

ランダムウォークと擬似乱数生成

樋口さぶろお

龍谷大学工学部数理情報学科

計算科学☆演習 II L01(2013-04-10 Wed)

今日の目標

- 1 ランダムウォークとは何か説明できるようになるう
- 2 Cで擬似乱数を使えるようになるう.



<http://hig3.net>

ここまで来たよ

① はじめに

- この授業どんなのり?

② ランダムウォーク

- ランダムウォーク
- 擬似乱数
- 擬似乱数の正しい/間違ったプログラム

講義+演習の目標

もう少し正確にはシラバスを見てね.

- 確率的な現象を数式で解析できる
- 確率的な現象をシミュレートするプログラムが作れ, その結果をコンピュータで解析できる
- チームで協力して問題を解決できる, 効率よく質問できる, 自分の学習方法を改善できる

計算科学 II(講義) ののり

必ず計算科学演習 II と同時に履修してね.

成績計算注文の多い科目です…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
- 10 ピーナッツ:授業時間外の予習復習
- 30 ピーナッツ:プチテスト
- 50 ピーナッツ:ファイナルトリアル
- その他追加ピーナッツ. その時に説明.

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. 1-503 前のレターボックスに残ってることも.

欠席届ピーナッツ的に考慮されたい場合は, 専用用紙に事情を説明する書類を貼って, 授業前後各 5 分に提出 (事前事後とも可. ファイナルトリアルが締切). 欠席に事前連絡は不要. 何回欠席しても期末試験受験資格を失うことはありません.

計算科学演習 II ののり

必ず計算科学 II と同時に履修してね.

成績計算注文の多い科目です…科目の成績 100 ピーナッツは

- 20 ピーナッツ:毎回の課題
- 80 ピーナッツ:プチテスト $\times 3 = 15, 30, 35$

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

1 週間のタイムライン

- ① 水午前まで 予習復習問題 (e ラーニング or 携帯)
- ② 水 3 計算科学 II(7-002), quiz
- ③ 金朝まで 予習復習問題 (e ラーニング or 携帯)
- ④ 金 2 計算科学演習 II(1-619), quiz 返却
- ⑤ 金昼 休憩スペース (1-614)
- ⑥ 金 3 総合演習 (1-619)
- ⑦ 金 4 オフィスアワー (1-619)

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 6(1-539), 金 4(1-619). 訪問歓迎な時間: 月火木 昼. お弁当持参可. お湯あげます.
- Web ページ: <http://hig3.net> 演習の指示や, スケジュールもここから.

ここまで来たよ

① はじめに

- この授業どんなのり?

② ランダムウォーク

- ランダムウォーク
- 擬似乱数
- 擬似乱数の正しい/間違ったプログラム

C 言語で数列の計算

現象の数学 A

数列 $\{X_t\}$, 時刻 $t = 0, 1, 2, \dots$

初項 $X_0 = 0$

漸化式 $X_{t+1} = X_t + R_{t+1}$.

階差数列 $R_{t+1} = \text{定数}$ なら X_t は等差数列.

C 言語で数列を書くと?

```
int x;  
int r;  
int t;  
x=初項;  
for( /* t ここ省略 */ ){  
    printf("%d\n",x);  
    r=(階差数列の一般項 R_{t+1});  
    x=x+r;  
}
```

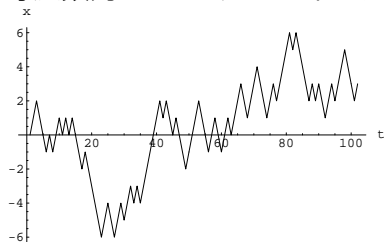
ランダムウォーク

ランダムウォーク \Leftrightarrow 階差数列 R_{t+1} が

つまり R_{t+1} がランダム. 例えば, こんな場合.

R_{t+1}	確率
+1	p
-1	$q(= 1 - p)$

等差数列 vs ランダムウォーク



ランダムウォークってどんなところに出てくる?

-
-
-
-

ここまで来たよ

① はじめに

- この授業どんなのり?

② ランダムウォーク

- ランダムウォーク
- 擬似乱数
- 擬似乱数の正しい/間違ったプログラム

擬似乱数列の生成

乱数列=ランダムな数列

R_{t+1} を C 言語でどう書く?

```
#include <stdlib.h>

/* 0以上 RAND_MAX 以下の正の整数をランダムに選んで返す関数 */
int rand();

/* その初期化 */
void srand(unsigned int seed);
```

RAND_MAX は M_PI みたいな定数. 値は処理系による. たとえば $2^{31} - 1$.
得られる値は, +1, -1 だけでいいんだけどな~

この授業の約束 (+世の中に近い習慣). `rand()` を生で使わず, $[0, 1)$ 一様乱数を返す `double getuniform()` にして使う.

```

/* [0,1) 一様乱数 */
double getuniform(){
    return rand()/(1.0+RAND_MAX);
}

/* yが[0,1)一様乱数なら, 返り値は確率1/4で0, 確率3/4で1*/
int getrandom(double y){
    if( y < 0.25){
        return 0;
    } else {
        return 1;
    }
}

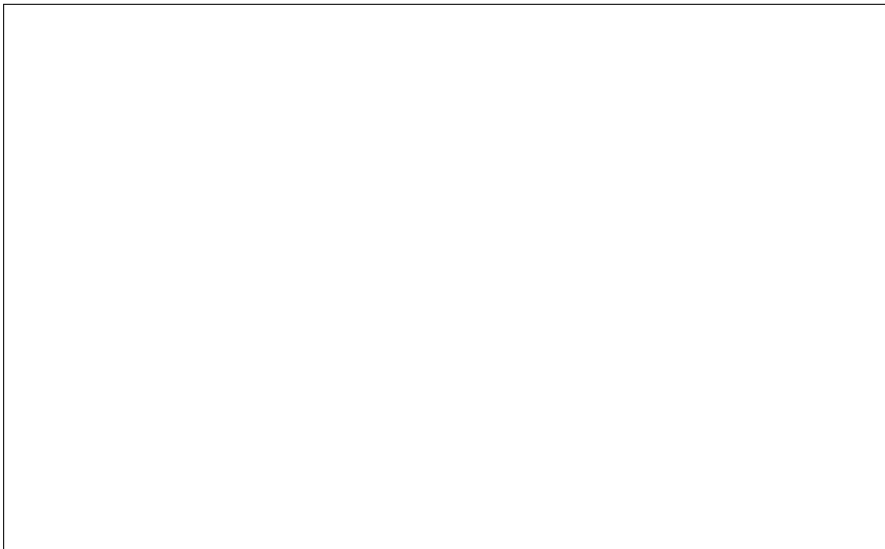
```

getuniform() の性質

- '値域' $0 \leq \text{getuniform}() < 1$.
- $(\text{getuniform}() < p \text{ となる確率}) = p$.

計算機の中はどうなってるの？

擬似乱数列 = 'ほぼ' ランダムな数列



ここまで来たよ

① はじめに

- この授業どんなのり?

② ランダムウォーク

- ランダムウォーク
- 擬似乱数
- 擬似乱数の正しい/間違ったプログラム

ソースコード 1: 乱数

```
1 /*
2 randl.c -- -1 or +1 を確率1/4, 3/4で選ぶ乱数
3 Time-stamp: "2013-04-09 Tue 18:57 JST hig"
4 */
5 #define _CRT_SECURE_NO_WARNINGS // VC++2008用おまじない
6 #include <stdio.h>
7 #include <stdlib.h> /* srand(), rand() を使うのに必要 */
8
9 /* 関数プロトタイプ宣言 */
10 double getuniform();
11 int getrandom(double y);
12
13 int main(){
14     int seed; /* 疑似乱数のシード */
15     int t; /* カウンタ */
16     int tmax=100; /* 疑似乱数を得る回数 */
17
18     scanf("%d",&seed);
19     srand(seed); /* シードの設定 */
20     for(t=0;t<tmax;t++){
21         /* srand(seed); /* /ここに置くかと? */
22         printf("%f\n",getrandom(getuniform()));
23     }
24     return 0;
25 }
26
27 /** [0,1) 一様疑似乱数を返す */
28 double getuniform(){
29     return rand()/(RAND_MAX+1.0);
30 }
31
32 /** -1 or +1 を確率1/4, 3/4 で返す乱数 */
33 int getrandom(double y){
34     if( y < 0.25 ){
35         return -1;
36     } else {
37         return +1;
38     }
39 }
```

Quiz(rand() の振る舞い)

次のプログラムで、A が出力される確率は？

```
if( getuniform()==getuniform() ){  
    printf("A\n");  
} else {  
    printf("B\n");  
}
```

- 0
- 0 に近い
- 1/2
- 1/2 くらい
- 1 に近い
- 1

Quiz(rand() の振る舞い)

次のプログラムで、A が出力される確率は?

```
if( getuniform() < 0.1 ){  
    if( getuniform() < 0.2 ){  
        printf("A\n");  
    }  
}
```

- ① 0
- ② 0.02
- ③ 0.1
- ④ 0.2
- ⑤ 0.3
- ⑥ 1

Quiz(疑似乱数の使いかた)

サンプルプログラムを参考に、引数 y として $[0, 1)$ 一様乱数が与えられたとき、下の確率で値を返す `int getrandom(double y)` を書こう。

返り値	確率
-1	1/3
0	1/2
+1	1/6

予習復習問題

- C-learning は携帯を利用した学習システム. ネットマン提供. 学生には (通信料以外) かかりません. <http://asp.c-learning.jp/s/>
- 明日 C-learning の小テスト公開するので金 11:00 までにやってね.
- C-learning からのメールのドメイン (拒否除外設定用)
@asp.c-learning.jp
- C-learning では、メールアドレスは運営会社に記録されますが、教員や他の学生には公開されません.