

$1 + 1 = 0$ の世界での代数・幾何・応用

松本 眞（東京大学数理科学研究科）

2010/11/26, 東京大学教養学部・高校生特別講座

email: matumoto “at mark と呼ばれるもの” ms.u-tokyo.ac.jp

1 ÷ 7 の 10 進小数展開の余りの列

$x_1, x_2, \dots$

は、 $x_1 = 1$ として以下の法則で求める。

$$1 \times 10 = 10, \div 7 = 1 \text{ 余り } 3 =: x_2$$

$$3 \times 10 = 30, \div 7 = 4 \text{ 余り } 2 =: x_3$$

$$2 \times 10 = 20, \div 7 = 2 \text{ 余り } 6 =: x_4$$

$$6 \times 10 = 60, \div 7 = 8 \text{ 余り } 4 =: x_5$$

$$4 \times 10 = 40, \div 7 = 5 \text{ 余り } 5 =: x_6$$

$$5 \times 10 = 50, \div 7 = 7 \text{ 余り } 1 =: x_7$$

$$\begin{array}{r} 0.\dot{1}42857\dot{1} \\ 7 \overline{) 10} \\ \underline{7} \\ 30 \\ \underline{28} \\ 20 \\ \underline{14} \\ 60 \\ \underline{56} \\ 40 \\ \underline{35} \\ 50 \\ \underline{49} \\ 10 \\ \underline{7} \\ 30 \end{array}$$

定義 整数 a, N に対し、

$$a \bmod N$$

で、

a を N で割ったあまり ($0, 1, \dots, N - 1$ のいずれか)

を表す。先の余りの列は

$$x_1 = 1$$

$$x_2 = x_1 \times 10 \bmod 7 = 10^1 \bmod 7$$

$$x_3 = x_2 \times 10 \bmod 7 = 10^2 \bmod 7$$

$\dots$

$$x_{n+1} = x_n \times 10 \bmod 7 = 10^n \bmod 7$$

定理

N を 10 と互いに素な自然数とする。

$1/N$ の小数展開の周期は $N - 1$ 以下で、

$1 = 10^P \pmod{N}$ となる最小の自然数 $P \geq 1$ となる。

証明：余りの種類は N 種。そのうち 0 は出てこないから、周期は $N - 1$ 以下。

（余りに 0 が小数点 n 桁目に出てきたら、 10^n を N が割り切ることが筆算の形から分かる）。

後半については、循環するのは

$10^n \pmod{N}$ が 1 となったとき。

筆算を見よ。

$$\begin{array}{r} 0.\dot{1}42857\dot{1} \\ 7 \overline{) 10} \\ \underline{7} \\ 30 \\ \underline{28} \\ 20 \\ \underline{14} \\ 60 \\ \underline{56} \\ 40 \\ \underline{35} \\ 50 \\ \underline{49} \\ 10 \\ \underline{7} \\ 30 \end{array}$$

$1 + 1 = 0$ の世界での多項式

二元体 $\mathbb{F}_2$

$\mathbb{F}_2 := \{0, 1\}$ とおく。

$0, 1$ の掛け算は普通に定義して、 $\mathbb{F}_2$ からはみ出ない。

足し算は $1 + 1 = 2$ だけが $\mathbb{F}_2$ からはみ出してしまっているので、

$$1 + 1 = 0$$

と定義する（2で割ったあまりを見ている）。

$1 + 1 = 0$ から 1 を移項して

$$1 = -1.$$

$\mathbb{F}_2$ 多項式 $\mathbb{F}_2[t]$

$$\mathbb{F}_2[t] := \left\{ \sum_{i=0}^n a_i t^i \mid a_i \in \mathbb{F}_2, n \in \mathbb{N} \right\}$$

を考える。係数が0または1の多項式のことである。
掛け算・足し算は通常が多項式同様

$$\begin{aligned} (t+1) \times (t+1) &= t(t+1) + 1(t+1) \\ &= t^2 + t + t + 1 = t^2 + 1 \end{aligned}$$

といった具合に計算できる。

($1+1=0$ より $t+t=(1+1)t=0$ 。)

係数のみを表記することにして、「 t 進くらい取り」で

$$t^3 + t^2 + 1 = 1t^3 + 1t^2 + 0t + 1 = 1101$$

と表わすことにする。

$\mathbb{F}_2$ 多項式での和差積商は、

「繰り上がり・繰り下がりのない世界での計算」になる。

$\begin{array}{r} 1\ 1 \\ \times 1\ 1 \\ \hline 1\ 1 \\ 1\ 1 \\ \hline 1\ 0\ 1 \end{array}$	$\begin{array}{r} t\ +1 \\ \times t\ +1 \\ \hline t\ +1 \\ t^2 + t \\ \hline t^2 + 0t + 1 \end{array}$
---	--

形式べき級数 ($1 + 1 = 0$ での無限小数)

$\mathbb{F}_2[t]$ の世界で $1 \div (t^3 + t^2 + 1) = 1 \div 1101$ を小数展開すると

$$\begin{array}{r}
 \overline{) 0.00111010} \\
 1101 \overline{) 0001} \\
 \underline{0000} \\
 0010 \\
 \underline{0000} \\
 0100 \\
 \underline{0000} \\
 1000 \\
 \underline{1101} \\
 1010 \\
 \underline{1101} \\
 1110 \\
 \underline{1101} \\
 0110 \\
 \underline{0000} \\
 1100 \\
 \underline{1101} \\
 0010 \\
 \underline{0000} \\
 0100
 \end{array}$$

検算：

$$\begin{array}{r}
 0.00111010011101 \dots \\
 \times 1101 \\
 \hline
 0.00111010011101 \dots \\
 00.0000000000000000 \dots \\
 000.11101001110100 \dots \\
 0001.11010011101001 \dots \\
 \hline
 0001.0000000000000000 \dots
 \end{array}$$

$1 \div 1101 = 0.00111010011101 \dots$ は次の省略形である。

$$1 \div (t^3 + t^2 + 1) = \\ 0 + 0t^{-1} + 0t^{-2} + 1t^{-3} + 1t^{-4} + 1t^{-5} + 0t^{-6} + 1t^{-7} \dots$$

この無限小数のような式を $\mathbb{F}_2$ 形式べき級数と言い、
右辺を左辺の形式べき級数展開という。

定理：定数項が1で次数 n の $\mathbb{F}_2$ 多項式 $f(t) = t^n + \dots + 1$ に
対し、 $1/f(t)$ のべき級数展開の係数は循環し、
周期は $2^n - 1$ 以下。

周期は $1 = t^P \pmod{f(t)}$ となる最小の自然数 $P \geq 1$ 。

証明：余りの種類が 2^n で、そのうち0はあらわれないから。
後半は、筆算を見よ。

$1 \div 1101 = 0.x_1x_2x_3x_4 \cdots$ とおくと $\mathbb{F}_2$ での漸化式

$$x_{n+3} + x_{n+2} + x_n = 0 \quad (n \geq 1) \quad \text{を満たす}$$

理由：

$$\begin{array}{cccccccccccc}
& & & & 0 & . & x_1 & x_2 & x_3 & x_4 & x_5 & \cdots \\
\times & 1 & 1 & 0 & 1 & & & & & & & \\
\hline
& & & & 0 & . & x_1 & x_2 & x_3 & x_4 & x_5 & \cdots \\
& & & 0 & 0 & . & 0 & 0 & 0 & 0 & 0 & \cdots \\
& & 0 & x_1 & x_2 & . & x_3 & x_4 & x_5 & x_6 & x_7 & \cdots \\
& 0 & x_1 & x_2 & x_3 & . & x_4 & x_5 & x_6 & x_7 & x_8 & \cdots \\
\hline
& 0 & 0 & 0 & 1 & . & 0 & 0 & 0 & 0 & 0 & \cdots
\end{array}$$

したがって、漸化式

$$x_{n+3} = x_{n+2} + x_n \quad (n \geq 1), \quad x_1 = 0, x_2 = 0, x_3 = 1$$

を解くことで上の循環「小数」が得られる。

$$x_{n+3} = x_{n+2} + x_n \quad (n \geq 1), \quad x_1 = 0, x_2 = 0, x_3 = 1$$

で $x_1x_2x_3x_4 \cdots$ を計算すると

0011

00111

001110

0011101

00111010

001110100

0011101001

00111010011

001110100111

周期の最大性と均等分布性

上の循環「小数」を三つずつ組にしてみると、000以外の $2^3 - 1$ 通りのパターンを一回ずつ一周期にとる。

00111010011101

001, 011, 111, 110, 101, 010, 100, 001

証明：漸化式が3階だから、連続する3個の数のパターンにより以後の数列は決まってしまう。000はあらわれないから、周期が $2^3 - 1$ ならば他の全パターンが一個ずつ現れる。

定理

1. 任意の自然数 n に対し、次数 n の $\mathbb{F}_2$ 多項式 $f(t)$ であって $1/f(t)$ のべき級数展開の係数の周期が $2^n - 1$ となるものがたくさん存在する。
2. このとき、連続する n 個の 0-1 の並びは、
 $000 \dots 0$ を除いてすべて一回ずつ一周期に現れる。

性質1を満たす $f(t)$ を $\mathbb{F}_2$ 原始多項式という。

このとき $f(t)$ で割り切れない任意の $\mathbb{F}_2$ 多項式 $g(t)$ に対して、 $g(t)/f(t)$ の「小数部分」は周期 $2^n - 1$ となる。

性質2は均等分布性と呼ばれ、数列のバランスの良さを示している。疑似乱数に用いるのに適している。

原始多項式の例：

$$t^3 + t^2 + 1 (\text{上でみた、周期 } 2^3 - 1 = 7)$$

$$t^{31} + t^3 + 1 (\text{周期 } 2^{31} - 1 = 2147483647)$$

$$t^{607} + t^{273} + 1 (\text{周期 } 2^{607} - 1 = 5.3 \times 10^{182})$$

など多数が知られている。

したがって、たとえば

$$x_{n+607} = x_{n+273} + x_n$$

を使って周期が $2^{607} - 1$ で、連続する 607 項が均等分布する数列を、きわめて高速に作り出すことができる。

実験では確かめられないが、証明できる。数学の強み。

整数の割り算に比べてはるかに速い（桁数に無関係に高速）

応用：ストリーム暗号

暗号化：絵に「パスワード」をかけて、パスワードを知る人だけが読めるようにしたい。

- ・絵のデータは膨大。一方、パスワードは短くしたい。

原始多項式 $f(t)$ と、任意の多項式 $g(t) \neq 0$ を用意して「パスワード」とする。

(例: $f(t) = t^{31} + t^3 + 1$, $g(t)$:30次以下の任意の多項式。)

1. 送りたい絵を用意する。
2. $g(t)/f(t)$ のべき級数展開の係数を、絵に合わせて長方形にならべる。
3. $1 + 1 = 0$ の約束で、場所ごとに足す。暗号化文。
4. 暗号化文に、先の同じ係数を足す。復号。

111 111
1 1 1
1 1
1 1
1 1
1 1
1 1
1

絵

000000000000

00111011100

01000100010

01000000010

00100000100

00010001000

00001010000

00000100000

0-1 化

11110111011
10000100000
10011100101
11011110010
10011100000
11100000100
10101110000
11101010111

$g(t)/f(t)$ の展開
=パスワードの展開

11110111011
10111111100
11011000111
10011110000
10111100100
11110001100
10100100000
11101110111

絵＋展開
＝暗号化文
赤字は展開の
反転したもの

11110111011
10111111100
11011000111
10011110000
10111100100
11110001100
10100100000
11101110111

絵十展開
=暗号化文

11110111011
10000100000
10011100101
11011110010
10011100000
11100000100
10101110000
11101010111

$g(t)/f(t)$ の展開
= パスワードの展開
を再度加える

000000000000

00111011100

01000100010

01000000010

00100000100

00010001000

00001010000

00000100000

復元（復号）

普通の幾何：互除法

問い:比 $1 : 0.384615384615 \dots$ を既約整数比であらわせ。

たて $x_1 = 1$

よこ $x_2 = 0.384615384615 \dots$

$x_1 \div x_2 = 2$ 余り $0.230769230769 \dots$

$x_1 = x_2 \times 2 + 0.230769230769$

x1=1

x2=0.384615...

たて $x_1 = 1$

よこ $x_2 = 0.384615384615 \dots$

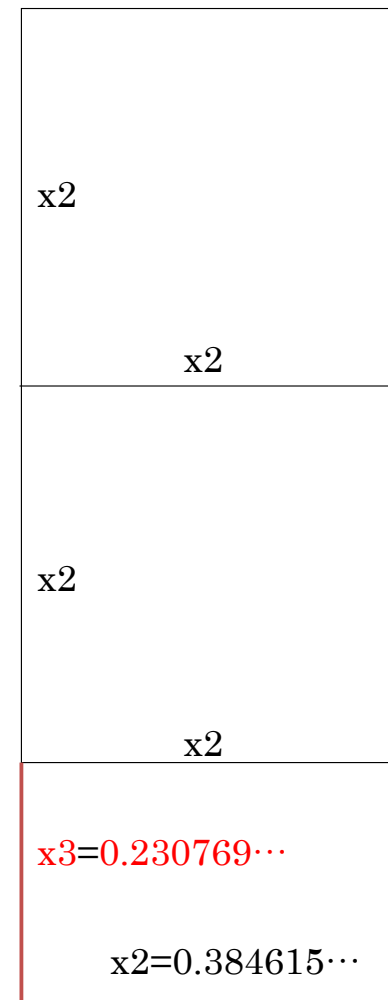
$x_1 \div x_2 = 2$ 余り $0.230769230769 \dots$

$x_1 = x_2 \times 2 + 0.230769230769$

$x_3 = 0.230769230769$ とおいて、

$x_2 \div x_3$

を計算



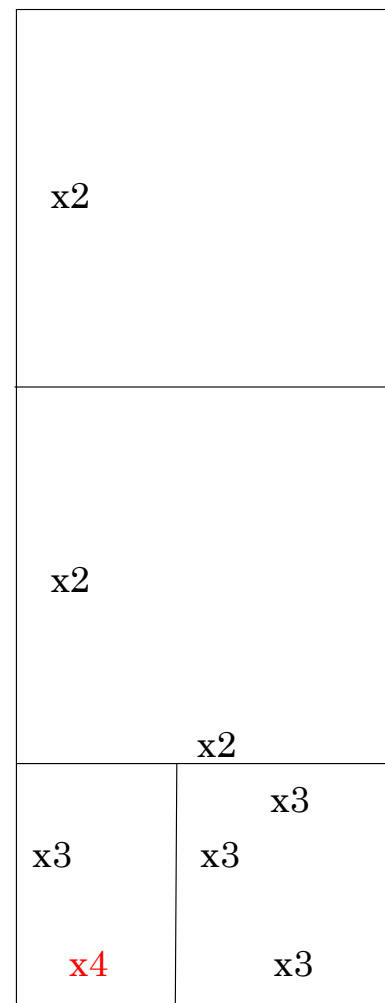
$$x_1 = x_2 \times 2 + x_3$$

$$x_2 = x_3 \times 1 + 0.153846153846 \dots$$

$$x_4 = 0.153846153846 \dots \text{ において、}$$

$$x_3 \div x_4$$

を計算



$$x_1 = x_2 \times 2 + x_3$$

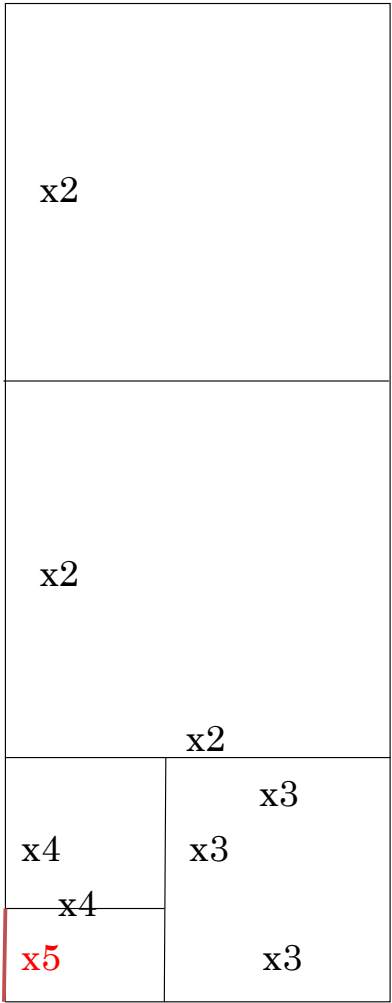
$$x_2 = x_3 \times 1 + x_4$$

$$x_3 = x_4 \times 1 + 0.076923076923 \dots\dots$$

$$x_5 = 0.076923076923 \dots \text{ とおいて、}$$

$$x_4 \div x_5$$

を計算



$$x_1 = x_2 \times 2 + x_3$$

$$x_2 = x_3 \times 1 + x_4$$

$$x_3 = x_4 \times 1 + x_5$$

$$x_4 = x_5 \times 2.$$

$\therefore$

$$x_4 = x_5 \times 2 = 2x_5$$

$$x_3 = x_4 \times 1 + x_5 = 3x_5$$

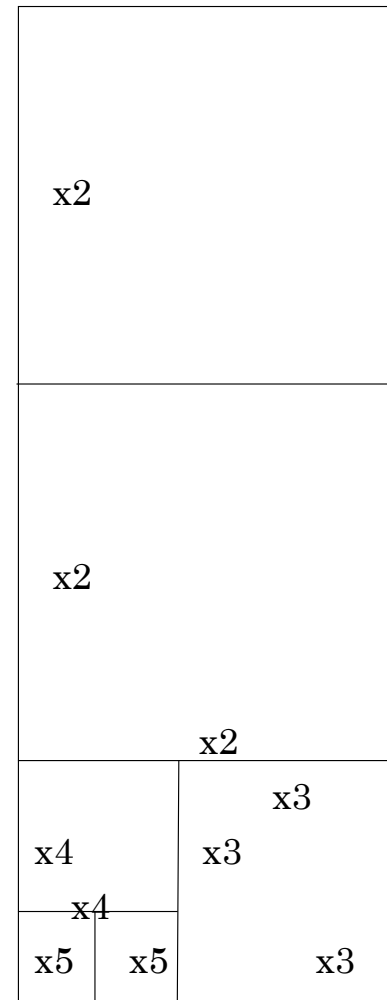
$$x_2 = x_3 \times 1 + x_4 = 5x_5$$

$$x_1 = x_2 \times 2 + x_3 = 13x_5.$$

最初は $x_1 = 1$,

$x_2 = 0.384615384615 \dots$ だから

$x_2 = x_2/x_1 = (5x_5/13x_5) = 5/13$ (答).



$1 + 1 = 0$ での互除法

問い: (先に見たような) $\mathbb{F}_2$ べき級数

$0.00101110010111\dots$

を $\mathbb{F}_2$ 多項式/ $\mathbb{F}_2$ 多項式 $= g(t)/f(t)$ の形で表わせ。

解答: 互除法。

整数が $\mathbb{F}_2$ 多項式に、大きさが「次数」に置き換わる。

$x_1 = 1, x_2 = 0.100101110010111 \dots$ とおく。

$$10 \times x_2 = 1.00101110010111 \dots$$

$$x_1 = 1.0000000000000000$$

$$10 \times x_2 = 1.00101110010111 \dots$$

$$\text{差} \quad 0.00101110010111 \dots$$

$$x_1 = 10 \times x_2 + 0.00101110010111 \dots = 10 \times x_2 + x_3,$$

$$x_3 = 0.00101110010111 \dots \text{ とおく。}$$

$$x_2 = 0.100101110010111 \dots$$

$$100 \times x_3 = 0.101110010111 \dots$$

$$\text{差} \quad 0.00101110010111 \dots$$

$$x_2 - 100 \times x_3 = 0.00101110010111 \dots = x_3$$

$$x_2 = 100 \times x_3 + x_3 = 101 \times x_3$$

$$\therefore x_1 = 10 \times x_2 + x_3 = 1010 \times x_3 + x_3 = 1011x_3$$

$$x_2 = x_2/x_1 = 101/1011 = (t^2 + 1)/(t^3 + t + 1)$$

定理 (Berlekamp-Massey, 1969)

べき級数 x がある未知の $\mathbb{F}_2[t]$ 多項式の商 $g(t)/f(t)$ で表わされたとする。 $f(t)$ の次数が N 以下だと分かっているとする。 x が小数点以下 $2N - 1$ 桁まで与えられれば、上の互除法により $g(t), f(t)$ は正確に求まる。(余りが小数点以下 $2N$ 桁まで0になったら、それを0として逆算すれば良い。)

応用:暗号解読

先に見た暗号で、 $f(t)$ の次数が N 以下とする。送信されたメッセージの最初の $2N - 1$ 個が0であるとわかっているとすると、暗号化されたメッセージの最初の $2N - 1$ 個からパスワード $g(t), f(t)$ が互除法で求まる。

変形版：メッセージの最初の $2N - 1$ 個の0-1が分かれば、残りのメッセージは解読できる。

$\therefore$ 暗号化文 - メッセージ = $g(t)/f(t)$ だから、その最初の $2N - 1$ 個が特定できればよい。

$1 + 1 = 0$ の応用例：メルセンヌツイスター疑似乱数

メルセンヌツイスター疑似乱数 (松本-西村 1998)

$\mathbb{F}_2$ 成分の 32 次元ベクトルの列を次の漸化式で生成し、

$$\vec{x}_{n+624} = \vec{x}_{n+q} + B\vec{x}_{n+1} + C\vec{x}_n$$

$T\vec{x}_n$ を出力列とする。ここに B, C, T はうまく選ばれた
3 2 次正方行列。

1. 周期は $2^{19937} - 1 > 10^{6000}$
2. 出力列は、623 次元空間内で均等分布している
3. 高位ビットはさらに高次元
(例えば上位 3 ビットは 6240 次元まで均等分布)
4. それまでの生成法よりも数倍高速に生成

高次元の互除法 (格子簡約) など、 $1 + 1 = 0$ の世界
での代数・幾何を利用して周期や均等分布の次元を求めた。

[ウェブ](#) [画像](#) [動画](#) [地図](#) [ニュース](#) [ショッピング](#) [Gmail](#) [もっと見る ▼](#)[ウェブ履歴](#) | [検索設定](#) | [ログイン](#)

mersenne twister

検索

セーフサーチ: オフ ▼

約 48,600 件 (0.10 秒)

[検索オプション](#)[すべて
もっと見る](#)[広島県広島市
場所を変更](#)

[Mersenne Twister: A random number generator \(since 1997/10\)](#) 🔍

News: SIMD-oriented Fast **Mersenne Twister** (SFMT) をリリースしました。SFMTはオリジナルの**Mersenne Twister**より約二倍速 ... 山本秀樹さんが Windows上で走るMT を書いてくれました。山本さんより: "MTをWindows上のいろいろな言語から利用できる ...

www.math.sci.hiroshima-u.ac.jp/~m-mat/MT/mt.html - [キャッシュ](#) - [類似ページ](#)

[Mersenne Twister: A random number generator \(since 1997/10\)](#) 🔍

- [[このページを訳す](#)]

News: we released SIMD-oriented Fast **Mersenne Twister** (SFMT). SFMT is ...

www.math.sci.hiroshima-u.ac.jp/~m-mat/.../emt.html - [キャッシュ](#) - [類似ページ](#)

[メルセンヌ・ツイスタ - Wikipedia](#) 🔍

メルセンヌ・ツイスタ(**Mersenne twister**、通称MT)は1996年から1998年(1996年国際会議で発表、1997年朝日新聞記事、1998年1月論文掲載)に松本眞と西村拓士によって開発された疑似乱数生成器である。既存の乱数生成アルゴリズムの欠点を改良して、高品質 ...

[長所](#) - [短所](#) - [余談](#) - [参照](#)

ja.wikipedia.org/wiki/メルセンヌ・ツイスタ - [キャッシュ](#) - [類似ページ](#)

Mersenne Twister 🔍

Mersenne Twister(以下MT)は、松本眞氏・西村拓士氏により96年から97年に渡って開発された疑似乱数生成アルゴリズムです。... リンク. **Mersenne Twister** Home Page MTの開発者によるページです。一次情報はここを参照してください。

www.emit.jp/mt/mt.html - キャッシュ - 類似ページ

Mersenne Twister 🔍

1998年11月28日 ... 慶応義塾大学の松本先生のページで公開されている疑似乱数生成アルゴリズム**Mersenne Twister**を、Windows上のDLLとして実現してみました。このアルゴリズムの詳細については、**Mersenne Twister** Home Pageを参照してください。...

www.asahi-net.or.jp/~jz6h-ymmt/toolbox/mt.htm - キャッシュ - 類似ページ

Freeware C++ Mersenne Twister pseudo-random number generator 🔍

- [[このページを訳す](#)]

Freeware C++ implementation of the famous **Mersenne Twister** pseudo-random number generator algorithm, known as one of the best generators, with a period of $2^{19937}-1$.
Written in 100% standard ISO C++. [Open Source]

www.bedaux.net/mtrand/ - キャッシュ - 類似ページ

Mersenne Twister Random Number Generator 🔍 - [[このページを訳す](#)]

30 Sep 2009 ... C++ class implementing the **Mersenne Twister** random number generator.

www-personal.umich.edu/.../MersenneTwister.html - キャッシュ - 類似ページ

Mersenne Twister C# Version 🔍

Mersenne Twister とは、松本 眞さん、西村 拓士さんによって開発された疑似乱数生成アルゴリズムです。非常にすぐれたアルゴリズムとして、様々な所で利用されています。**Mersenne Twister** に関する詳しい情報、オリジナルの C ソースコード等は ...

takel.jp/mt/ - キャッシュ - 類似ページ

Excel Random Generator based on Mersenne Twister - NtRand 🔍

今回新しくなった、**Mersenne Twister**アルゴリズムに基づく乱数生成フリー Excel アドイン “NtRand 3.0” を自信を持ってお届けします。他では見られない様々な確率分布と統計の実用関数に対応していて、かつモンテカルロ VaR シミュレーション もカバーし ...

www.ntrand.com/jp/ - キャッシュ

Mersenne Twister DLL 🔍

Mersenne Twister DLL. == 2002-01-13. presented by Einguste == == 2002-01-22.
BCC55(Borland C++ Compiler 5.5)用のMakefileを追加 == ... **Mersenne Twister**の詳細はこちら(松本 眞先生のページ) DLL版アーカイブ(ソース含む) ...

hp.vector.co.jp/authors/VA020429/mt.html - キャッシュ - 類似ページ

1 2 3 4 5 6 7 8 9 10

[次へ](#)

mersenne twister

検索

[絞り込み検索](#)

[ヘルプ](#)

[ご意見をお寄せください](#)

[ウェブ](#) [画像](#) [動画](#) [地図](#) [ニュース](#) [ショッピング](#) [Gmail](#) [もっと見る ▼](#)[ウェブ履歴](#) | [検索設定](#) | [ログイン](#)[すべて
もっと見る](#)

mersenne twister

検索

セーフサーチ: オフ ▼

約 36,600 件 (0.16 秒)

[検索オプション](#)

英語のページを検索

[Mersenne twister - Wikipedia, the free encyclopedia](#) 🔍 - [[このページを訳す](#)]

The **Mersenne twister** is a pseudorandom number generator developed in 1997 by Makoto Matsumoto (松本 眞) and Takuji Nishimura (西村 拓士) that is based on a matrix linear recurrence over a finite binary field F_2

en.wikipedia.org/wiki/Mersenne_twister - [キャッシュ](#) - [類似ページ](#)[Mersenne Twister: A random number generator \(since 1997/10\)](#) 🔍- [[このページを訳す](#)]

News: we released SIMD-oriented Fast **Mersenne Twister** (SFMT). SFMT is roughly twice faster than the original **Mersenne Twister**, and has a better equidistribution property, as well as a quicker recovery from zero-excess initial state. ...

www.math.sci.hiroshima-u.ac.jp/~m-mat/.../emt.html - [キャッシュ](#) - [類似ページ](#)[SIMD-oriented Fast Mersenne Twister \(SFMT\)](#) 🔍 - [[このページを訳す](#)]

SFMT is a new variant of **Mersenne Twister** (MT) introduced by Mutsuo Saito ...

www.math.sci.hiroshima-u.ac.jp/~m-mat/MT/SFMT/ - [キャッシュ](#) - [類似ページ](#)

[Freeware C++ Mersenne Twister pseudo-random number generator](#) 🔍

- [[このページを訳す](#)]

Freeware C++ implementation of the famous **Mersenne Twister** pseudo-random number generator algorithm, known as one of the best generators, with a period of $2^{19937}-1$.

Written in 100% standard ISO C++. [Open Source]

www.bedaux.net/mtrand/ - [キャッシュ](#) - [類似ページ](#)

[Mersenne Twister Random Number Generator](#) 🔍 - [[このページを訳す](#)]

30 Sep 2009 ... C++ class implementing the **Mersenne Twister** random number generator.

www-personal.umich.edu/.../MersenneTwister.html - [キャッシュ](#) - [類似ページ](#)

[PDF]

[Mersenne Twister: A 623-Dimensionally Equidistributed Uniform ...](#) 🔍

- [[このページを訳す](#)]

ファイルタイプ: PDF/Adobe Acrobat - [クイックビュー](#)

M MATSUMOTO 著 - 1998 - [引用元 1757](#) - [関連記事](#)

A new algorithm called **Mersenne Twister** (MT) is proposed for generating uniform pseudorandom numbers. For a particular choice of parameters, ... tion method, k-distribution, Mersenne Primes, **Mersenne Twister**, m-sequences, MT19937, ...

saluc.engr.uconn.edu/refs/crypto/.../matsumoto98mersenne.pdf - [類似ページ](#)

[PDF] [Parallel Mersenne Twister](#) 🔍 - [[このページを訳す](#)]

ファイルタイプ: PDF/Adobe Acrobat - [クイックビュー](#)

V Podlozhnyuk 著 - 2007 - [引用元 16](#) - [関連記事](#)

The **Mersenne Twister** (MT) is a pseudorandom number generator algorithm developed by Makoto Matsumoto and Takuji Nishimura [1]. and to enable efficient implementation of **Mersenne twister** on parallel architectures, dcmt, a ...

[developer.download.nvidia.com/.../MersenneTwister/.../MersenneTwister.pdf](#) - [アメリカ合衆国](#)

[Excel Random Generator based on Mersenne Twister - NtRand](#) 🔍

- [[このページを訳す](#)]

We proudly offer this NEW Excel Add-in Random Number Generator based on **Mersenne Twister** algorithm including probability distributions and statistic utility functions. This generator covers Monte Carlo VaR calculation as well. ...

[www.ntrand.com/](#) - [キャッシュ](#)

[PDF] [Mersenne Twister Random Number Generator](#) 🔍 - [[このページを訳す](#)]

ファイルタイプ: PDF/Adobe Acrobat - [クイックビュー](#)

Mersenne Twister Random Number Generator. **Mersenne Twister** is a pseudorandom number generator that has generated a great deal of excitement in the scientific world. It is essentially a large linear-feedback shift register, and ...

[wundt.uni-graz.at/projects/pksirt/mtinfo.pdf](#) - [類似ページ](#)

まとめ：数学の予期せぬ効用

- $1 + 1 = 0$ の数学の研究は、ガロア (1830 ごろ) に遡る
- 当時は応用の見えなかった純粋数学が、
現在実用されている。
- $\mathbb{F}_2[t]$ と整数は良く似ており、代数・幾何が展開できる。
前者が扱いやすい (現代整数論の指導原理の一つ)

終わり

周期保証：Mersenne素数の利用

定理 $f(t)$ を定数項が1の n 次 $\mathbb{F}_2$ 多項式とする ($n \geq 2$)。

$2^n - 1$ が素数とする。

$1/f(t)$ のべき級数展開の周期が最大値 $2^n - 1$ を満たす必要十分条件は、

$$t = t^{2^n} \pmod{f(t)}$$

となること。

n 回 ($\pmod{f(t)}$) で二乗すればよい。

n が 1 0 0 万くらいまでなら計算機でチェックできる。

必要性：周期が $2^n - 1$ ならば $1 = t^{2^n - 1} \pmod{f(t)}$ 。

十分性： $t = t^{2^n} \pmod{f(t)}$ と $f(t)$ が t と互いに素なことから

$$1 = t^{2^n - 1} \pmod{f(t)}.$$

周期を P とすると

$$1 = t^P \pmod{f(t)}.$$

$2^n - 1$ を P で割ったあまりを $r < P$ とすると

$$1 = t^{2^n - 1} \pmod{f(t)} = t^{Pq+r} \pmod{f(t)} = t^r \pmod{f(t)}.$$

周期 P の最小性から $r = 0$ 。

すなわち P は $2^n - 1$ の約数である。

$2^n - 1$ が素数であることを使うと $P = 1$ または $P = 2^n - 1$ 。

$n \geq 2$ より $P = 1$ は不可能。

よって $P = 2^n - 1$ 。

注 $2^n - 1$ が素数となるとき、メルセンヌ素数という。
2010年11月現在で47個知られており、既知の最大は

$$2^{43,112,609} - 1。$$

メルセンヌツイスターに使った $2^{19937} - 1$ は
24番目のメルセンヌ素数。