

LISP-STATによる乱数生成 (v00-20)

上條哲男 (上智大学経済学部)

2004 年 4 月 17 日

1 はじめに

このメモの主要な目的は、乱数表（一様乱数表）および正規乱数表を示することである。

使用するコンピュータ・プログラミング言語は LISP である (栗原 (1993), Steele(1990), 竹内 (1986), Winston and Horn(1989), および湯浅, 萩谷 (1986) などを参照)。プログラムの作成および実行には、LISP-STAT を利用している (竹村 (1997), 垂水 (1999), および Tierney(1990) などを参照)。

ミネソタ大学の School of Statistics の Luke Tierney が LISP-STAT を考案し、コンピュータ・ソフトウェアである XLISP-STAT を開発した。XLISP-STAT は、LISP 言語の方言の 1 つである XLISP 言語を使用している

使用する乱数生成法は、Lehmer 生成法である (伏見 (1989), Knuth(1981), Law and Kelton(1991), および Park and Miller(1988) などを参照)。具体的な漸化式は、

$$z_i = 16807z_{i-1} \pmod{2147483647} \quad (\text{ただし、} i = 1, 2, \dots) \quad (1)$$

である。この z_i を種子 (seed) と呼び、 z_0 を種子の初期値と言う。ここでは、種子の初期値 z_0 を

$$z_0 = 123456789$$

とし、乱数の最大桁数を 9 桁としている。9 桁の乱数 y_i は、まず、 $x_i = z_i/2147483647$ を、つぎに $y_i = 10^9 x_i$ を計算し、さらに、この y_i の小数部分を切り捨てたものを、改めて y_i とすることにより求められる (上條 (1998) および上條 (2000) を参照)。

上條(1998)においては、Lotus 1-2-3による手順 および HP-BASIC(HP: Hewlett Packard) によるプログラム、さらに、それらによって生成された乱数を提示している。上條(2000)においては、LISP および Schemeによる乱数生成のための関数、さらに、それらによって生成された乱数を提示している。

2 プログラム・ファイルのロード

プログラムが保存されているフロッピーディスクのフォルダ book-stat からプログラム・ファイルを XLISP-STAT にロードする。手順はつぎのとおりである (なお、そのフロッピーディスクの内容を消去してしまうことがないように、そのフロッピーディスクをライトプロテクトしておくこと)。

- (1) XLISP-STAT を起動する。
- (2) フロッピーディスクをフロッピードライブ (a:とする) に挿入する。
- (3) XLISP-STAT において、フロッピードライブ (a:) に挿入されているフロッピーディスク内のフォルダ book-stat 内のサブフォルダ programs 内のサブサブフォルダ random-number 内のサブサブサブフォルダ random-number-seed 内にあるプログラム・ファイル

random-number-seed-v01-01.lsp

(v01-01 は version 01.01 を示しており、変更されている場合がある)。

およびフォルダ book-stat 内のサブフォルダ programs 内のサブサブフォルダ random-number 内のサブサブサブフォルダ table-for-random 内にあるプログラム・ファイル

t-random-number-1-v01-01.lsp

(v01-01 は version 01.01 を示しており、変更されている場合がある)。

t-random-number-2-v01-01.lsp

(v01-01 は version 01.01 を示しており、変更されている場合がある)。

t-random-number-3-v01-01.lsp

(v01-01 は version 01.01 を示しており、変更されている場合がある)。

t-random-number-9-v01-01.lsp

(v01-01 は version 01.01 を示しており、変更されている場合がある)。

をロードする。

3 の「乱数の生成」および 4 の「乱数表の作成」においてはここでロードしたプログラム・ファイル内のプログラムを使用する。

3 乱数の生成

3.1 1個の乱数の生成

種子の値が与えられた場合における 1 個の乱数を求める関数は (random-number seed) である。seed の値を 123456789 として、関数 (random-number seed) を実行すると、

```
> (random-number 123456789)
218418296
```

となる。この乱数を [0 1) の一様乱数として使用する場合には、1000000000(= 10^9) で除すことになる。すなわち、

```
> (/ 218418296 1000000000)
0.218418296
```

となる。乱数「218418296」のつぎの乱数を生成するための seed は *seed* と定義されており、その値は、

```
> *seed*
469049721
```

である。

3.2 第 n 番目の乱数の生成

第 n 番目の乱数を生成するためには 2 つの方法がある。

(1) 第 n 番目の乱数を生成するための第 1 の方法

第 n 番目の乱数を生成するための第 1 の方法は、関数 (random-n-th n seed) を使用する方法である。種子の初期値を 123456789 として、第 10 番目の乱数を生成するための方法はつぎのとおりである。

```
> (random-n-th 10 123456789)
(10 633965712)
```

(2) 第 n 番目の乱数を生成するための第 2 の方法

第 n 番目の乱数を生成するための第 2 の方法は、関数 (seed-n-th n seed) および関数 (random-number seed) を使用する方法である。種子の初期値を 123456789 として、第 10 番目の乱数を生成するためには、まず、関数 (seed-n-th n seed) によって、第 9 番目の種子の値をつぎのとおり求める。

```
> (seed-n-th 9 123456789)
(9 94122056)
```

つぎに、得られた種子の値、この場合 94122056 を種子として関数 (random-number seed) によって第 10 番目の乱数をつぎのとおり生成する。

```
> (random-number 94122056)
633965712
```

3.3 n 個の乱数の生成

n 個の乱数を生成するためには、関数 (random-number-list n seed) を使用する。種子の初期値を 123456789 として、10 個の乱数を求めるための方法はつぎのとおりである。

```
> (random-number-list 10 123456789)
(218418296 956317576 829509233 561695442 415307081
 66118734 257577792 109956793 43828997 633965712)
```

4 乱数表の作成

9 桁、1 桁、2 桁、および 3 桁の乱数表を作成する。

(1) 9 桁の乱数表を作成する。

9 桁の乱数表を作成するには、関数 (table-random-number-9 no-within-class name seed) を使用する。no-within-class は数字のみの場合にはそのまま、文字を含む場合には、最初に quote(') を付ける。name は、最初に quote(') を付ける。seed は数字である。例えば、no-within-class が 123456789 で、name が tatsuo kamijo で、そして seed(種子) が 123456789 の場合には、

```
> (table-random-number-9 123456789 'tetsuo-kamijo 123456789)
```

とすれば、付録 A.1 の「9 桁の乱数表」にあるとおりの表が出力される。

1 つの乱数表における乱数の個数は 200 個であり、つぎの 200 個からなる乱数表を作成する場合には、seed の値を *seed* とする。その *seed* の値は、作成された表の下端に表示されている「seed for next table」の値である。第 1 表の場合には、

```
*seed* = 597239251
```

となっている。

(2) 1 桁の乱数表を作成する。

1 桁の乱数表を作成するには、関数 (table-random-number-1 no-within-class name seed) を使用する。no-within-class は数字のみの場合にはそのまま、文字を含む場合には、最初に quote(') を付ける。name は、最初に quote(') を付ける。seed は数字である。例えば、no-within-class が 123456789 で、name が tatsuo kamijo で、そして seed (種子) が 123456789 の場合には、

```
> (table-random-number-1 123456789 'tetsuo-kamijo 123456789)
```

とすれば、付録 A.2 の「1 桁の乱数表」にあるとおりの表が出力される。

乱数の個数は 400 個であり、つぎの 400 個からなる乱数表を作成する場合には、seed の値を *seed* とする。その *seed* の値は、作成された表の下端に表示されている「seed for next table」の値である。この場合には、

```
*seed* = 1349476249
```

である。

(3) 2 桁の乱数表を作成する。

2 桁の乱数表を作成するには、関数 (table-random-number-2 no-within-class name seed) を使用する。no-within-class は数字のみの場合にはそのまま、文字を含む場合には、最初に quote(') を付ける。name は、最初に

quote(')を付ける。seedは数字である。例えば、no-within-classが123456789で、nameがtetsuo kamijoで、そしてseed(種子)が123456789の場合には、

```
> (table-random-number-2 123456789 'tetsuo-kamijo 123456789)
```

とすれば、付録A.3の「2桁の乱数表」にあるとおりの表が出力される。

乱数の個数は400個であり、つぎの400個からなる乱数表を作成する場合には、seedの値を*seed*とする。その*seed*の値は、作成された表の下端に表示されている「seed for next table」の値である。この場合には、

```
*seed* = 1349476249
```

である。

(4) 3桁の乱数表を作成する。

3桁の乱数表を作成するには、関数(table-random-number-3 no-within-class name seed)を使用する。no-within-classは数字のみの場合にはそのまま、文字を含む場合には、最初にquote(')を付ける。nameは、最初にquote(')を付ける。seedは数字である。例えば、no-within-classが123456789で、nameがtetsuo kamijoで、そしてseed(種子)が123456789の場合には、

```
> (table-random-number-3 123456789 'tetsuo-kamijo 123456789)
```

とすれば、付録A.4の「3桁の乱数表」にあるとおりの表が出力される。

乱数の個数は400個であり、つぎの400個からなる乱数表を作成する場合には、seedの値を*seed*とする。その*seed*の値は、作成された表の下端に表示されている「seed for next table」の値である。この場合には、

```
*seed* = 1349476249
```

である。

5 各ストリームに対する種子の初期値および乱数

乱数生成法で生成される乱数のある長さのリストをストリームとよぶことにする。

乱数を使用する分析において、異なるストリームを使用するのが望ましい場合がある。例えば、消費者行動をモンテカルロ法で分析する場合を考えてみる。消費者が異なれば、異なるストリームを使用するのがよいであろう。ここでは、各ストリームの大きさ（リストの長さ）を、25 万としている。第 0 番目のストリームをストリーム 0 とし、このストリームを参照用ストリームとよぶことにする。

- (1) XLISP-STAT を起動する。
- (2) フロッピーディスクをフロッピードライブ (a:とする) に挿入する。
- (3) XLISP-STAT において、フロッピードライブ (a:) に挿入されているフロッピーディスク内のフォルダ book-stat 内のサブフォルダ programs 内のサブフォルダ random-number 内のサブサブフォルダ random-number-seed 内にあるプログラム・ファイル

random-number-seed-v01-01.lsp

(v01-01 は version 01.01 を示しており、変更されている場合がある)。

およびフォルダ book-stat 内のサブフォルダ programs 内のサブサブフォルダ random-number 内のサブサブサブフォルダ table-for-random 内にあるプログラム・ファイル

table-random-number-for-stream-v01-01.lsp

(v01-01 は version 01.01 を示しており、変更されている場合がある)。

table-seed-for-stream-v01-01.lsp

(v01-01 は version 01.01 を示しており、変更されている場合がある)。

をロードする（なお、すでにロードされているプログラム・ファイルを再度ロードする必要はない）。

5.1 各ストリームに対する種子の初期値

ストリーム 0 (参照用ストリーム) に対する「種子の初期値」 z_0 を

$$z_0 = 123456789$$

とする。ストリーム 0 に対する「種子の初期値」が 123456789 である場合のストリーム 1 に対する「種子の初期値」は、つぎのように求められる。すなわち、

```
> (seed-stream-next 123456789)
1145470673
```

である。ストリーム 0 に対する「種子の初期値」が 123456789 である場合のストリーム 10 に対する「種子の初期値」は、つぎのように求められる。すなわち、

```
> (seed-stream-n-th 10 123456789)
694143352
```

である。ストリーム 0 に対する「種子の初期値」が 123456789 である場合のストリーム 1 からストリーム 10 までのそれぞれに対する「種子の初期値」は、つぎのように求められる。すなわち、

```
> (seed-stream-list 10 123456789)
(1145470673 2022640624 12040923 880431333 369265507
 333469926 1178903832 845941495 1224334554 694143352)
```

である。

ストリームに対する「種子の初期値」の表を作成するには、関数 (table-seed-for-stream no-within-class name seed) を使用する。no-within-class は数字の場合にはそのまま、文字を含む場合には、最初に quote(') を付ける。name は、最初に quote(') を付ける。seed は数字である。例えば、no-within-class が 123456789 で、name が tetsuo kamijo で、そして seed(種子) が 123456789 の場合には、

```
(table-seed-for-stream 123456789 'tetsuo-kamijo 123456789)
```

とすれば、付録 A.5 の「ストリームに対する seed の表」にあるとおりの表が出力される。seed の個数は 200 個であり、つぎの 200 個からなる表を作成する場合には、seed の値を *seed-st-next* とする。その *seed-st-next* の値は、作成さ

れた表の下端に表示されている「seed for next table」の値である。この場合には、

```
*seed-st-next* = 2089487851
```

である。

5.2 各ストリームに対する第1番目の乱数

ストリーム0（参照用ストリーム）に対する「種子の初期値」 z_0 を

$$z_0 = 123456789$$

とする。ストリーム0に対する「種子の初期値」が123456789である場合のストリーム1における第1番目（最初）の乱数は、つぎの2つの方法で求められる。

第1の方法は、ストリーム1の「種子の初期値」1145470673を使用して、

```
> (random-number 1145470673)
876462740
```

のように求められる。第2の方法は、ストリーム0の「種子の初期値」123456789および関数 (seed-stream-next seed) を使用して、

```
> (random-number (seed-stream-next 123456789))
876462740
```

のように求められる。ストリーム0に対する「種子の初期値」が123456789である場合のストリーム1からストリーム10までのそれぞれに対する第1番目の乱数は、つぎのように求められる。すなわち、

```
> (random-number (seed-stream-list 10 123456789))
(876462740 932402539 236709622 580719626 8212551
 858777789 536710583 650511740 94316678 622191727)
```

である。

各ストリームに対する第1番目の乱数の表を作成するには、関数 (table-random-number-for-stream no-within-class name seed) を使用する。no-within-class は数字の場合はそのまま、文字を含む場合には、最初に quote(') を付ける。name

は、最初に `quote(')` を付ける。seed は数字である。例えば、no-within-class が 123456789 で、name が `tetsuo kamijo` で、そして seed(種子) が 123456789 の場合には、

```
(table-random-number-for-stream 123456789 'tetsuo-kamijo 123456789)
```

とすれば、付録 A.6 の「ストリームに関する乱数表」にあるとおりの表が出力される。乱数の個数は 200 個であり、つぎの 200 個からなる表を作成する場合には、seed の値を `*seed-st-next*` とする。その `*seed-st-next*` の値は、作成された表の下端に表示されている「seed for next table」の値である。この場合には、

```
*seed-st-next* = 2089487851
```

である。

6 「つぎの種子の値」の計算と乱数の生成

- (1) XLISP-STAT を起動する。
- (2) フロッピーディスクをフロッピードライブ (a:) に挿入する。
- (3) XLISP-STAT において、フロッピードライブ (a:) に挿入されているフロッピーディスク内のフォルダ `book-stat` 内のサブフォルダ `programs` 内のサブサブフォルダ `random-number` 内のサブサブサブフォルダ `random-number-seed` 内にあるプログラム・ファイル

```
random-number-seed-v01-01.lsp
```

(v01-01 は version 01.01 を示しており、変更されている場合がある)。

をロードする (なお、すでにロードされているプログラム・ファイルを再度ロードする必要はない)。

6.1 「つぎの種子の値」の計算

種子の値 z_{i-1} (ただし、 $i = 1, 2, \dots$) が与えられた際に、つぎの種子の値 z_i は、式 (1) で計算される。例えば、種子の初期値 z_0 を

$$z_0 = 123456789$$

とすれば、つぎの種子の値 z_1 は、

$$z_1 = 469049721$$

となる。XLISP-STAT で、つぎの種子を計算する関数は (seed-next seed) である。この関数を XLISP-STAT で実行することにする。種子の値を、123456789 とすると、

```
> (seed-next 123456789)
469049721
```

となり、つぎの種子の値である 469049721 が得られる。

6.2 「つぎの種子の値」からの乱数の生成

種子の値が与えられると、「つぎの種子の値」が計算され、その「つぎの種子の値」から乱数を求めることになる。したがって、種子の値が、123456789 の場合には、「つぎの種子の値」は 469049721 であるので、この値を用いて乱数を求めることになる。

「つぎの種子の値」が与えられた場合における乱数を求める関数は (random-n-for-seed-next seed-next) である。seed-next の値を 469049721 として、関数 (random-n-for-seed-next seed-next) を実行すると、

```
> (random-n-for-seed-next 469049721)
218418296
```

となる。この値「218418296」は、

```
> (random-number 123456789)
218418296
```

と同じ値である。

7 正規乱数の生成と正規乱数表の作成

正規乱数の生成には、Box and Muller Approachを使用している (Rubinstein and Melamed(1998, p.36) を参照)。このアプローチでは、2つの一様乱数から、2つの独立な正規乱数が生成される。

- (1) XLISP-STAT を起動する。
- (2) フロッピーディスクをフロッピードライブ (a:とする) に挿入する。
- (3) XLISP-STAT において、フロッピードライブ (a:) に挿入されているフロッピーディスク内のフォルダ book-stat 内のサブフォルダ programs 内のサブフォルダ random-number 内のサブサブフォルダ random-number-seed 内にあるプログラム・ファイル

random-number-seed-v01-01.lsp

(v01-01 は version 01.01 を示しており、変更されている場合がある)。

およびフォルダ book-stat 内のサブフォルダ programs 内のサブサブフォルダ normal-random-number 内にあるプログラム・ファイル

normal-random-list-generation-fun-v01-02.lsp

(v01-02 は version 01.02 を示しており、変更されている場合がある)。

normal-random-v01-01.lsp

(v01-01 は version 01.01 を示しており、変更されている場合がある)。

table-normal-random-v01-02.lsp

(v01-02 は version 01.02 を示しており、変更されている場合がある)。

をロードする (なお、すでにロードされているプログラム・ファイルを再度ロードする必要はない)。

ストリーム 64 (第 64 番目のストリーム) の乱数を使用して正規乱数を生成させることにする。

ストリーム 0 (第 0 番目のストリーム、参照用ストリーム) の種子の初期値を「123456789」とした場合のストリーム 64 の第 1 番目の乱数のための種子の値は、

```
> (seed-stream-n-th 64 123456789)
1208899767
```

となる。上記の種子の値「1208899767」から 8 個の正規乱数を生成させると、

```
> (normal-random-list-generation-fun 8 1208899767)
( -1.4434101  0.59290252 -0.24855668      -0.22411394
  -0.94916356 1.61563327   0.9988176600000001 0.30132173 )
```

となる。

正規乱数表を作成するには、関数 (table-normal-random seed) を使用する。seed は数字である。例えば、seed(種子) が 1208899767 の場合には、

```
(table-normal-random 1208899767)
```

とすれば、付録 A.7 の「正規乱数表」にあるとおりの表が出力される。

1 つの正規乱数表における正規乱数の個数は 160 個であり、つぎの 160 個からなる表を作成する場合には、seed の値を *seed-for-normal* とする。その *seed-for-normal* の値は、作成された表の下端に表示されている「seed for next table」の値である。第 1 表の場合には、

```
*seed-for-normal* = 1296113863
```

となっている。

付録

- A 乱数表 (一様乱数表), ストリームに関する表,
および正規乱数表

A.1 9桁の乱数表

A.1.1 9桁の乱数表 (seed=123456789)

XLISP-STATにおいて、下記のように、促進記号(>)の後に、
(table-random-number-9 123456789 'tetsuo-kamijo 123456789)
と入力することによって出力される。

```
> (table-random-number-9 123456789 'tetsuo-kamijo 123456789)
```

Table of Random Number(9 digits)

No within class: 123456789

Name TETSUO-KAMIJO

Initial seed: 123456789

If a digit number of random number is n(<=8),

then add (9-n) number of zero(s) before the

random number.

For example, random number is 66118734(8 digits),

then the number becomes 066118734.

i	i	i+1	i+2	i+3	i+4
1	218418296	956317576	829509233	561695442	415307081
6	66118734	257577792	109956793	43828997	633965712
11	61727229	449538960	401306281	754673486	797286954
16	1838371	897504060	350752337	94544750	13616891
21	859096855	840847450	123103915	7512364	260302997
26	912483707	113664046	351628659	822887316	267132270
31	692066499	561662474	861215790	453793775	911977028
36	597916877	188954691	761492056	396988475	185314117
41	574365861	367026667	617204827	361528704	212929997
46	714471214	117706867	299329148	825002954	824660073
51	61861770	710780524	88283333	777994008	745303068
56	308674919	899373090	763536724	761730646	406969640
61	938749454	562088285	17820021	501103225	41909255
66	368850581	271723601	858572561	29036558	17442279
71	152383787	114318671	353907259	119307827	206652759
76	212923957	612947552	809519074	587089634	215491643
81	768056363	723296722	448018990	855176118	945017496
86	909056924	519725721	30194625	481066955	292312883
91	902639843	667841511	412278035	156948490	833281632
96	964404164	740789914	456098861	653561422	406826949
101	540538601	832280939	145755764	717128318	775650756
106	362261623	531110845	379976752	269285263	877417819
111	761284802	913675004	135794085	291195228	118210970
116	771780843	320632915	877412951	679475764	949175543
121	793361180	21366233	102278301	991406155	563259625
126	704527526	994145554	604334274	46145179	562029006
131	21515388	609134812	728788505	748406594	469633795
136	135193808	202347838	860120225	40624381	773972507
141	155940722	895727204	487127308	148681700	893344749
146	445207315	599347217	228688778	572299614	639618372
151	65984537	2119914	629405178	412841577	628397089
156	469879128	258508626	754493277	768513356	403978118
161	660235871	584298490	304735190	684344665	780800314
166	910878278	131230770	595554847	490322732	854158896
171	848579945	83142974	383971625	411112335	565026143
176	394391505	538036249	775243041	509805754	305324214
181	584075669	559778367	195029220	856103379	529502248
186	344292860	530100924	406241019	692813821	121905097
191	858970808	722382340	79995959	492084645	466643562
196	878359268	584231199	173777874	684731249	278111198

seed for next table: 597239251

A.1.2 9桁の乱数表 (seed=597239251)

XLISP-STATにおいて、下記のように、促進記号(>)の後に、
(table-random-number-9 123456789 'tetsuo-kamijo 597239251)
と入力することによって出力される。

```
> (table-random-number-9 123456789 'tetsuo-kamijo 597239251)
```

Table of Random Number(9 digits)

No within class: 123456789

Name TETSUO-KAMIJO

Initial seed: 597239251

If a digit number of random number is n(<=8),

then add (9-n) number of zero(s) before the

random number.

For example, random number is 66118734(8 digits),

then the number becomes 066118734.

i	i	i+1	i+2	i+3	i+4
1	214914548	68821287	679375320	261012041	829383123
6	442158119	351522149	32761876	628850283	86720829
11	516987624	11002505	919104950	396910289	871227586
16	722047682	455403510	966804220	78531013	870749579
21	688185721	337421092	36307152	214310010	908339139
26	455909700	474341528	258065220	302161857	434341128
31	971341430	335417308	358696723	615840092	424428302
36	366480565	438869409	78172746	849352383	65508649
41	3867933	8358812	486560497	622276337	598398912
46	290530399	944417488	824733175	290476522	38914391
51	34182799	510303879	677300980	397581448	151404905
56	662249580	428704500	236546015	628882122	621841001
61	281714125	769302268	663225084	824001108	986633583
66	350636731	151554665	179256892	770598071	441795871
71	263209495	761996179	869792357	600144544	629358237
76	623896598	830125529	919769935	573309951	620358706
81	368776366	24396000	23587244	430811725	652666979
86	373921223	493997631	618198075	55056938	341972549
91	532633387	969347606	825220744	485052976	285376329
96	319971142	754983931	14931680	956748716	75678434
101	927452403	692546508	629162171	328611647	975965637
106	54465039	393913715	507813549	822332110	935783507
111	713414555	358430625	143523989	207696195	749954791
116	490182810	502491461	373998686	795929810	192331267
121	511614727	708718278	428111376	267911656	791217256
126	988433455	601080061	352596690	92578524	967257928
131	704005200	215408898	377365330	379108199	671502188
136	937284244	936300015	394360726	20736720	522060143
141	264824629	907554470	267984437	14439640	687031291
146	934912534	74965666	947949859	193285077	542296443
151	376331832	9115563	205279270	128703466	119156605
156	665070647	842375616	806986004	13777462	557808982
161	95568341	217119173	121952603	657409174	75990918
166	179360974	519893344	847437114	875589274	28932785
171	273328063	824758294	712662185	713357986	407683287
176	933020357	273155942	931928731	926181979	340536325
181	394020951	310124647	264950235	18615835	876344810
186	727232561	597665926	971230380	469008246	621593706
191	125418987	916917594	634004223	708991445	19231764
196	228267777	496533951	246122652	583425263	628398847

seed for next table: 1349476249

A.1.3 9桁の乱数表 (seed=1349476249)

XLISP-STATにおいて、下記のように、促進記号(>)の後に、
(table-random-number-9 123456789 'tetsuo-kamijo 1349476249)
と入力することによって出力される。

```
> (table-random-number-9 123456789 'tetsuo-kamijo 1349476249)
```

Table of Random Number(9 digits)

No within class: 123456789

Name TETSUO-KAMIJO

Initial seed: 1349476249

If a digit number of random number is n(<=8),

then add (9-n) number of zero(s) before the
random number.

For example, random number is 66118734(8 digits),
then the number becomes 066118734.

i	i	i+1	i+2	i+3	i+4
1	499431498	945196655	920189885	631411499	133066586
6	450123446	224769316	697906085	707573104	181175386
11	14722880	447451698	320701767	34612083	725292825
16	996509979	343217250	452322758	188609586	961315013
21	821431058	791798122	751046469	838009847	431502775
26	267139834	819205868	393030860	669674848	225186871
31	715748789	589898777	428745095	918817582	567101291
36	271408675	565614683	285990911	649252422	985464222
41	697191296	694111944	939451202	356364148	412249467
46	676808562	121516837	333492821	13846543	718850862
51	726439380	266666806	869023447	677079269	671282690
56	248176393	100640759	469242976	566702096	562128406
61	692124344	533854064	485258934	746916842	431364559
66	944148216	299080366	643718219	972117858	384847141
71	125899630	995092458	518943108	876822782	760499718
76	718776445	475722961	475819223	93694103	716796846
81	204606326	818527673	994612754	456569683	566678158
86	159813894	992124844	642266628	575227409	847077669
91	834391363	615651711	258309750	411976188	83802020
96	460553774	527293477	221475551	339601444	681473318
101	522061324	284687955	750460777	994292168	68474461
106	850279843	653327435	474201618	906601271	247563090
111	792866341	704604502	287867872	195327703	872719984
116	804786348	44161332	219512404	344982608	122699078
121	203407804	674971515	246263144	944672174	105232793
126	647559028	524598487	926786086	493763429	681956962
131	650661486	667595845	283383016	818360105	178293776
136	583506365	991486276	909841241	701742075	179066465
141	570082627	378712604	22751058	377047404	35725024
146	430487535	204003485	686580467	357922705	606903893
151	233741248	489165236	400137175	105516900	422552135
156	833749243	823542431	277652036	497777925	153587235
161	340662163	508979847	424303516	269193610	337016168
166	230746888	162949933	699528648	977993290	133239503
171	356328275	809329459	400223377	554310899	303279834
176	224178021	760013705	550340065	565473422	911807012
181	740454675	821739104	969121394	23272301	137564272
186	42723170	48334445	357023185	488682916	293771729
191	421465124	564350253	34706101	305444604	607473872
196	813373749	372608235	426610340	39993841	176489379

seed for next table: 379008056

A.1.4 9桁の乱数表 (seed=379008056)

XLISP-STATにおいて、下記のように、促進記号(>)の後に、
(table-random-number-9 123456789 'tetsuo-kamijo 379008056)
と入力することによって出力される。

> (table-random-number-9 123456789 'tetsuo-kamijo 379008056)

Table of Random Number(9 digits)

No within class: 123456789

Name TETSUO-KAMIJO

Initial seed: 379008056

If a digit number of random number is $n(<=8)$,

then add (9-n) number of zero(s) before the

random number.

For example, random number is 66118734(8 digits),

then the number becomes 066118734.

i	i	i+1	i+2	i+3	i+4
1	256998553	374681287	268393196	884450055	952076727
6	553551301	536718465	627251973	223921434	447543368
11	861392993	432048159	433414817	402830297	368806972
16	538784120	344712953	590612993	432587648	500600137
21	586519045	625598772	438573109	98246160	223219098
26	643390627	466274910	682426211	537340746	85934443
31	300190884	308200156	920037444	69324300	133513310
36	958217747	765682064	818463412	914581623	373343715
41	787820566	900258590	646129202	493512538	465234426
46	195014558	609682317	930704772	355118227	472047182
51	697003859	543873095	875118968	124495492	395748168
56	339474366	545672800	122752631	103471574	46759132
61	880736862	544443898	468604142	829823504	843641618
66	84684811	297620263	103761741	923581129	628049149
71	622047323	749359205	480170437	224540682	855252695
76	232048268	35241491	303750049	127089172	987724911
81	692586165	295687082	612801072	347625055	534300167
86	982920685	947957995	330027401	770530881	312521224
91	544220530	714451336	783606990	82691105	789403770
96	509178393	761263882	562075365	800661753	722095616
101	261034309	203640101	579187126	398029515	682070455
106	558143333	715011039	190535001	321763148	873232057
111	411194179	940573151	212959972	218254604	205129643
116	613918843	133994867	51732295	464685709	972726251
121	610108147	87637711	927014779	337402941	731243105
126	2865977	168485298	732417456	740194439	447941337
131	550066433	966542317	676738324	941026430	831218246
136	285073255	226202188	780184315	557797009	894336735
141	117510436	997908324	845217158	564782576	300762106
146	908721237	877836996	806402403	205192355	667911643
151	590999683	931682944	795252104	802123023	281649324
156	680192548	996164869	542969770	692925248	994645691
161	10144434	497509101	635463393	233256054	334503266
166	996394768	406871752	293544893	609029346	956225437
171	280921201	442631130	301414994	881815305	669845521
176	93676426	419707869	30168411	40497150	635604940
181	612237518	875965144	346182392	287472523	550701070
186	632897496	108216154	788907664	171113285	900994128
191	8309898	664463017	629934202	304144902	763372896
196	8265239	913882899	629900120	731325109	381120909

seed for next table: 818450921

A.1.5 9桁の乱数表 (seed=818450921)

XLISP-STATにおいて、下記のように、促進記号(>)の後に、
(table-random-number-9 123456789 'tetsuo-kamijo 818450921)
と入力することによって出力される。

```
> (table-random-number-9 123456789 'tetsuo-kamijo 818450921)
```

Table of Random Number(9 digits)

No within class: 123456789

Name TETSUO-KAMIJO

Initial seed: 818450921

If a digit number of random number is n(<=8),

then add (9-n) number of zero(s) before the

random number.

For example, random number is 66118734(8 digits),

then the number becomes 066118734.

i	i	i+1	i+2	i+3	i+4
1	499128462	852077290	863026686	889518570	138615555
6	711636153	468837703	755279881	988965010	534927529
11	526988843	101496087	844749337	702114616	440357532
16	89041863	526601686	594543116	486158075	858767155
21	299574529	949116062	793659903	41997765	856447718
26	316804913	540188782	952861908	750093397	819725099
31	119745639	564966005	383647221	958848712	370302842
36	679868369	547681382	880998142	935772731	532304544
41	442475204	680763074	584993354	983313836	555651991
46	843027642	765585637	197802345	464026300	890039534
51	894452254	59036990	234701940	635520691	196259486
56	533183130	208869578	471004405	171039511	661070272
61	608072164	868872980	148183702	523488929	278442123
66	776762994	55654169	379619831	270514100	530486268
71	882718109	843258054	638127931	16140101	266685658
76	185866180	852894457	597146434	240117182	649480153
81	812942282	120944923	721326103	327818233	641053332
86	183364034	799323641	232436572	561470810	639913508
91	26345200	783782841	38217713	325104791	36236159
96	21133058	183309984	890909134	509829006	696118310
101	660451022	200331532	972066553	522569081	818553169
106	423114298	282008310	713673659	713186835	531140398
111	876677277	315009769	369190926	991907324	986397367
116	380554397	977763281	267467018	318177263	605273800
121	836760845	439529057	164866564	912355806	964034333
126	525048317	487066617	128647516	178802686	136749669
131	351703238	76335138	964667915	173653289	590840938
136	263658011	300198523	436576091	534376356	263424265
141	371627251	939213019	353215404	491303179	332532414
146	872288653	555392683	484835656	632880233	818084819
151	551563408	126214193	281953593	794051791	628466648
156	638966515	110225876	566312127	7919817	108368273
161	345571189	14979378	758411546	622855727	336210163
166	684209558	510057537	537026420	803049103	846278577
171	404059965	35837664	323631474	274190800	324782190
176	614284063	272263235	928199709	252515207	23094604
181	151013936	91229950	301779498	8031065	978119194
186	249309744	148873060	109528585	846942444	561660689
191	831201664	6376245	165566410	674660950	26599215
196	53012487	980885293	739124927	472649477	819770734

seed for next table: 1760444247

A.2 1桁の乱数表

XLISP-STAT において、下記のように、促進記号 (>) の後に、
 (table-random-number-1 123456789 'tetsuo-kamijo 123456789)
 と入力することによって出力される。

```
> (table-random-number-1 123456789 'tetsuo-kamijo 123456789)
```

```
Table of Random Number(1 digit)
No within class: 123456789
Name            TETSUO-KAMIJO
Initial seed:   123456789
```

i	i	i+1	i+2	i+3	i+4	i+5	i+6	i+7	i+8	i+9
1	2	9	8	5	4	0	2	1	0	6
11	0	4	4	7	7	0	8	3	0	0
21	8	8	1	0	2	9	1	3	8	2
31	6	5	8	4	9	5	1	7	3	1
41	5	5	6	3	2	7	1	2	8	8
51	0	7	0	7	7	3	8	7	7	4
61	9	5	0	5	0	3	2	8	0	0
71	1	1	3	1	2	2	6	8	5	2
81	7	7	4	8	9	9	5	0	4	2
91	9	6	4	1	8	9	7	4	6	4
101	5	8	1	7	7	3	5	3	6	8
111	7	9	1	2	1	7	3	8	2	9
121	7	0	1	9	5	7	9	6	0	5
131	0	6	7	7	4	1	2	8	0	7
141	1	8	4	1	8	4	5	2	5	6
151	0	0	6	4	6	4	2	7	7	4
161	8	6	3	6	7	9	1	5	4	8
171	8	0	3	4	5	3	5	5	5	3
181	5	5	1	8	5	3	5	4	6	1
191	8	7	0	4	4	8	5	1	6	2
201	2	0	6	2	8	4	3	0	6	0
211	5	0	9	3	8	7	4	9	0	8
221	6	0	0	2	9	4	4	2	3	4
231	9	3	3	6	4	3	4	0	8	0
241	0	0	4	6	5	2	9	8	2	0
251	0	5	4	3	1	6	4	2	6	6
261	2	7	6	8	9	3	1	2	7	4
271	2	7	8	6	6	6	8	9	5	6
281	3	0	0	4	6	6	4	9	0	3
291	5	9	8	4	2	3	7	0	9	0
301	9	6	6	3	9	0	3	5	8	9
311	7	3	1	2	7	4	5	3	7	1
321	5	7	4	2	7	9	6	3	0	9
331	7	2	3	3	6	9	9	3	0	5
341	2	9	3	0	6	9	0	3	1	5
351	3	0	2	1	1	6	8	8	0	5
361	0	2	1	6	0	1	5	8	8	0
371	2	8	7	7	4	9	2	9	9	3
381	3	3	2	0	8	7	5	9	4	6
391	1	9	6	7	0	2	4	2	5	6

```
seed for next table: 1349476249
```

A.3 2桁の乱数表

XLISP-STAT において、下記のように、促進記号 (>) の後に、
 (table-random-number-2 123456789 'tetsuo-kamijo 123456789)
 と入力することによって出力される。

```
> (table-random-number-2 123456789 'tetsuo-kamijo 123456789)
```

```
Table of Random Number(2 digits)
```

```
No within class: 123456789
```

```
Name TETSUO-KAMIJO
```

```
Initial seed: 123456789
```

```
If a digit number of random number is 1,  

then add one zero before the random number.  

For example, random number is 6, then  

the number becomes 06.
```

i	i	i+1	i+2	i+3	i+4	i+5	i+6	i+7	i+8	i+9
1	21	95	82	56	41	6	25	10	4	63
11	6	44	40	75	79	0	89	35	9	1
21	85	84	12	0	26	91	11	35	82	26
31	69	56	86	45	91	59	18	76	39	18
41	57	36	61	36	21	71	11	29	82	82
51	6	71	8	77	74	30	89	76	76	40
61	93	56	1	50	4	36	27	85	2	1
71	15	11	35	11	20	21	61	80	58	21
81	76	72	44	85	94	90	51	3	48	29
91	90	66	41	15	83	96	74	45	65	40
101	54	83	14	71	77	36	53	37	26	87
111	76	91	13	29	11	77	32	87	67	94
121	79	2	10	99	56	70	99	60	4	56
131	2	60	72	74	46	13	20	86	4	77
141	15	89	48	14	89	44	59	22	57	63
151	6	0	62	41	62	46	25	75	76	40
161	66	58	30	68	78	91	13	59	49	85
171	84	8	38	41	56	39	53	77	50	30
181	58	55	19	85	52	34	53	40	69	12
191	85	72	7	49	46	87	58	17	68	27
201	21	6	67	26	82	44	35	3	62	8
211	51	1	91	39	87	72	45	96	7	87
221	68	33	3	21	90	45	47	25	30	43
231	97	33	35	61	42	36	43	7	84	6
241	0	0	48	62	59	29	94	82	29	3
251	3	51	67	39	15	66	42	23	62	62
261	28	76	66	82	98	35	15	17	77	44
271	26	76	86	60	62	62	83	91	57	62
281	36	2	2	43	65	37	49	61	5	34
291	53	96	82	48	28	31	75	1	95	7
301	92	69	62	32	97	5	39	50	82	93
311	71	35	14	20	74	49	50	37	79	19
321	51	70	42	26	79	98	60	35	9	96
331	70	21	37	37	67	93	93	39	2	52
341	26	90	26	1	68	93	7	94	19	54
351	37	0	20	12	11	66	84	80	1	55
361	9	21	12	65	7	17	51	84	87	2
371	27	82	71	71	40	93	27	93	92	34
381	39	31	26	1	87	72	59	97	46	62
391	12	91	63	70	1	22	49	24	58	62

```
seed for next table: 1349476249
```

A.4 3桁の乱数表

XLISP-STATにおいて、下記のように、促進記号(>)の後に、
 (table-random-number-3 123456789 'tetsuo-kamijo 123456789)
 と入力することによって出力される。

```
> (table-random-number-3 123456789 'tetsuo-kamijo 123456789)
```

Table of Random Number(3 digits)

No within class: 123456789
 Name: TETSUO-KAMIJO
 Initial seed: 123456789

If a digit number of random number is $n(\leq 2)$,
 then add $(3-n)$ zero(s) before the random number.
 For example, random number is 61, then
 the number becomes 061.

i	i	i+1	i+2	i+3	i+4	i+5	i+6	i+7	i+8	i+9
1	218	956	829	561	415	66	257	109	43	633
11	61	449	401	754	797	1	897	350	94	13
21	859	840	123	7	260	912	113	351	822	267
31	692	561	861	453	911	597	188	761	396	185
41	574	367	617	361	212	714	117	299	825	824
51	61	710	88	777	745	308	899	763	761	406
61	938	562	17	501	41	368	271	858	29	17
71	152	114	353	119	206	212	612	809	587	215
81	768	723	448	855	945	909	519	30	481	292
91	902	667	412	156	833	964	740	456	653	406
101	540	832	145	717	775	362	531	379	269	877
111	761	913	135	291	118	771	320	877	679	949
121	793	21	102	991	563	704	994	604	46	562
131	21	609	728	748	469	135	202	860	40	773
141	155	895	487	148	893	445	599	228	572	639
151	65	2	629	412	628	469	258	754	768	403
161	660	584	304	684	780	910	131	595	490	854
171	848	83	383	411	565	394	538	775	509	305
181	584	559	195	856	529	344	530	406	692	121
191	858	722	79	492	466	878	584	173	684	278
201	214	68	679	261	829	442	351	32	628	86
211	516	11	919	396	871	722	455	966	78	870
221	688	337	36	214	908	455	474	258	302	434
231	971	335	358	615	424	366	438	78	849	65
241	3	8	486	622	598	290	944	824	290	38
251	34	510	677	397	151	662	428	236	628	621
261	281	769	663	824	986	350	151	179	770	441
271	263	761	869	600	629	623	830	919	573	620
281	368	24	23	430	652	373	493	618	55	341
291	532	969	825	485	285	319	754	14	956	75
301	927	692	629	328	975	54	393	507	822	935
311	713	358	143	207	749	490	502	373	795	192
321	511	708	428	267	791	988	601	352	92	967
331	704	215	377	379	671	937	936	394	20	522
341	264	907	267	14	687	934	74	947	193	542
351	376	9	205	128	119	665	842	806	13	557
361	95	217	121	657	75	179	519	847	875	28
371	273	824	712	713	407	933	273	931	926	340
381	394	310	264	18	876	727	597	971	469	621
391	125	916	634	708	19	228	496	246	583	628

seed for next table: 1349476249

A.5 ストリームに対する seed の表

XLISP-STAT において、下記のように、促進記号 (>) の後に、
 (table-seed-for-stream 123456789 'tetsuo-kamijo 123456789)
 と入力することによって出力される。

```
> (table-seed-for-stream 123456789 'tetsuo-kamijo 123456789)
```

```
Table of initail seed for each stream
                                (stream1, stream2, ... )
No within class:                123456789
Name                            TETSUO-KAMIJO
Initail seed for stream:        123456789
```

i	i	i+1	i+2	i+3	i+4
1	1145470673	2022640624	12040923	880431333	369265507
6	333469926	1178903832	845941495	1224334554	694143352
11	341757915	233211304	1132000895	1791985719	661124451
16	1989617657	767404073	48696603	1305794881	465185814
21	1140633399	2039301772	1010065390	280672924	201419047
26	1371062036	501208371	294923811	1533686112	943194238
31	1687078546	969688974	1052101280	1365840818	922673144
36	798989604	1582362874	2242891	875680037	379880543
41	732104911	1054842401	1289224409	130022074	1976655396
46	1820501599	367174665	1958997525	1131349664	443957597
51	1783614091	1074191695	1974827082	599757102	1048182371
56	680854387	224172772	661967464	1578258001	751282651
61	1848171435	303064085	474694396	1208899767	1387801234
66	1199064061	2132315592	695831691	790461616	1369358939
71	1488328708	1667008051	346441664	697615678	510000699
76	1682546364	220826869	841008278	1445911115	1984522335
81	1800747260	1642535927	218653776	287570376	409894048
86	977199702	1693763582	1137852001	969203113	2110330124
91	923492547	1597983496	1321511680	321075038	1857459189
96	2015817872	138101768	779753942	1443711280	1479672206
101	1394554139	445294232	92981725	1468443024	836542373
106	2078069499	1750781628	1657203843	977193171	861235581
111	579535316	326324124	1131761471	1982713178	228076621
116	680973716	2014234602	1943988254	209336570	1451006002
121	1088084408	2117943436	2133463996	1251441372	1682697484
126	743511537	1453310679	241092947	1618747820	1684961022
131	1440102298	1815086916	1522634581	2085821130	1736238587
136	1807193097	1782511027	1454134070	920415717	770906592
141	513892316	565154126	1324244920	725422944	735351923
146	1811055760	422528874	1822111098	1932405246	1857708918
151	1247693781	470585328	217428503	820574403	136518944
156	939566271	641793778	108748777	763009118	1084841188
161	20191559	1965996604	61841737	1988336409	1120162999
166	1191607694	549738589	229735215	864513847	533583337
171	2111715659	1763201387	1099287452	485700468	229573026
176	2072973152	1872531665	1953383632	163675234	1143606610
181	1587093387	2040114981	228692323	548108569	1469780362
186	504416603	1654813980	544252510	479799440	127216550
191	872336204	1980873641	392004185	1274573441	1245174123
196	1195919839	818178121	489610357	1482474325	2089487851

seed for next table: 2089487851

A.6 ストリームに関する乱数表

XLISP-STAT において、下記のように、促進記号 (>) の後に、
 (table-random-number-for-stream 123456789 'tetsuo-kamijo 123456789)
 と入力することによって出力される。

```
> (table-random-number-for-stream 123456789 'tetsuo-kamijo 123456789)
```

```
Table of initail random number for each stream
      (stream1, stream2, ... )
No within class:      123456789
Name                  TETSUO-KAMIJO
Initial seed for stream: 123456789
```

i	i	i+1	i+2	i+3	i+4
1	876462740	932402539	236709622	580719626	8212551
6	858777789	536710583	650511740	94316678	622191727
11	723640121	197780576	457006270	741944511	204079961
16	482468755	987646485	117596757	632915773	715954610
21	18170207	328699072	144718305	649944346	379837703
26	437771315	641787361	182648283	193840564	786389950
31	699670650	143977317	133115594	574591263	180730699
36	187675346	156154329	553693174	395322669	85403989
41	723370124	586140725	946283098	600763000	35027732
46	917751148	644510995	838008950	360231991	576089649
51	222492480	20860478	725967245	920546214	462286666
56	617844561	458891581	801810989	29902468	815444926
61	472105032	891438479	133628476	295974175	444915970
66	318107092	289200622	835756176	444492218	108705309
71	210048211	620565561	380389313	797897192	453792939
76	229140767	272619196	41333861	234302297	604597287
81	312999658	92686574	265190943	631950647	982176888
86	925708052	19230899	249922402	341449722	222809713
91	593681149	408910163	638388323	851807372	161497427
96	535026021	835431747	642281584	17581278	461067064
101	295644074	37088724	709314228	577342251	89512254
106	739245923	263513346	889213457	874594035	347676261
111	657381893	933092682	583185636	445466557	11855388
116	551757379	143770364	369911790	345296321	108769294
121	750362431	807400712	277034385	242283794	411861690
126	995837205	146022520	882429065	918177210	127145911
131	772355495	540442568	700478076	406372492	444304325
136	760500206	589506299	590184531	513273420	399653517
141	911024597	104878563	29733838	427829003	135685026
146	991127169	866990693	514284645	717014047	115968796
151	912252794	974544995	675752002	118282710	447666650
156	392580547	915095029	108085312	591105776	367725121
161	26596657	615385680	996269406	454948794	809260919
166	961825616	457193669	992624483	4596508	20226038
171	66518245	465133393	429521605	271212043	723273479
176	853352427	124260257	901609349	982819906	287617379
181	178895854	693172992	830566593	700055249	43843264
186	750595663	185263139	521113629	87587905	643696210
191	225250869	48570728	969503050	282397522	192478879
196	710264686	364094929	871447121	391485067	103485009

```
seed for next table:      2089487851
```


A.7 正規乱数表

A.7.1 正規乱数表

XLISP-STATにおいて、第0番目のストリームのためのseedの値を「123456789」とした場合の第64番目のストリームの第1番目の乱数のためのseedの値は、関数に引数を代入して、(seed-stream-n-th 64 123456789)と実行すると得られ、その値は「1208899767」である。この値を、関数に代入して、(table-normal-random 1208899767)と実行すると、該当する正規乱数表が出力される。

```
> (seed-stream-n-th 64 123456789)
1208899767
> (table-normal-random 1208899767)
```

Table of Normal Random Number
seed: 1208899767

i	i	i+1	i+2	i+3
1	-1.44341010	0.59290252	-0.24855668	-0.22411394
5	-0.94916356	1.61563327	0.99881766	0.30132173
9	0.56785394	0.52938106	1.00113997	1.79256375
13	-0.99877261	1.18672364	-0.63502809	1.84594329
17	-2.59516518	-0.27790881	-0.35284283	0.41604301
21	1.60276894	-0.47043800	-0.60321147	-1.38821740
25	-0.67161019	0.22048143	1.64530575	2.59140758
29	0.02217095	1.06122543	-0.44862022	-0.26008676
33	-0.44865663	-0.18404038	-0.72488110	0.04536503
37	0.25653353	1.49931712	1.07721367	-0.62524935
41	1.52340786	0.71729802	-1.50928411	1.41460086
45	1.69859073	-0.42678076	-0.90365554	-0.63970211
49	-1.24405953	-0.99983619	-1.13701419	0.79742189
53	0.45415027	1.13641987	0.16072053	-0.69640006
57	-0.14260835	-0.22594733	-1.65848233	0.41703334
61	0.90587706	-0.63856829	0.09426599	1.08687758
65	-1.07932436	1.02338518	-0.71706756	0.09151627
69	0.17743673	-0.27676332	0.12652943	1.71545090
73	-0.22976429	-0.61078579	-0.59805988	-0.27029325
77	0.02105906	0.36867699	1.54731961	1.34127640
81	-0.81465153	1.09017895	0.42802752	1.33923743
85	0.43419523	1.54034377	0.52813317	-0.38288546
89	2.42525497	0.76587313	0.10941043	1.78024635
93	1.02674079	0.31706752	1.49818371	-1.04028485
97	1.13778819	-0.27904640	0.89659561	-0.10102304
101	0.01422140	-0.52292416	-0.02270085	-0.20516447
105	-0.86266001	-0.85754069	1.29839062	-0.27513617
109	0.12597669	-1.28766185	-1.03452044	1.79556986
113	-0.16631490	-0.35103454	-0.57108506	0.58615820
117	-0.11842438	-0.68363554	-1.28961391	0.06983936
121	0.45652391	-0.53579354	-0.46103589	0.07720190
125	-0.76765899	0.37439870	0.84547194	-0.01453335
129	-0.18955655	-2.72975787	-0.65155342	-0.15300832
133	1.10403384	0.46651499	-0.33719809	-1.26178920
137	-0.36051489	0.69998796	-0.09536922	-0.34267542
141	-0.74653840	-1.71561070	1.36187393	-0.28788545
145	-0.31163867	-0.68194682	-0.89982777	0.18149651
149	1.30998532	-1.65028200	-0.02367118	-0.22597148
153	0.22238539	-2.34294614	-1.15741702	-0.74866424
157	-1.26111331	0.88124238	-1.96216777	-1.49358194

seed for next table: 1296113863

A.7.2 正規乱数表 (seed=1296113863)

XLISP-STAT において、下記のように、促進記号 (>) の後に、
(table-normal-random 1296113863) と入力することによって出力される。

```
> (table-normal-random 1296113863)
```

```
Table of Normal Random Number  
seed: 1296113863
```

i	i	i+1	i+2	i+3
1	-0.09826705	-0.52800655	-0.04197747	2.41908831
5	0.92985529	-1.66557714	-1.09621795	-1.23710416
9	0.76244243	0.39772000	-2.04196183	-0.08269475
13	-0.10076447	-0.71806476	0.60386474	-1.38387779
17	0.39057973	-0.42534031	-0.70811398	-1.54869081
21	0.39242445	2.09652316	-0.66280080	0.17972552
25	-1.68121009	0.62830260	-0.24688652	0.57928870
29	1.21989396	0.52666137	-1.58951216	-0.97762283
33	0.25119911	1.19585554	0.12640254	-0.40413492
37	2.04626166	-0.52260132	0.81287877	-1.80103731
41	1.40159941	0.52201951	-0.86710273	0.02943208
45	-0.75808761	-0.16521383	1.16532689	0.31094587
49	0.60261180	-1.00888285	-0.99169663	0.56132048
53	0.00359427	-0.74418404	-0.02574478	-1.88466954
57	0.20040786	0.79838344	-0.46370130	-0.11195211
61	1.62285568	0.87476939	0.35259837	-1.24549559
65	-1.47074659	0.99828275	1.04912592	-0.93400595
69	0.17474459	-1.70869238	-0.46872921	0.28323938
73	0.37194148	-1.04880180	-0.27401928	0.48645910
77	-0.22251760	-0.17151973	-1.72314512	1.35514681
81	-0.01434187	0.48033373	0.92472096	-0.43072843
85	-0.05242517	0.20160018	-1.44064012	-0.70205177
89	-0.99567072	0.79039304	0.36676002	-0.72628874
93	2.13716525	-0.14409293	0.32530896	-0.23300660
97	-0.76891769	-1.14566746	1.39207704	-1.62333819
101	1.32382108	0.43669385	0.25898813	1.48971522
105	0.68383372	1.35622664	-0.17385609	1.12975131
109	-0.32280793	-1.80609411	-1.73794564	0.87496884
113	0.33797722	-0.27926591	0.66648000	-0.10760162
117	0.04655043	-0.59767025	0.24069328	1.86669581
121	0.73154545	-0.27618277	1.17673635	-0.73404916
125	-2.06504918	-0.56743096	0.63194235	-0.17657268
129	-1.20193307	1.40300633	-0.74993409	0.77549595
133	0.35293842	0.55564329	-2.58179750	0.29408474
137	1.10690267	-1.75639283	-1.17333993	0.40957778
141	0.54049815	-1.90950383	-2.09635289	-0.41305733
145	-0.26275710	0.40679571	0.05354236	0.63216336
149	-0.42699396	-0.66480176	-1.94851330	-1.51217458
153	-0.23482371	0.44762348	0.53695220	2.10837055
157	-0.06585508	-0.85513947	0.88805087	-0.22057944

```
seed for next table: 2064273564
```

A.7.3 正規乱数表 (seed=2064273564)

XLISP-STAT において、下記のように、促進記号 (>) の後に、
(table-normal-random 2064273564) と入力することによって出力される。

```
> (table-normal-random 2064273564)
```

```
Table of Normal Random Number  
seed: 2064273564
```

i	i	i+1	i+2	i+3
1	-0.08684590	-0.72288745	-1.25567620	0.36747137
5	0.27540739	-0.12533760	-0.96135725	-0.40612572
9	-0.82257301	-0.26301037	-1.40683549	-0.64766658
13	0.40571969	-0.96192870	0.40935060	-0.14025240
17	-0.56387037	2.42087822	0.34613741	-0.37717492
21	-1.06275721	-0.79635465	0.86687684	0.26650041
25	-0.26997496	-0.59262127	0.45380755	-0.47220497
29	-0.14060157	0.05829600	1.22240028	-1.56908025
33	0.69811652	-0.69959849	0.48960766	-1.49797841
37	-0.68382922	-1.49971398	0.35738203	-0.31185266
41	0.41765154	0.45869778	-2.15331430	0.81438618
45	-0.91243599	-1.20597474	-0.71079063	0.51268143
49	-0.59821513	-2.57177347	0.17534092	0.38654821
53	0.94754136	0.00646587	1.62186507	0.34470338
57	1.43331473	-1.17928594	-0.23697606	1.33668261
61	2.02612914	0.70003975	-0.56164052	0.04894713
65	0.24718901	0.06763980	-0.27941855	1.20735588
69	-1.04680309	1.89275831	2.20816790	-0.28857285
73	0.46495710	-1.27090715	-0.83995333	1.05777057
77	-0.32660917	-0.12525158	-0.19924330	-2.55082111
81	-0.28285667	-0.72824298	0.04089085	1.64081694
85	2.01036830	-0.72255257	-1.93359453	1.40248307
89	-0.21775475	0.29821993	0.64726339	0.03950949
93	-1.50108695	-1.69661769	0.39160931	0.67929154
97	0.28656814	-0.41712546	-1.21867833	-0.48444764
101	0.91942332	-0.43255569	0.76248254	-0.22329604
105	-0.01441984	-0.32558081	-0.34051646	0.44073255
109	1.25673485	-0.68513829	-0.44690639	0.28570872
113	0.01032975	0.29652690	0.96020755	-0.28931064
117	0.47182316	-1.76687902	-0.36590688	-1.58546913
121	-1.02137991	0.45661462	0.13793982	0.26833362
125	0.07842021	-0.98916592	0.51740716	0.07132157
129	0.47849385	-1.24135045	-1.01193166	-0.94733489
133	1.75259833	0.01210905	-1.20272065	-0.12713462
137	-1.68680865	-0.54076581	0.50644768	1.35903859
141	-0.01439377	1.04345457	0.19396833	0.91379329
145	-1.64713783	0.04400919	1.93428762	-1.54072647
149	0.23023277	-0.65437841	-1.65164816	-0.72959903
153	1.21454291	-1.44475411	-0.17405075	1.98458401
157	0.58290097	0.49834337	-1.31424608	1.63515619

```
seed for next table: 768264379
```

A.7.4 正規乱数表 (seed=768264379)

XLISP-STAT において、下記のように、促進記号 (>) の後に、
(table-normal-random 768264379) と入力することによって出力される。

> (table-normal-random 768264379)

Table of Normal Random Number
seed: 768264379

i	i	i+1	i+2	i+3
1	0.57998407	0.56447374	0.18420473	-0.95143957
5	-0.24723803	0.61656496	0.37443452	0.39797935
9	-0.04926559	1.37704410	-1.31659448	-0.24447324
13	0.99572218	0.15834527	0.48745226	0.30429475
17	0.00479228	0.71517711	0.50423866	0.35755554
21	-0.31211922	0.30906660	0.31204447	-0.65238046
25	0.76156928	0.47974279	-0.86245353	0.07139992
29	-0.87066037	-0.64626370	-0.75408297	-0.84429975
33	1.08129019	1.45245674	0.25278376	-0.17329408
37	-0.06422589	1.19344760	1.06948915	-0.06925179
41	-2.56012786	0.41264489	-0.33318880	-2.63057998
45	-0.25780842	-1.66965412	-0.14239137	-1.24344134
49	-1.55058191	-0.51517093	0.14012196	-1.11507154
53	0.90780280	-0.30415561	1.54918803	-0.77433955
57	-1.93271611	-0.55272249	-1.02434069	0.04571767
61	-1.59670022	0.85393190	-0.53845336	0.28521825
65	0.83020196	0.87136099	0.68056786	0.82201042
69	-0.36349139	0.06962840	0.49753928	-1.61373837
73	-1.48122634	-0.86062940	-0.27992623	1.24383181
77	-0.31130689	-0.40140332	0.31092939	-0.71601680
81	-1.93651375	0.86598366	-0.83877477	0.33557544
85	1.18281625	0.00366596	-1.57152956	-0.05068419
89	0.67815499	-0.37645021	-0.67498427	1.03585556
93	0.84219579	0.69705905	1.20028881	-0.62645555
97	0.85437827	-1.33518660	-0.29799308	0.29810476
101	-1.99848419	-0.42830129	-0.32849892	-1.69008073
105	0.00212006	0.78902442	0.93552752	0.52426288
109	1.22847541	0.89976737	0.01774099	-0.23860040
113	0.70039228	-0.12251454	0.69996127	0.00641045
117	-1.07688706	0.48842622	-0.68149788	0.94762659
121	0.57103316	-1.05948105	-0.60893882	-0.87705269
125	-0.11587716	-0.52232405	-1.51312687	0.50864740
129	1.41672261	-1.96492337	-1.61316523	0.56307794
133	0.57053456	-1.75111524	-0.42797317	0.60771113
137	0.08642167	0.15626076	1.15385087	-0.67690960
141	-0.02522640	-0.61064944	0.51304367	-0.39810681
145	0.69044429	0.15173137	0.68750327	0.64963581
149	1.35828218	-1.43364752	-0.36082738	0.04948317
153	-0.06857389	-0.34169215	-0.45541367	-1.15807669
157	-1.04432984	2.45689280	-0.54178484	0.09785232

seed for next table: 1012670482

A.7.5 正規乱数表 (seed=1012670482)

XLISP-STAT において、下記のように、促進記号 (>) の後に、
(table-normal-random 1012670482) と入力することによって出力される。

```
(table-normal-random 1012670482)
```

```
Table of Normal Random Number  
seed: 1012670482
```

i	i	i+1	i+2	i+3
1	-0.47853465	1.01438829	0.31411414	0.52571798
5	0.09615417	-2.12548314	1.78958235	0.50573521
9	-0.77841736	0.22947644	0.58473634	-0.70010924
13	0.51777020	-0.91977234	0.64254877	0.46814959
17	1.51352798	-0.58862042	0.41985044	-0.36604290
21	-0.11015449	-0.15272350	-0.55815331	-0.33682076
25	0.23701428	0.48656952	-0.10320976	-1.66062091
29	-0.00094312	1.75607708	0.25546647	1.81446586
33	1.00366731	-0.16395795	0.55790999	-0.05619552
37	-0.98008647	0.72838797	0.08186234	-0.47864678
41	1.30331311	-0.61635091	-1.37332798	-0.26805653
45	1.47727753	1.39098915	-0.57926441	1.16776895
49	-1.19280932	-0.08247640	-0.19135429	-1.89718184
53	-1.41045253	0.22207787	0.60297493	-0.43108038
57	-0.52601764	0.53079622	0.13859496	-1.11837343
61	2.37983844	0.00771776	-0.27699501	0.84280215
65	-0.86470058	-1.78985057	-1.03537923	0.84030179
69	-0.59043055	-0.16110249	0.95514445	2.58277112
73	-1.51309768	0.68831495	-1.00176562	0.58263263
77	1.22186335	-0.60343826	-0.29151890	-0.21267740
81	-0.63395051	1.07990515	-0.29587511	-0.79700367
85	-1.23128884	0.51611933	0.54864745	0.48196980
89	1.86711836	-1.49504684	-0.19084597	0.28400729
93	-1.25341661	0.42997633	0.07142367	1.11434564
97	0.72320282	-0.85070976	-0.41998593	-0.21377478
101	0.34101142	0.23045119	0.69489746	-0.54402018
105	0.74035171	-1.40267547	-1.08373767	1.03725579
109	-1.48191962	0.82748474	-1.59097024	0.25894730
113	-0.36128001	-0.22529985	-0.27803017	2.49954059
117	2.00585769	1.14204596	0.70881758	0.96420730
121	2.17689989	-0.62393114	-0.95821543	-1.09835255
125	0.08292599	1.41564720	-0.64273669	1.56995145
129	-1.84118124	-0.28082152	0.53294391	-1.31639946
133	-0.52815201	-1.61127656	0.22721351	0.04088611
137	-1.51021076	-0.73823825	-0.53807571	-0.62435601
141	0.25535722	-1.05540955	1.01947403	-1.33103625
145	-0.52766437	-0.82850992	-0.08209100	1.53544571
149	0.66133240	0.70753089	-1.94298569	0.70692021
153	-1.36184980	1.63170717	1.88775150	-0.58592235
157	-0.12965022	0.18473995	0.41355929	-0.89084432
seed for next table: 1759160633				

参考文献

- [1] 伏見正則 (1989), 『乱数』, 東京大学出版会.
- [2] Harvey, B. and Wright, M.(1994), *Simply Scheme: Introducing Computer Science*, The MIT Press.
- [3] IBM(1959), *Random Number Generation and Testing: Reference Manual*, IBM.
- [4] 上條哲男 (1998), 「乱数について」, 『上智経済論集』, Vol.43, No.2, pp.139-178.
- [5] 上條哲男 (2000), 「LISP および Schemeによる乱数生成」, 『上智経済論集』, Vol.45, No.1, 2, pp.57-90.
- [6] Knuth, D.E.(1981), *The Art of Computer Programming, Vol.2, Seminumerical Algorithms*, 2nd ed.,Addison-Wesley.
 - 邦訳 1: 渋谷政昭訳 (1981), 『準数値算法/乱数』 (第 3 分冊), サイエンス社.
 - 邦訳 2: 中川圭介訳 (1986), 『準数値算法/情報構造』 (第 4 分冊), サイエンス社.
- [7] 栗原正仁 (1993), 『対話による Common Lisp 入門』, 森北出版.
- [8] Law, A.M. and Kelton, W.D.(1991), *Simulation Modeling and Analysis*, Second Edition, McGraw-Hill.
- [9] Marse, K. and Roberts, S.D.(1983), “Implementing a Portable FORTRAN Uniform (0, 1) Generator,” *Simulation*, Vol.41(October), pp.135-139.
- [10] McCarthy, J., Abrahams, P.W., Edwards, D.J., Hart, T.P., and Levin, M.I.(1962), *LISP 1.5 Programmer's Manual*, The MIT Press.
- [11] Park, S.K. and Miller, K.W.(1988), “Random Number Generators: Good Ones Are Hard To Find,” *Communications of the ACM*, Vol.31, No.10(October), pp.1192-1201.
 - 邦訳: 西村恕彦訳 (1993), 「乱数生成系で良質のものはほとんどない」,
 - 邦訳 1: 『bit』, Vol.25, No.4(April), pp.19-27.
 - 邦訳 2: 『bit』, Vol.25, No.5(May), pp.12-20.
- [12] Rubinstein, R. and Melamed, B.(1998), *Modern Simulation and Modelling*, Wiley.
- [13] Soft Warehouse(1992), *muLISP 90, Reference Manual*, Soft Warehouse.
- [14] Springer, G. and Friedman, D.P.(1989), *Scheme and the Art of Programming*, The MIT Press.

- [15] Steele, Jr., G.L.(1990), *Common LISP: The Language*, Second Edition, Digital Press.
邦訳: 井田昌之 翻訳監修 (1991),『COMMON LISP』, 第2版, 共立出版.
- [16] 竹村彰通 (1997),『統計』, 共立出版.
- [17] 竹内郁雄 (1986),『初めての人のための LISP』, サイエンス社.
- [18] 垂水共之 (1999),『Lisp-Stat による統計解析入門』, 共立出版.
- [19] Tierney, L.(1990), *LISP-STAT: An Object-Oriented Environment for Statistical Computing and Dynamic Graphics*,Wiley.
邦訳: 垂水共之, 鎌倉稔成, 林 篤裕, 奥村晴彦, 水田正弘 共訳 (1996),『LISP-STAT 入門』, 共立出版.
- [20] Winston, P.H. and Horn, B.K.P.(1989), *LISP*, 3rd Edition, Addison Wesley.
邦訳: 白井良明, 安部憲広, 井田昌之 共訳 (1996), LISP, 原書第3版,
() ().
- [21] 湯浅太一 (1991),『Scheme マニュアル』, 岩波書店.
- [22] 湯浅太一 (1991),『Scheme 入門』, 岩波書店.
- [23] 湯浅太一, 萩谷昌己 (1986),『Common Lisp 入門』, 岩波書店.