

[目次](#) [前回](#) [次回](#) [略解](#)

計算科学☆演習 II

樋口さぶるお*¹ 配布: 2011-04-15 Fri 更新: Time-stamp: "2012-04-09 Mon 14:08 JST hig"

はじめに

講義ののり

必ず計算科学演習 II と同時に履修してね.

■**成績計算** 科目の成績 100 ピーナッツは

- 10 ピーナッツ: 毎回授業での quiz. 授業の最後にやります.
- 30 ピーナッツ: プチテスト いまのところ 2011-05-27 を予定
- 50 ピーナッツ: ファイナルトライアル
- 10 ピーナッツ: 授業時間外の予習復習 [シラバスから変更するかも](#) 平常点, 予習
な何か. 今日のところは e ラーニングシステムでの予習復習はありません.

■**資料** 授業で配布. 授業後に欲しい人は <http://hig3.net> から各自ダウンロード.
前のレターボックスに残ってることも.

■**欠席届** 去年とポリシー違います. 専用用紙に事情を説明する書類を貼って, 授業前
分に提出 (欠席の前でも後でも可. ただしファイナルトライアルが締切)



演習ののり

かならず計算科学 II と同時に履修してね.

実習室に行ったら, <http://hig3.net> → 計算科学☆演習 II へ. 別紙も参照.

担当者ののり

- なまえ: 樋口さぶろお hig@math.ryukoku.ac.jp
- へや: 1-502
- オフィスアワー: 木昼, 金 4. その他, 質問歓迎な時間: 月昼, 火昼. (水昼, 金昼や会議の前後) お弁当持参可. お湯あげます.
- Web ページ: 携帯/PC <http://hig3.net> 今日の演習の指示や, プチテストジュールもここから.

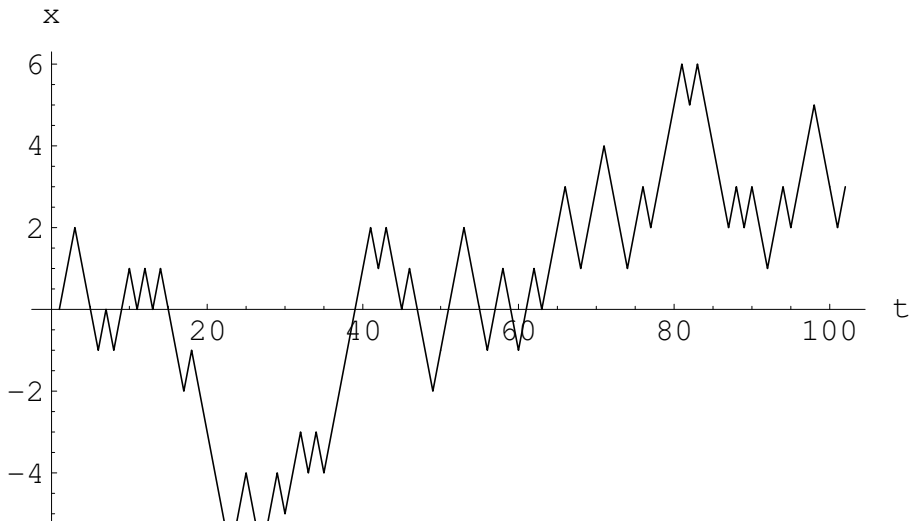
1 ランダムウォークと乱数

今日の目標

- ランダムウォークとふつうの数列の違いが説明できる
- C 言語で疑似乱数を生成する方法を説明できる

1.1 quiz:疑似乱数の使いかた

サンプルプログラムを参考に, 関数 `double uniform()` とライブラリ関数 `srand(int)` は与えられたとして, `scanf` でシードを取得, 設定し, $-1, 0, +1$ をそれぞれの確率で出力することを 100 回繰り返すプログラムを書こう.



疑似乱数の正しい/間違ったプログラム

ソースコード 1 乱数

```
1  /*
2  rand1.c -- [0,1)疑似乱数のサンプルプログラム
3  Time-stamp: "2011-04-15 Fri 12:48 JST hig"
4  */
5  #define _CRT_SECURE_NO_WARNINGS // おまじない
6  #include <stdio.h>
7  #include <stdlib.h> /* srand(), rand() を使うのに必要 */
8
9  double uniform(); /* 関数プロトタイプ宣言 */
10
11 int main(){
12     int seed; /* 疑似乱数のシード */
13     double s;
14     int t; /* カウンタ */
15     int tmax=100; /* 疑似乱数を得る回数 */
16
17     scanf("%d",&seed);
18
19     srand(seed); /* シードの設定 */
20     for(t=0;t<tmax;t++){
21         /* srand(seed); /* ここに置くとどうなるの? */
22         s=uniform();
23         printf("%f\n",s);
24     }
25     return 0;
26 }
27
28 /** [0,1) 疑似乱数を返す */
29 double uniform(){
30     return rand()/(RAND_MAX+1.0);
31 }
```