

## 教科書 「第2章 2. 乱数の利用」 プログラム訂正

<例2 - 2 . 1 > 整数乱数を出す (教科書 p28)

```

/* Display Random Number */
#include <stdio.h>
#include <math.h>

void main(void)
{
    int    i, rn, seed;

    printf("乱数の種を入力してください ==> ");
    scanf( "%d",&seed );
    srand( seed );

    for(i=0; i<50; i++){
        rn=rand();
/*      printf("%5d\n", rn); */
        printf("%10d\n", rn);
    }
}

```

```

-----実行開始-----
乱数の種を入力してください ==> 123
 47324114
1761033810
 806037626
 609924630
 936386220
   :
   中略
   :
429473559
 79947632
990075079
115750724
 751325151
-----おしまい-----

```

乱数とはサイコロの出目のような不規則な数のこと

乱数の出し方：

1. 関数 `srand` を用いて「乱数の種」という初期値 (下記関数 `rand` からの出力値の範囲内の任意の値) と与える。
2. 関数 `rand` により、 $0 \sim 2147483647 (=2^{31}-1)$  の整数乱数を出す。 `srand` で同じ種を与えると、同じものが出てくる。

関数 `srand`, `rand` を用いるときは、プログラムのはじめに、`#include <stdlib.h>` と書いておくこと。

## &lt;例 2 - 2 . 2 &gt; 実数乱数を小数第 4 位まで表示する (教科書 p29)

```

/* Display Random Number 2 */
#include <stdio.h>
#include <math.h>

void main(void)
{
    float rn;
    int seed,i;

    printf("乱数の種を入力してください ==> ");
    scanf("%d",&seed);
    srand(seed);

    for(i=1;i<=50;i++){
/*      rn = rand()/65536.0; */
      rn = rand()/2147483648.0;
      printf("%6.4f\n",rn);
    }
}

```

```

-----実行開始-----
乱数の種を入力してください ==> 321
0.0575
0.2833
0.2358
0.0506
0.5983
:
中略
:
0.1619
0.2039
0.9211
0.0857
0.2374
-----おしまい-----

```

実数乱数の作り方：

0.0 以上 1.0 未満の実数乱数を得るために、関数 rand で得られる整数乱数を 2147483648 ( $=2^{31}$ ) で割る。

## &lt;例 2 - 2 . 3 &gt; 円周率の計算 (教科書 p29 ~ p30)

```

/* 円周率を求めよう */
#include <stdio.h>
#include <stdlib.h>
#include <time.h>

#define MAXV 2147483648U /* MAXV = 2^31 (2 の 31 乗) */
/* U は unsigned int 型の定数であることを示す */

void main(void)
{
    int seed, ix, iy;
    float x, y;
    /* long nowtime, i, n; */
    int nowtime, i, n;
    int imax=100000;

    time( &nowtime );

    /* seed = (int)( nowtime%65536 ); */
    seed = (int)( nowtime%MAXV );
    srand( seed );

    printf("ただいま計算中です\n");
    n=0;

    /* for(i=1; i<=100000L; i++){ */
    for(i=1; i<=imax; i++){
        ix=rand();
        iy=rand();
    /* x=ix/65536.0; */
    /* y=iy/65536.0; */
        x=ix/(float)MAXV;
        y=iy/(float)MAXV;
        if(x*x+y*y<=1.0)
            n++;
    }
    /* printf(" =%10.8f\n", n/25000.0); */
    printf(" =%10.8f\n", (float)n*4.0/(float)imax );
}

```

```

-----実行開始-----
ただいま計算中です
=3.14416000
-----おしまい-----

```

乱数の種を時間から得る方法：

1. 関数 `time` により、システムで持っている時計の時間を得る。

関数 `time` の使い方： `#include <time.h>`

```

int nowtime;
time( &nowtime );

```

引数 `nowtime` に、グリニッジ標準時 1970 年 1 月 1 日 0 時 0 分 0 秒からの経過秒が得られる。

2. `nowtime%2147483648U` (`nowtime` を  $2^{31}$  で割った余り) を乱数の種として用いる。

## &lt;例 2 - 3 . 1 &gt; 乱数の頻度分布を調べる (教科書 p32)

```

/* Frequency of Random Number */
#include <stdio.h>
#include <math.h>

#define MAXV 2147483648U /* MAXV = 2^31 (2 の 31 乗) */
/* U は unsigned int 型の定数であることを示す */

void main(void)
{
    int frequency[6], irn, rn, i, seed;

    printf("何か数字をいれてください==>");
    scanf("%d", &seed);
    srand(seed);

    for(i=0; i<=5; i++)
        frequency[i]=0;

    for(i=0; i<10000; i++){
/*        irn=rand()/65536.0*6; */
        irn=rand()/(float)MAXV *6;
        frequency[irn]++;
    }

    printf("random number    frequency\n");

    for(i=0; i<=5; i++)
        printf("  %d                %5d\n", i, frequency[i]);
}

```

```

-----実行開始-----
何か数字をいれてください==>777
random number    frequency
  0                1726
  1                1653
  2                1659
  3                1666
  4                1652
  5                1644
-----おしまい-----

```