

課題 1：最大公約数の計算

講義ノート

例題 2.1 (教科書より) 最大公約数
与えられた 2 つの正整数の最大公約数を計算せよ。

教科書では、この問題に対して次の 2 つのアルゴリズムを与えている。

```
1 program gcd_slow (int x,y)
2   N ← x と y の小さな方
3   for (k ← 1 ~ N) {
4     if (x % k = 0) ∧ (y % k = 0) {
5       ans ← k
6     }
7   }
8   return(ans)
9 program end.
```

プログラム 2.1 (教科書より): シラミつぶしの gcd 計算法

```
1 program euclid_gcd (int x,y)
2   r ← x % y
3   while (r > 0) { #余り r が 0 になるまで以下を繰り返す
4     x ← y; y ← r
5     r ← x % y
6   }
7   return(y)
8 program end.
```

プログラム 2.2 (教科書より): ユークリッドの互除法で gcd を求める

講義 (演習) では、前者を gcd1、後者を gcd2 と呼ぶことにする。また、 $t_1(n)$, $t_2(n)$ で各々の最悪時間計算量を表わすことにする。つまり、

$$\begin{aligned} t_1(n) &= \text{gcd1 に } n \text{ 桁の数を与えたときの最悪時計算時間} \\ t_2(n) &= \text{gcd2 に } n \text{ 桁の数を与えたときの最悪時計算時間} \end{aligned}$$

とする。ただし、 n は、与えられる 2 数の合計の桁数 (しかも 2 進表記での桁数) である。また、計算時間としては、アルゴリズム中の除余演算の回数を用いることにする。

【宿題】（~~切~~：10月11日）

問 1 桁数 n に対して、どんな入力用最悪の入力になるか示せ。また、その考察をもとに、計算量 $t1(n)$ を表わす式を求めよ。

問 2 互いに素な数の対 a, b を考える（ちなみに素とは、 $\gcd(a, b) = 1$ となること。なお、 a, b ともに 20 以上で、差は 5 以上になるような数を考えること。）それに対し $\gcd2(a, b)$, $\gcd2(b, a)$ を実行してみたとき、変数 x, y の値がどのように変化していくか、その変化の過程を示せ（教科書の p.48 のような説明）。

問 3 5 番目、6 番目のフィボナッチ数に対して、上と同様に $\gcd2$ の計算過程を示せ。

実験ノート

1. 実験課題

最大公約数を計算する 2 つのプログラム ($\gcd1$, $\gcd2$) を、入力される 2 数の合計桁数 (2 進) n を、10 ~ 50 の範囲で、適当な種類 (例: $n = 10, 14, 18, 22, 26, 30, 34, 38, 42, 46$) に対し実行し、その最悪時の計算時間 (除余演算の回数) を測定せよ ($\gcd2$ については、適当なフィボナッチ数の入力でよい。) その結果を gnuplot でグラフ化せよ。

さらに、 $\gcd1$ に対しては、適当な n に対して (その他いろいろやってみてもよい)、実際の計算時間 (CPU time) を計測し、それを基に n が大きくなったときの計算時間を予測せよ。

2. 使用ソフト

実験のための次のソフト (コマンド) 等は、ディレクトリ `~owatanab/pub/gcd` に用意してある。

実験データ生成用

gen (プログラムのソースは `gen.c`)

[自分のプロンプト] `./gen n`

○ n は欲しいデータの桁数。

- n 桁の自然数で $\gcd1$ の苦手なものを作る (さて何でしょう?)

genrand (プログラムのソースは `genrand.c`)

[自分のプロンプト] `./genrand n seed`

○ n は欲しいデータの桁数。seed は乱数の初期値。適当な値を使う。

- n 桁の自然数をランダムに作り、それを出力する。

genfib (プログラムのソースは `genfib.c`)

[自分のプロンプト] `./genfib t`

○ t は自然数 (欲しいフィボナッチ数の番号)。

- $t, t+1$ 番目のフィボナッチ数を求め、それを出力。各フィボナッチ数の 2 進桁数も同時に出力する。

実験対象プログラム

gcd1, gcd2 (プログラムのソースは各々 gcd1.c, gcd2.c)

[自分のプロンプト] `./gcdi x y`

- 最大公約数を求めたい自然数の対 x, y .
- x と y の最大公約数を求める .
- 計算中に行なわれた除余演算の回数も同時に出力 .

計測・データ処理用

time (Unix システムのコマンド)

[自分のプロンプト] `time command`

- `command` の実行にかかった時間を出力 . 出力される値のうち, 最初のが CPU time .

gnuplot (使用法については, 「項 4 . gnuplot 入門」を参照)

3 . レポートの構成について

レポートは以下のような構成にすること (レポートの体裁や基本的な注意事項については, 最初に配った「授業説明書」に従うこと.)

1. 実験手順: 実験手順についての説明 . どのような実験を, どのような種類のデータに対して, それぞれ何回行なったか, などを明確に書くこと .
2. 実験結果のまとめ: 実験で得られた結果を表とグラフにまとめたもの (詳しい個々のデータは付録にする) .
3. 考察: 計算量 $t_1(n)$ について, 理論的に $t_1(n)$ を導き, 実験で得られた結果と比較する . また, 実際の計算時間 (CPU time) と $t_1(n)$ の理論式から, n が大きくなったときに, 計算時間がどの程度になるかを予測せよ .
4. 付録: 今回の実験結果 (データが多量にある場合には, その抜粋) .

4 . gnuplot 入門

以下は奈良先端科学技術大学院大学の佐藤氏の書かれた「Gnuplot FAQ, version 1.2 (1994/4/28)」をもとに, 今回の実験に最低限必要な部分をまとめたものである . なお, 以下では例を中心に説明する .

(1) gnuplot とは

gnuplot とは, 対話的にグラフを書くプログラムで, 最初の版が T. Williams と C. Kelley により作成され, その後, いろいろな人によって改良が加えられている . グラフは, 関数 (例: x^{**2} や $\sin(x)$ など) の他に, 実験データなどの離散的な数値も描くことができる . また, 最小二乗法に基づいて, 実験データなどに合う関数を求めるためにも利用できる .

(2) gnuplot の基本的な使い方

まず, gnuplot の起動するには,

[自分のプロンプト] `gnuplot`

- gnuplot が起動されメッセージが出たあと, プロンプト (`gnuplot >`) が表示される .

関数のグラフを表示させるには、

```
gnuplot > plot x+x**2+sin(x)
```

- 「 $x+x**2+sin(x)$ 」に関数値を求める式を書く。
- グラフ用のウィンドウ上に、与えた式に基づいた関数（例： $y = x + x^2 + \sin x$ ）のグラフが描かれる。

データを表示させるには、

```
gnuplot > plot 'file1'
```

- 「file1」にファイルを指定する。このファイルには (x,y) の値を次のように入れておく（少数も可。また、順序はとくに関係ない。）

```
# 表示したいデータ x, y の組 ← # で始まる行は無視される
1, 4
2, 6
3, 8
⋮
```

- グラフ用のウィンドウ上に、与えられたデータ (x,y) に対応する点がプロットされる。さらに、プロットした点同士を線で結びたい場合には、

```
gnuplot > plot 'file1' with line
```

- データの順に線でつなぐので、この場合にはデータの順序が影響する。
- グラフ用のウィンドウ上に、与えられたデータ (x,y) を折れ線で結んだグラフが表示される。

二つ以上の関数やデータを表示させたい場合には、たとえば、

```
gnuplot > plot 'file1' with line, x+0.5*x**2
```

(3) 各種設定と出力

グラフ用のウィンドウの x 軸の範囲を自分で設定して表示させるには、

```
gnuplot > plot [-5:20.5] x+0.5*x**2
```

- 「[-5:20.5]」に x 軸の範囲を指定する。

さらに y 軸の範囲も指定したい場合には、

```
gnuplot > plot [-5:20.5] [-2:200] x+0.5*x**2
```

- 「[-2:200]」に y 軸の範囲を指定する。
- x 軸、 y 軸の範囲は、次に変えるまで同じものが使われる。

グラフ用のウィンドウの y 軸のメモリを対数メモリにするには、

```
gnuplot > set logscale y
```

- 以後、 y 軸のメモリは対数メモリになる。

逆に対数メモリを解除するには、

```
gnuplot > set nologscale y
```

自分の関数を定義するには、

```
gnuplot > f(x)=a*x+b*x**2
```

- 好きな関数名を使ってよい。また、パラメータ（例： a, b ）を使ってもよい。ただし、これらのパラメータは使う前までには値を定義しておく。

したがって、たとえば、次のように使える。

```
gnuplot > f(x)=a*x+b*x**2
```

```
gnuplot > a=3.4
```

```
gnuplot > b=35.45
```

```
gnuplot > plot f(x)
```

グラフをファイルに出力させるには、

```
gnuplot > set terminal postscript
```

```
gnuplot > set output 'file2'
```

- 「file2」に出力先のファイル名を指定する。
- このあと、画面に表示させるのと同じような操作をする。
- この設定以降に表示（plot）したグラフは、画面には表示されずに、すべて file2 へ書かれる（毎回上書きされるので注意。）なお、出力は ps ファイル形式なので、ghostview やプリンタへの出力は従来通りにできるはず。

再び画面（ウィンドウ）に表示するには、

```
gnuplot > set terminal x11
```

(4) 最小二乗法による係数の求め方

まず、パラメータを使って関数を定義しておく。そして fit というコマンドを使って、データファイルのデータに関数をもっとも合うようにパラメータを選ばせる。たとえば、以下のように実行する。

```
gnuplot > g(x)=a*x+b*x**2
```

```
gnuplot > fit g(x) 'data1' via a,b
```

- 「g(x)」で関数を、「data1」でデータファイルを指定する。
- 「via a, b」のところで、どのパラメータを調整するかを指定する。
- パラメータ（例： a, b ）を適当な初期値から始めて、何度か調整しながら、適当な精度（最小二乗誤差の意味で）が得られるまで、あるいは精度が上がらなくなるまで繰り返す。途中にいろいろと表示されるが、結果は、たとえば次のよう出力される。

Final set of parameters	68.3% confidence interval (one at a time)	
=====	=====	
a	= 0.5	+/- 2.31063e-16
b	= -0.5	+/- 2.7763e-15

パラメータの値がすでに何かに定義されている場合，それが初期値になるので注意が必要（もしかしたら，非常に悪い初期値かもしれないから）それを防ぐには，関数を定義する際に，今まで使わなかったパラメータ名を使えばよい．逆に，このことを利用して，パラメータの初期値を自分で適当なものに決めることもできる．

たとえば，

```
gnuplot > g(x)=a*x+b*x**2
gnuplot > a=3.5
gnuplot > fit g(x) 'data1' via a,b
```

参考文献

アルゴリズムについて少し勉強してみたい人のために

[1] は，コンピュータ・サイエンスのいろいろな話題を取り上げた本．その中にアルゴリズムや計算量に関する話題もある．[2] 以降の本はアルゴリズムに関する入門書．

- [1] A.W. Biermann (和田監訳)，やさしいコンピュータ科学，アスキー出版局，1993.
- [2] 五十嵐善英，西谷泰昭，アルゴリズムの基礎，コロナ社，1997.
- [3] R. Sedgewick (野下他訳)，アルゴリズムC (1～3巻)，近代科学社，1998.
- [4] A.V. Aho 他 (野崎他訳)，アルゴリズムの設計と解析I，サイエンス社，1977.

プログラムを読んでみたい人のために

この講義で使うプログラムは，すべてC言語というプログラミング言語で書かれている．C言語については，いろいろな入門書がある．ここでは一例として [5,6] をあげておこう．[7] には，様々なアルゴリズムが載っている．

- [5] B.W. カーニハン，D.M. リッチー (石田訳) プログラミング言語C，共立出版 ISBN 4-320-02697-6.
- [6] S. Oualline (岩谷訳)，現実的なCプログラミング，ソフトバンク株式会社，ISBN 4-89052-321-9.
- [7] 岡村，C言語によるアルゴリズム事典，技術評論社，1998.

gnuplot について

佐々先生のコンピュータ・サイエンス入門の Web page . gnuplot の各種情報へのリンクを掲載 .

- [8] <http://www.is.titech.ac.jp/sassa/cs-nyumon00/>