C プログラミング 入門

— 乱数・数学ライブラリ —

早稲田大学

今回の目標

- **擬似乱数**が使える．
  - `#include`
  - 初期化
  - 整数乱数
  - 実数乱数
- **数学ライブラリ**が使える
  - 数学関数・定数
  - コンパイル方法（gcc ファイル名 -lm）

ヘッダファイルのインクルード（#include）

- プログラムの先頭で「#include<stdio.h>」と記述してきたprintf 関数や scanf 関数を使用するためにインクルード（読み込み）していた
- C 言語には，関数を集めた**標準ヘッダファイル**が準備されている
- プログラム中で使用する関数に応じて，必要な標準ヘッダファイルをインクルードする必要がある
- 主な標準ヘッダファイルは次の通り：

stdio.h：	printf, scanf 等（入出力関数）
stdlib.h：	rand, srand, exit, malloc 等
math.h：	exp, sqrt, sin, cos 等（数学関数）
string.h：	strcpy, strlen 等（文字列処理関数）
time.h：	time 等（時刻，日付に関する関数）

擬似乱数

プログラムで (擬似) 乱数を利用する

乱数とは

- それぞれの数字が同じ確率で現れるように並べられた数字の列のことである
- C 言語のプログラムでは，rand 関数を使う．この関数は乱数を計算で求めている．各数字が正確に同じ確率で現れることはないので，擬似乱数とも呼ばれる
- rand 関数はある値を元に乱数を計算し，2 回目以降は前回の乱数値を元に計算するので，最初のある値が同じであれば，同じ乱数の列が現れる

擬似乱数

使用方法：必要なヘッダファイルを include する

```
#include <time.h>      /* time 関数を使う場合に必要 */
#include <stdlib.h>    /* これは必要 */
```

現在時刻を使って乱数を初期化する

```
srand((unsigned)time(NULL)); /* 乱数系列を初期化 */
```

- 実行させる度に同じ乱数を得たい場合は，これを省略するか「srand(56);」などと固定値を指定

擬似乱数（規則性がかなり残っている）

0〜RAND__MAX の整数乱数を得る

```
a = rand();
```

(RAND_MAX は, stdlib.h の中で 0x7fffffff と定義されている)

0 以上 1 未満の実数乱数を得る

```
a = rand()/ (RAND_MAX +1.0);
```

0 以上 N 未満の整数乱数を得る

```
a = (int) (rand()/(RAND_MAX +1.0)*N);
```

擬似乱数使用例

さいころの目（saikoro.c）

```
#include <stdio.h>
#include <time.h>
#include <stdlib.h>

int main(void){
    int a;
    srand((unsigned) time(NULL));
    a = (int) (rand()/(RAND_MAX+1.0)*6);
    printf("Result is %d. ¥n", a+1);
    return 0;
}
```

→　０～５までの乱数を作り，最後に１を足せばよい．

例題 1

今の運勢を表示するプログラムを作れ．

下の確率で右のメッセージを表示するプログラムを作れ．

確率	30 %	60 %	10 %
メッセージ	Very lucky	lucky	Not lucky

- 何回も実行させて，だいたい上記確率で表示されることを確かめよ
- time() は 1970 年 1 月 1 日 0 時 0 分（標準時）からの経過秒数を返すため，1 秒以内に何度も実行すると，同じ数字で乱数を初期化することになり，結果も同じになってしまう
- 結果を日ごとに変えたい場合は，次のようにする

```
srand((unsigned)time(NULL)/(60*60*24));
```

例題１のヒント

- #include，乱数の初期化を忘れずに
- 判定は if 文を使う（例えば）：

```
if(a<0.3)              printf("...")
else if(a<0.9)         printf("...")
else                   printf("...")
```

数学ライブラリ

- プログラムで $\sin$ や $\log$ などの数学関数，π などの数学定数を利用することができる．

数学関数や数学定数を使うには

1. ヘッダファイルを include する

```
#include <math.h>
```

2. 「-lm」（エル・エム）とつけてコンパイル

```
$ gcc ファイル名 -lm
```

これを忘れると「:undefined reference to 〜」とエラー

3. プログラムの中で数学関数を使用する（10 の自然対数）

```
a= log(10.0);
```

数学ライブラリ（参考）

数学関数

使用法	意味	使用法	意味
fabs(x)	**絶対値**	exp(x)	**指数**
log(x)	**自然対数**	log10(x)	**常用対数**
pow(x,y)	x **の** y **乗**	sqrt(x)	**平方根**
floor(x)	**（小数点以下）切り下げ**	ceil(x)	**切り上げ**
cos(x)	$\cos$	acos(x)	$\arccos$
sin(x)	$\sin$	asin(x)	$\arcsin$
tan(x)	$\tan$	atan(x)	$\arctan$
atan2(y,x)	$\arctan(y/x)$	cosh(x)	$\cosh$
sinh(x)	$\sinh$	tanh(x)	$\tanh$

数学定数

使用法	意味	使用法	意味
M_PI	π	M_SQRT2	**２の平方根**
M_E	e	M_LN10	$\log_e 10$

例題 2

例題 2：乱数を用いて，定積分の値を近似的に求めよ．

- **０以上１未満の乱数 $x,\ y$ を入力された回数だけ繰り返し，$y < \sin(\pi x)$ となった割合 r を計算する**
- **この割合は，$\displaystyle\int_0^1 \sin \pi x = \frac{2}{\pi}$ に収束する**
- **求めた値 r を表示し，$2/\pi$ との語差 (Error) を表示せよ**
- **表示は以下のようにする：**

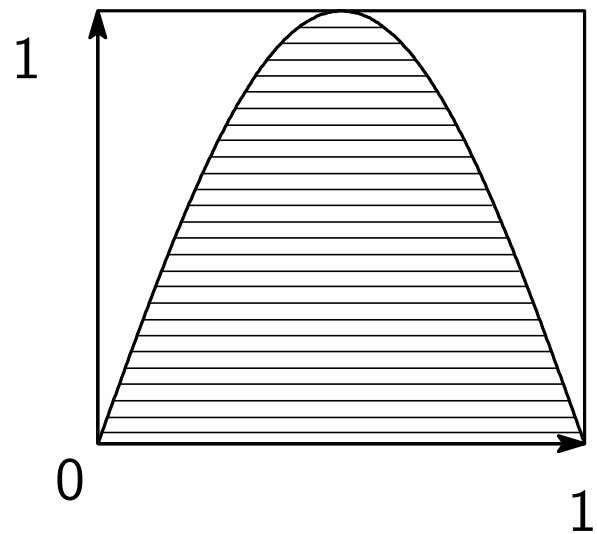
How many trials? 1000【Enter】
Result is 0.652000 (Error: -0.015380)

例題２のヒント

- ヘッダファイルの include を忘れずに（time.h,stdlib.h,math.h）
- -lm で数学ライブラリをリンクするのを忘れずに
- (x, y) の座標にそれぞれに対して 0〜1 の乱数を対応させる
- (x, y) の組を乱数により N 回発生させると，1×1 の正方形の中に N 個の点をばらまくことに相当する
- 求めたい割合：

$$\frac{\text{条件を満たした点の数}}{\text{点の数 N}}$$

- 点の数のカウントは int 型で行うこと
- 割合を計算するときにキャスト演算子を忘れずに



まとめ

- **擬似乱数**
  - `#include`
  - **初期化**
  - **整数乱数**
  - **実数乱数**
- **数学ライブラリ**
  - **数学関数・定数**
  - **コンパイル方法（**gcc **ファイル名** -lm**）**