

## コンピュータ概論B

－ ソフトウェアを中心に －

#13 アルゴリズム・計算可能性

京都産業大学  
安田豊

## コンピュータは万能か？

- プログラミングできれば何でもできる
  - 本当だろうか？
- まず
  - 何でもプログラミングできるのだろうか？
  - 問題の内容が分かっていたら書けるはず
- プログラムさえ出来れば完成か？
  - できあがったプログラムは必ず答えを出せるのだろうか？
- そこに計算量の問題があります

## アルゴリズム

- 教科書 pp.120-
- 算法 - 問題解決のための処理方法
  - アルゴリズム自体は自然言語で記述可能
  - それを手続き的に表現すれば
  - 手続き型プログラミング言語に載り
  - ノイマン型計算機で実行可能となる
- アルゴリズムを考えるのは人間の仕事
  - コンピュータはアルゴリズムを生成しない

## アルゴリズムの例

- $x, y (x > y)$  二数の最大公約数を得る
  - ユークリッドの互除法
    1.  $x/y$  の余りを  $z$  とし、
    2.  $z = 0$  なら最大公約数は  $y$  そのもの
    3.  $z \neq 0$  なら  $y, x, z$   $y$  として 1. に戻る
- 停止性：有限時間に必ず終了する
- 確定性：必ず同じ解に到達する
- 効率：計算機の処理速度は無限に速くない
  - プログラムには待てる時間以内に必ず処理が終了することが求められる

## アルゴリズムの効率

- 教科書 pp.121
- アルゴリズムの効率化
  - コンピュータ資源の有効な利用に直結
  - 計算過程の再利用と加工（ノイマン型計算機処理の本質）にて
  - 可能な限り反復処理を削減する

## 計算可能性

- 教科書 pp.121
- 計算なら何でも出来る？
  - できるものとできないものがある
- 計算可能性
  - アルゴリズムが無い場合 = 計算不可能
  - 複数存在する（普通のこと）アルゴリズムのうち、たいていは計算量が異なる
  - 時間計算量（計算実行時間）の有限性
  - 領域計算量（メモリ所要量）の有限性

## 計算量

- 有限であっても実用的でなければ
  - ex. 入力に対する時間計算量の増加 (要素数  $n$  の並び替えに要する時間  $t$  が)
  - 指数関数的に増える:  $t = a^n$ 
    - 「難しい問題」と呼び、すぐに解けなくなる
  - 多項式のオーダーで増える:  $t = a * n^2 + b * n$ 
    - 「易しい問題」
  - 例えば  $a=2, b=4$  なら
    - $n=10$  で  $1024 : 600$
    - $n=25$  で  $33554432 : 1350$
    - $n=50$  で  $1125899906842624 : 5200$

## 効率の良いアルゴリズム

- 教科書 pp.122- 囲み記事
  - アルゴリズム次第で
  - 離散フーリエ変換:  $n^2$  のオーダーから  $n \log n$  のオーダーへ
  - 巡回セールスマン問題: 指数関数的増加から多項式のオーダーへ
- トレードオフ
  - 時間計算量と領域計算量
  - わかりやすさと効率 (教科書 pp.123)
  - アルゴリズムの開発、プログラミングの時間と、演算時間のトータルでの問題解決時間が重要

## 計算量と処理時間

- 指定した値  $i$  は  $m$  個の要素  $x$  の何番目かを調べる作業を  $j$  入力に対して行う
  - 頭から最後まで  $x_n < i$  のものを数える (単純だが毎回常に  $m * j$  時間かかる)
  - 並べ替えておいて、頭から比較して  $x_n = i$  になった時の  $n$  をとる (並べ替え時間が一回、あとは  $(m/2) * j$  程度か?)
  - 並べ替えておいて、二分探索で  $x_n = i$  になった時の  $n$  をとる (並べ替え時間が一回、 $\log_2 m * j$  程度か?)
- 万能の手法はない

## ソート

- ソート(並べ替え)にも多様なアルゴリズムが
- バブルソート
  - 両隣を比較し続けて交換する処理を  $n$  回繰り返す
  - 計算量は  $n^2$  で増える
- クイックソート
  - 全体を指標値の上下で二つに分け、その後上下パートごとに同様の処理を繰り返す
  - 平均計算量  $n \log_2 n$ 、最悪ケースは  $n^2$
- 他に何種類も研究されています (追求すると面白い)

## 将来へ

- 教科書 pp.182-
  - ネットワーク技術を抜きに将来は語れませんが、Aクラスで扱いましたのでここでは説明しません
- 技術は?
  - ノイマン型主流の趨勢はしばらく続く
  - ノイマン型の計算・記憶の二大機能はますます発展
  - 通信機能強化、高速・無線 (携帯化・遍在化)
  - 世界で最も DSL の安い国である事を忘れずに

## 将来へ

- 家庭内利用の普及
  - 他の技術との融合 (pp.182 最後)
  - Cocoon : 中身は Linux
  - SHARP のホームサーバ
- PCのさらなる進出
- 電子出版
- 大規模電子図書館
- インターネットと個人・組織・社会・人類の付き合い
  - その存在はもはや消せない事をわすれずに

## 将来へ

- 新しいコンピュータ
  - 非ノイマン型への期待  
(教科書の a,b,c 分類はノイマン型も含む)
- 新しいアプローチ
  - 超並列、グリッド
  - AI、ファジイ、ニューロ
  - 非同期回路

## 将来へ

- 教科書 pp.187-
- 技術とのかかわり
  - 正しく理解し、結果を正しく評価する
  - 現在の技術の特徴と限界を知り、新しい技術を創る
- 情報化社会(\*)とのかかわり  
(\* 曖昧なので、余り好きな表現ではないですが)
  - コンピュータとネットワークのある世界での個人としての自立を求められている
  - コンピュータとネットワークを手にして、人間にできることはなんだろう? (人間が自分たちの幸福のためにできることはなんだろう?)
  - 人とコンピュータ、人と人との協働の可能性を忘れない