

社工情報処理実習 1

– 課題 7 の解答例 –

安藤 和敏
筑波大学社会工学系

平成 12 年 6 月 11 日

1. 2次元の一様分布 (統計学)

プログラムはファイル 1 を見よ。
実行結果は、以下のようになるはず。

```
banana:~/java% java twodurand
π = 3.136
banana:~/java% java twodurand
π = 3.232
banana:~/java% java twodurand
π = 3.092
banana:~/java% java twodurand
π = 3.124
banana:~/java% java twodurand
π = 3.144
banana:~/java% java twodurand
π = 3.164
banana:~/java% java twodurand
π = 3.124
```

2. 数値積分 (微分積分学)

プログラムの例はファイル 2 を見よ。実行結果は、こうなった。

```
面積 = 13.931481533685872
本当の値 = 13.932675629073547
```

```

/*****

twodurand.java:  $\pi$  の推定値を計算するプログラム

安藤 和敏 (筑波大学社会工学系)
2000.6.5

*****/
public class twodurand {
    public static void main (String[] args) {

        double x;          // x 座標のための変数
        double y;          // y 座標のための変数

        int N = 1000;      // 発生させる点の総数 1000 個
        int k = 0;        // 円の内部に入る点の個数を数えるための変数

        for (int i=0; i<N; i++) { // 以下を N 回繰り返す

            x = Math.random();    // [0,1) 区間の一様乱数を x に代入
            y = Math.random();    // [0,1) 区間の一様乱数を y に代入

            if (x*x + y*y <= 1) { // もし点 (x,y) が半径 1 の円の内部にあれば
                k++;              // それを数える
            }

        } // for の終わり

        System.out.println("  $\pi$  = " + 4.0*k/N); // 結果の出力 (4.0 とするところ
                                                // がみそ)

    } // main メソッドの終わり
} // twodurand クラスの終わり

```

ファイル 1.1: twodurand.java

```

/*****

daikei.java: 台形公式による積分の近似

安藤 和敏 (筑波大学社会工学系)
2000.6.5

*****/
public class daikei {
    public static void main (String[] args) {

        double a = 0.0;          // 積分区間の左端
        double b = 5.0;          // 積分区間の右端
        int N = 50;              // 分割した区間の数

        double h = (b-a)/N;      // 区間の幅 0.1 = (5-0)/50

        double sum = 0;         // 面積の途中計算結果を入れるための変数

        for (int i=0; i<N; i++) { // 以下を N 回繰り返す

            sum = sum + (i*h+2*Math.sin(i*h)+(i+1)*h+2*Math.sin((i+1)*h))*h/2;

        }                        // for の終わり

        System.out.println("近似値 = " + sum); // 結果の出力

        System.out.println("本当の値 = " +
            ((5.0*5.0/2 - 2*Math.cos(5.0))-(0.0*0.0/2- 2*Math.cos(0.0))));

    } // main メソッドの終わり
} // daikei クラスの終わり

```

ファイル 2.1: daikei.java