社工情報処理実習 1
– 課題 8 –

安藤 和敏
筑波大学社会工学系

2000 年 6 月 6 日

以下の 1 から 4 のうち 1 つを選択なさい。プログラム中にはできるだけ何をやっているのかが分かりやすいように日常言語 (日本語か英語) でコメント文を入れなさい。

1. コイン投げ

Math.random() メソッドは、$[0,1)$ 区間上の一様乱数を返すものだった。したがって、Math.random() が返す値が $\frac{1}{2}$ 以下である確率は $\frac{1}{2}$ である。($\frac{1}{2}$ より大きい確率も $\frac{1}{2}$。)「Math.random() が返す値が $\frac{1}{2}$ 以下である」という事象を「コイン投げで表が出る」ということと同一視しよう。

次の numOfheads(N) メソッドは、コインを N 回投げたときに、表の出た回数を数えて、それを返す。

```
public static int numOfheads (int N) {

    int head = 0;                          // 表の出る回数を数える変数

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

        if (Math.random() <= 1/2.0) {   // もし表がでたら
            head ++;                    // head に 1 を足す
        }

    } // for 文の終わり

    return head;              // head を返す

}
```

ここで課題。$N = 100$ とする。for 文を使って、numOfheads(N) メソッドを main メソッドから 10 回程度呼び出して、その各呼出しにおいてそれが返す値を表示するプログラムを書け。余力があれば、表の出る確率が $\frac{1}{2}$ でないような「いんちきコイン」を投げて表の出る回数を数えるメソッドを書いて、同様なことをやってみよ。

(正しいコインを投げたときの) 出力結果は以下のような感じになるであろう。

```
banana:~/java% java coin1
54
52
```

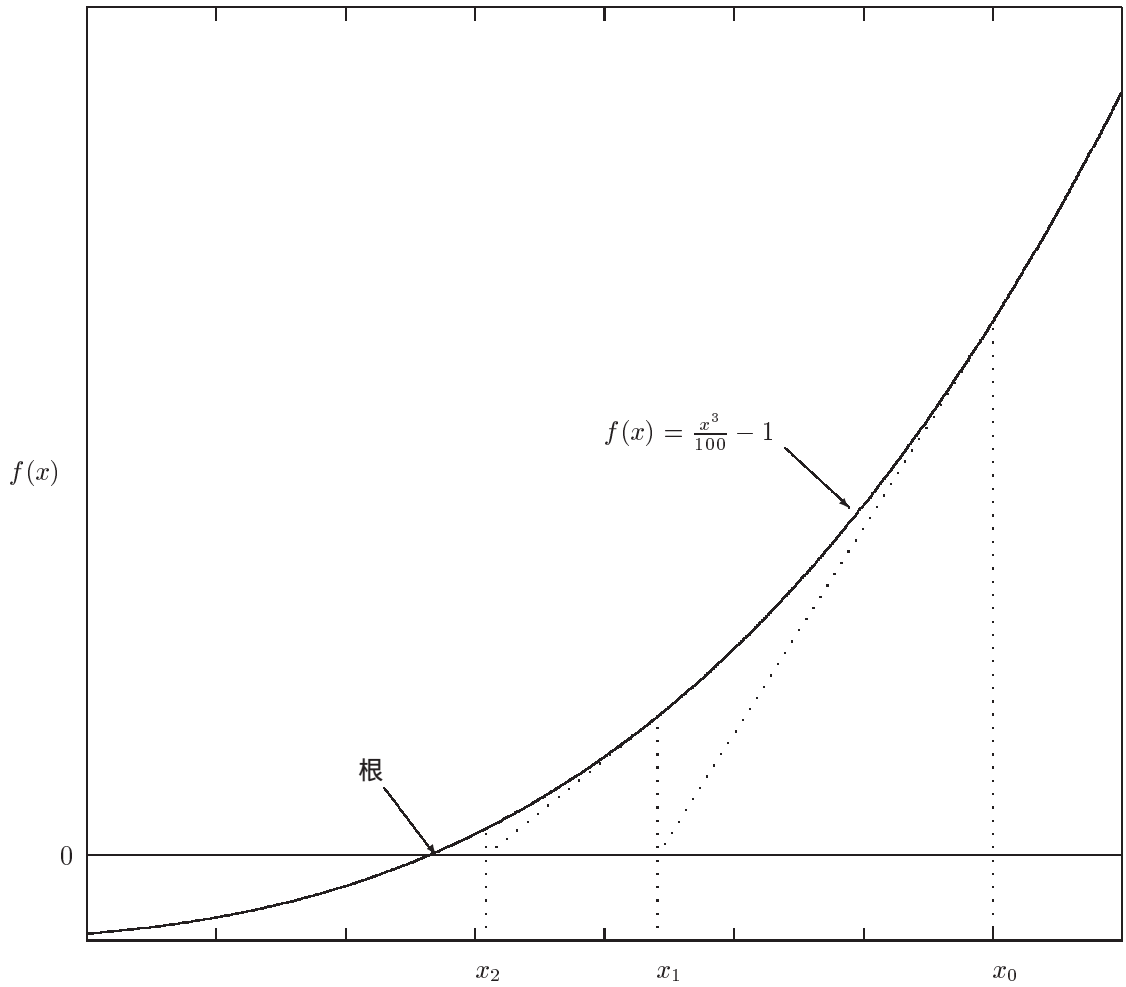

図 2.1: Newton 法

43
53
53
43
49
51
44
46

2. 方程式を解く – Newton 法 –

$f\colon \mathbb{R} \to \mathbb{R}$ を、1 変数関数とする。方程式 $f(x) = 0$ を数値的に解く方法として **Newton 法** と呼ばれるものがある。

まず適当に定めた初期点を x_0 として $y = f(x)$ のグラフの点 $(x_0, f(x_0))$ における接線を考えます。この接線の方程式は、

$$y - f(x_0) = f'(x_0)(x - x_0)$$

で与えられるので、この接線が x 軸と交わる点の x 座標を x_1 とすると、x_1 は

$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)}$$

で与えられる。

今度は、点 $(x_1, f(x_1))$ における $y = f(x)$ のグラフの接線を考えます。この接線の方程式は、

$$y - f(x_1) = f'(x_1)(x - x_1)$$

で与えられるので、この接線が x 軸と交わる点の x 座標を x_2 とすると、x_2 は

$$x_2 = x_1 - \frac{f(x_1)}{f'(x_1)}$$

で与えられる。図 2を見よ。

これを以下同様に繰り返して、 x_i が得られているときに x_{i+1} を

$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)}$$

で次々と計算して、数列 $x_0, x_1, x_2, \cdots$ が得られる。この計算は、$|f(x_i)|$ が十分小さくなったときに、x_i を $f(x) = 0$ の近似解として終了する。

以下に、for 文を使って $f(x) = \frac{x^3}{100} - 1$ のとき 方程式 $f(x) = 0$ の根を Newton 法を用いて求めるプログラムをの断片を示した。

```
public class newton {

    public static void main (String[] args) {

        double x = 9.0;                     // x の初期値 = 9.0
        double epsilon = 0.0001;            // ε  = 0.0001

        while (Math.abs(f(x))>= epsilon) {  // |f(x)| ≧ ε  である間は、
            x = x - f(x)/fdash(x);          // x を新しい x で置き換える。
        }                                   // while の終わり

    }

    public static double f (double x) {     // 関数 f(x) の定義


    }

    public static double fdash(double x) {   // 関数 f'(x) の定義


    }

} // newton クラスの終わり
```

このプログラムでは、main() メソッドの他に二つのメソッドがある。一つは $f(x) = \frac{x^3}{100} - 1$ を計算するメソッドで、もう一つは、$f'(x) = \frac{3}{100}x^2$ を計算するメソッドである。これらの、メソッドは main メソッドから呼び出される。Math.abs メソッドは絶対値を返す。

課題は、上の不完全なプログラムを完成させることである。

実行結果は、こんな感じであろう。

```
banana:~/java% java newton
4.641588853127458
```

余力があれば、次のことも試しなさい。上のプログラムでは初期点を 9.0 としたが、別の初期点で出発するとどうなるか？ 別の方程式ではどうなるか？

3. 数値積分 (前回の続き)

前回の台形公式を用いた近似積分のプログラム (課題 7.2.2) において, $f(x) = x + 2\sin(x)$ を計算するメソッドを定義し、main メソッドからそれを呼び出すようなものに改良しなさい。さらに余力があれば、別の関数の積分も試してみよ。

4. 自由課題

今日まで習ったことを使って、何か面白いプログラム、 あるいは、自分らしいプログラムを作れ。 オリジナリティがあり、かつ、技術的にも高度なものを評価する。あるまとまった処理を一つのメソッドで置き換えることによってプログラムが単純化されるものは、メソッドを使うことが望ましい。

自分のプログラムで「やりたいことがあるのにうまく行かない」という人は実習時間中に質問してください。それから、

http://infoshako.sk.tsukuba.ac.jp/~shakoj13/origin.html

や

http://infoshako.sk.tsukuba.ac.jp/~shakoj13/guide.html

も参考にしても良い。