の上界がｎの多項式で表現され、能率の良いものである。例えば、バブルソートは　n(n―１)/2　で要素数ｎの2次多項式である。

7-1　多項式時間（Polynomial time）内で決定性アルゴリズムによって解ける問題は P 問題と呼ばれる。

また、その操作の中に可能性の集合Sの中から選択する機能$choice(S)$を含む非決定性アルゴリズムによって多項式時間で解ける問題は NP 問題（Non-deterministic Polynomial time）と呼ばれる。

決定性アルゴリズムにより多項式時間内で解ける問題は、非決定性アルゴリズムによる多項式時間内で解けるから、$P \subset NP$が成立する。$NP \subset P$　が成り立てば未解決問題「$NP \neq P$予想」は解けたことになる。もし、$NP = P$となるようなことになれば暗号系は破綻するといわれる。

NP 問題を解く決定性アルゴリズムでは可能なすべてをしらみつぶしに調べる方法をとるしかない。その時間量は入力サイズnの場合は$O\left(2^{p(n)}\right)$、$p(n)$はある多項式になることが分かっている（伊理、野崎、野下1980）。

従前の因数分解アルゴリズムの多く、例えば数体ふるい法、楕円曲線法などは決定性アルゴリズムである。サイズnの数の素因数分解を一般数体ふるい法で行う際の計算量は$O\left(\exp\left(c(\log n)^{1/3}(\log\log n)^{2/3}\right)\right)$　である(CRYPTREC,2001)など指数時間を要する。

他方、サイズnの大きさは暗号系の設計上重要な問題でもある。

7-2　近年、量子計算機が開発され、重ね合わせを用いた超並列計算が可能となった。さらに、1994 年、Shor が量子計算を用いた非決定性アルゴリズムを考案した(Shor,1994)。

Shor アルゴリズムは以下のように、S1～S6 からなる。いま、素因数分解したい合成数をnとし、$n = pq$かつp, qは素数と仮定する。

S1.　$\{1, 2, 3, \cdots, n\}$からランダムにaを選ぶ。

S2.　$\gcd(a, n) = 1$なら S3 へ行く。そうでないならば S1 に戻る

S3.　$a^r \equiv 1 \pmod{n}$となるrを、量子計算で求め S4 へ行く

S4.　rが偶数ならば S5 に行く。奇数ならば S1 に戻る

S5.　$\gcd\left(a^{r/2}+1, n\right), \gcd\left(a^{r/2}-1, n\right)$を求めて S6 へ行く

S6.　S5 で求めた数のいずれかがnならば S1 に戻る。そうでなければこれらの数が求める素因数p, qとなる。

このアルゴリズムの論拠を考える。

$$p = \mathrm{gcd}\left(a^{r/2}+1, n\right)$$
$$q = \mathrm{gcd}\left(a^{r/2}-1, n\right)$$
$$a^r - 1 = nn'$$
$$n = pp'' = qq''$$
$$a^{r/2}+1 = pp'$$
$$a^{r/2}-1 = qq'$$

$$\gcd(p', p'') = \gcd(q', q'') = 1$$

$$a^r - 1 = \left(a^{r/2}+1\right)\left(a^{r/2}-1\right)$$
$$= pp'qq' = nn' = pp''n' = qq''n''$$

これから

$$p'qq' = p''n', pp'q' = q''n' \qquad p' \mid n', q' \mid n'$$

から $n' = p'q'l$　$p = q''l$　p は素数だから $p = q'', l = 1$

または $p = l, q'' = 1$　$n = pq$ となる。$q = p''l$ からも同じ結果が出てくる。

S3 は、複数の r 値を選び、それぞれの計算を並行に走らせ成り立つものが出てきた時点で処理を終了させる木構造の計算となる。また、S5 は Euclid の互除法を使う。

このアルゴリズムの計算量は、数 n を２進数表示したときの桁数 $\log_2 n$ のオーダーになることが分かっている(栗山他、2005)。

例４　$n = 21$ に対して Shor アルゴリズムを適用してみる。S1 で、$a = 2$ を選ぶと $(2, 21) = 1$ だから S3 に行く。

$2^r \equiv 1 \pmod{21}$ となる最小の r は、$2^6 = 64 = 3 \times 21 + 1$　なので６である。

S4 に行き r は偶数だから⑤に行く。

S5 では、

$$2^{\frac{6}{2}} + 1 = 9, 2^{\frac{6}{2}} - 1 = 7$$

$$\gcd(9, 21) = 3, \gcd(7, 21) = 7$$

よって、$p = 3, q = 7$ となり因数分解出来たことになる。

7-3　このアルゴリズムをノイマン型計算機でシミュレートすることができる。VC++で書いた算譜は以下となる。

```
// 素因数分解の算譜.cpp //

#include "stdafx.h"
#include <iostream>
#include <cstdlib>
#include <cmath>
#include <time.h>
using namespace std;
// 関数gcdのproto-type宣言
 long double gcd ( long double x,long double
y);

int main(void)
{ int n,r ;
  long double
 a,aa,b,bb,bbb,c,cc,d,dd,rr ;
  srand((unsigned)time(NULL));
  cout  << "素因数分解する合成数を入力して
ください¥n";
  cin  >> n  ;
   // 1～nの間の数を無作為抽出
j1:  do {  a = rand() % n;
  // nと互いに素な数aを見つける}
```

```
while( aa != 1);
          aa = gcd ( a , n);
  for ( r = 1;r <= n ; r++)
   { b = powl ( a , r)-1;
    bb = fmodl ( b , n);
  if ( bb == 0)
  {   //  aのr乗 ≡ 1 (mod n)
   bbb = fmodl( r , 2);
     // r:偶数ならば素因数の生成へ進む
   if (bbb == 0)  goto j2;
// r:奇数ならばaを選びやり直す
   else goto j1;
    }
   }
j2: rr = r / 2;
         c = powl( a , rr)+1;
         cc = gcd ( c , n);  //第一因子
// 生成した数がnならば最初へ戻る
   if (cc == n) goto j1;
    else
         d = powl( a , rr )-1;
         dd = gcd ( d , n);   // 第二因子
//  生成した数がnならば最初に戻る
   if (d == n)  goto j1;
    else
         cout << "素因数分解成功！因数は¥n";
         cout <<  cc << "¥n"  <<  dd << "¥n" ;

         return 0;
}

long double gcd ( long double x, long double y)
{
         do
{
         if (x >y)
         x = x-y;
         else y = y-x;
}  while (x != y);
  return y;
}
```

例３で扱った $n=143(=13\times11)$ の計算実行の結果は次のようになる。

```
素因数分解する合成数を入力してください
143
a=82          ・・・初めに選んだ乱数
 b=81
 b=6723
 b=551367
 b=4.52122e+007
 b=3.7074e+009
 b=3.04007e+011
 b=2.49285e+013
 b=2.04414e+015
 b=1.6762e+017
 b=1.37448e+019
 b=1.12707e+021
 b=9.24201e+022
 b=7.57844e+024
 b=6.21432e+026
 b=5.09575e+028
a=5           ・・・別の乱数で試みる
 b=4
 b=24
 b=124
 b=624
 b=3124
 b=15624
 b=78124
 b=390624
```

b=1.95312e+006
b=9.76562e+006
b=4.88281e+007
b=2.44141e+008
b=1.2207e+009
b=6.10352e+009
b=3.05176e+010
b=1.52588e+011
b=7.62939e+011
b=3.8147e+012
b=1.90735e+013
b=9.53674e+013　・・r=20 で見つかる
rr=10
c=9.76563e+006
d=9.76562e+006
素因数分解成功！　因数は
13　・・ｃから抽出された第１因数
11　・・ｄから抽出された第２因数

8　終わりに

身近なインターネットにおける通信処理に整数論の定理が生かされていることを知れば、整数論に対する興味・関心が増すであろう。本稿で取り上げた素材は生徒に数学の有用性を理解させ、数学のよさを認識するきっかけづくりに好都合な教材となる。

また、整数の問題を計算機で解くアルゴリズムやプログラム言語への関心を少しでも持たせる教材になると考えている。

「数学活用」の教科書ではトピックスとして暗号の数理が取り上げられている(根上、2012)。「数学Ａ」での整数の性質の授業での数学的活動における課題学習のテーマとして取り上げてみるのもよかろう。

さらに、様々な素材が開発され、それらを用いて生徒を啓発し、眠っている数学的能力を目覚めさせ、伸ばす数学的活動が求められている（林、2014)。

整数に関するいろいろな問題はそういう素材として最適であり、今後、一層、教材開発が進展していくことが期待されている。

参考文献

[1] 伊理・野崎・野下(1980)：計算の効率化とその限界、数学セミナー、日本評論社
[2] 岩堀長慶(1983)：２次行列の世界、岩波書店
[3] 河田敬義(1992)：数論、岩波書店
[4] Knuth(1972)：The Art of Computer Programing,Volume1
[5] CRYPTREC(2001)：素因数分解問題調査研究報告書
[6] 栗山、佐野、古市(2005)：shor の素因数分解アルゴリズムにおける計算量の精密な評価、統計数理研究所、講究 1452 巻
[7] P.W.Shor(1994):Algorithms for Quantum computation:Discrete log and Factoring,Proc.of the 35th Annual IEEE Symp. On Foundations of Computer Science,p.124-134
[8] 根上生也(2012)：数学活用、啓林館
[9] 林雄一郎（2014)：高校数学における発展的オプション教材の意義について、北海道情報大学紀要第 25 巻第２号
[10] 一松信(1980)：暗号の数理、講談社
[11] 深川久(2002)：Miller-Rabin による素数の確率的判定法、大阪市立大学