アルゴリズム論 (第1回)

2003.9.29
櫻井　彰人
http://www.sakurai.comp.ae.keio.ac.jp/
質問は Algorithm@soft.ae.keio.ac.jp

きょうの講義概要

- 講義のねらいと概要
  - シラバス
- アルゴリズムとは
  - 語源、直観的定義、現代的定義
- アルゴリズムの例（ユークリッドの互除法）
  - アルゴリズムの定義との対応

0. 講義のねらいと概要

- 問題解決と意思決定
  - アルゴリズムの活用で「効率」よく
- 「本物」には大量な情報
  - 「効率」が本質的
    - » 実用になる、ならないの分かれ目
- 講義で扱うこと
  - 問題解決を行なう際に重要なアルゴリズム
    - » アルゴリズム全般に共通する性質
    - » アルゴリズム各論
    - » 利用するうえでの留意点

講義の内容

- アルゴリズムの定義
- 計算可能性と計算量
- アルゴリズムの分類
- 代表的アルゴリズム
  - 整列、探索、グラフなど
- 問題解決とアルゴリズム

⇩

- 本質的なところを中心に
  - 必要に応じた理解、パッケージの利用

人間の視点からのアルゴリズム

- アルゴリズム研究の出発点
  - 数学
    - » 効率のよい問題解決手法
      - 最大公約数を求めるユークリッドの互除法など
  - コンピュータ
    - » プログラミングの前に
      - コンピュータへの手順
- この講義の視点
  - 人間の問題解決
    - 実際的な問題と対比させて
    - 「数学」、「コンピュータ」の枠にとらわれないで
  - 複雑な現象の「抽象化」と「モデル化」

講義の進め方

- 分かる講義を目指す
  - 具体的なイメージが沸くように
  - 記述は厳密に
  - 検討は数学的に
- 資料
  - PowerPoint による
  - 研究室 web page に

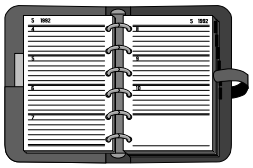
講義の進め方 (2)

- 理解を完全にするために
  - 知的好奇心
  - 資料を読んでおくこと
  - 演習問題を課す場合も

日程 (予定)

- 若干の変更も
- 講義の開始時刻は守りたい
  - 遅れないこと
  - 終了時刻は臨機応変

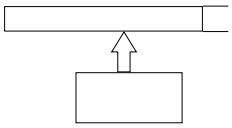
日程 (2)



- (1) 9月29日 「講義の概要、アルゴリズムとは」
  - シラバス説明
  - アルゴリズムの語源と定義
    - » ユークリッドの互除法を例に

日程 (3)

- (2) 10月6日 「計算可能性と計算量の考え方」
  - 計算可能性
    - » チューリング機械
  - 計算量
    - » 線形・2分探索とを例に
    - » オーダー記法
- (3) 10月13日 (体育の日)



日程 (4)

- (4) 10月20日 「離散数学のまとめ、アルゴリズムの分類と正当性の考え方」
  - 計算量にでてくる数式とそれらの大小関係
  - 集合の濃度
  - 分割統治法、発見的方法など
  - 繰り返しと再帰的アルゴリズムの正当性と数学的帰納法 (累積帰納法) による証明

日程 (5)

- (5) 10月27日 「整列 (1)」
  - 整列とは
    - » 単純法、挿入法、シェルソート
- (6) 11月3日 (文化の日)

日程(6)

- (7) 11月10日 「整列(2)」
  - クイックソート
- (8) 11月17日 「整列(3)」
  - ヒープソート
- (9) 11月26日 「探索(1)」
  - 探索とは
  - 探索のためのデータ構造
    - » 配列、表、スタック、キュー、木、2分木
  - 木の探索
    - » 2分木の巡回
    - » 縦型探索
    - » 横型探索

日程(7)

- (10) 12月1日「探索(2)」
  - 木の探索
    - » ゲーム木の探索
    - » MINI-MAX戦略
    - » α-β法
  - ハッシュ表と探索
  - 探索をめぐる話題
- (11) 12月8日 (休講?)

日程(8)

- (12) 12月22日 「グラフ(1)」
  - グラフとその表現
  - ダイクストラのアルゴリズム
  - クラスカルのアルゴリズム
- (13) 1月8日 「グラフ(2)」
  - フロイトの方法
  - グラフアルゴリズムのまとめ

日程(8)

- (14) 1月19日(月) 「さまざまなアルゴリズム」
  - バックトラック
  - 並列と分散
  - NP完全問題
  - ソフトコンピューティング
    - » 遺伝的アルゴリズム
    - » ニューロコンピューティング
  - 試験の概要
  - レポート返却

単位と評価

- 2通のレポート+期末試験
  - レポート: 内容を理解するために
  - 試験: 基本事項の理解を確認
    - » きちんとした試験勉強を
  - みんなに力を発揮してもらうため!
- 期末試験: 自筆ノート(A3用紙1枚)持ち込み可
  - 少しずつ、きちんと整理を
- Aは1/4〜1/3, Bは1/2前後

受講者への希望

- 簡単な演習を何回か
  - フィードバック
- 遠慮なく質問を
  - 講義時間外でも
    - » 25-609A室(櫻井)もしくは23-620室(TA)
    - » e-mailはAlgorithm@soft.ae.keio.ac.jp
    - » 私と研究室についての情報は、 http://www.sakurai.comp.ae.keio.ac.jp/

受講者への希望 (2)

- 講義全般
  - 受講者全員が最大限のものを得られるように工夫するつもり
- チャンスを十分に活用して身に付けてくれることを
- 自分の中からの「知的好奇心」が一番大切

参考書など

- 教科書は、なくても大丈夫
- 知識を補充するための文献リスト

参考書など (2)

- 1) 全体的に講義の参考にした本
  - 石畑清：アルゴリズムとデータ構造、岩波書店、1989.
  - Aho, A. V. and Ullman, J. D.: Foundations of Computer Science, W. H. Freeman and Company, 1992. 絶版
- 2) 問題解決とアルゴリズムの初歩
  - グロゴノほか著 (永田訳)：問題解決とプログラミング、近代科学社、1985. 絶版
  - 川島・永田 (共編)：はじめての情報処理、培風館、1992.

参考書など (3)

- 3) 計算可能性と計算量
  - 足立晄生：アルゴリズムと計算理論、森北出版、1990. 絶版
  - 笠井琢美：計算量の理論、近代科学社、1987. 絶版
  - Michael Sipser 著, 渡辺治/太田和夫 監訳: 計算理論の基礎, 共立出版, 2000.
- 4) アルゴリズム各論
  - エイホほか著 (大野訳)：データ構造とアルゴリズム、培風館、1987.

1 アルゴリズムとは

- 語源と定義
  - 語源
    - » アラビアの数学者 Abu Ja'far Muhammad ibn Musa Al-Khwarizmi (Born: about 780 in Baghdad, died: about 850)
  - 定義
    - » 直観的定義
    - » 計算とアルゴリズム
- アルゴリズムの例
  - ユークリッドの互除法
    - » アルゴリズムの記述

1.1 語源

- アラビアの数学者 Abu Ja'far Muhammad ibn Musa Al-Khwarizmi
  - インドの記数法（現在のアラビア数字）をアラビアに紹介（AD825年）
  - ラテン語に訳されてヨーロッパへ
    - » アラビア数字の記数法がAl-Khwarizmiのなまったalgorismiと呼ばれた
- 吉田洋一『零の発見』岩波新書、岩波書店、1939.

1.2 アルゴリズムの定義

- 直観的な定義
  - ある問題のすべての具体例を正しく解くための手段であって、具体的に定義され実行可能な一連の手続き
    - » 最大公約数を求めるユークリッドの互除法
    - » 線形計画問題を解くシンプレックス法やカーマーカ法
- アルゴリズムとはいいにくいもの
  - ことばや数式での表現ができない（むずかしい）
    - » 泳ぎ方や自転車の乗り方
    - » 名人芸（芸術、技能など）

狭義のアルゴリズム

- 狭い意味でのアルゴリズム
  - コンピュータに仕事をさせるための手順

「アルゴリズム」概念の萌芽

- Hilbert の第10問題 (1990)
  - "n 個の未知数を含む整数係数の多項式 $P(x_1,x_2,...,x_n)$ に対し，方程式 $P(x_1,x_2,...,x_n) = 0$ (ディオファントス方程式または不定方程式と呼ぶ) が整数解を持つか否かを有限的に判定する方法をみつけよ"
  - Given a diophantine equation with any number of unknown quantities and with rational integral numerical coefficients: *To devise a process according to which it can be determined by a finite number of operations whether the equation is solvable in rational integers*.
  - 実は、この方法(process)は存在しない. それを証明するには、「方法(process)」を厳密に定義しないといけない

「計算」とは (計算のモデル)

- 有限個の (良く定義された) 基本的手段がある
  - 四則、比較;文字を書く、写す、消す、進む、
- その手段が忠実に実行できる実行者がいる
- 使用する記号は有限個
- 計算の経過・結果を記す場所がある
- 計算は一歩、一歩行われる
- 何度繰り返しても同一の経過を辿る
- 有限回の実行で終了する

アルゴリズムの定義

- アルゴリズム =「ある問題のすべての具体例を正しく解くための計算」
  - 「問題」は「個別・具体的な問題」の集合
  - ディオファントス方程式で考えてみよ
  - アルゴリズムは「個別・具体的な問題」を入力し、解を出力
- 「アルゴリズム」定義の簡易版
  - 明確に定義された個別のステップの集合
  - ステップ間の制御のつながりや実行の順序も明確に定義
  - 正当な入力の範囲が明確（この要請は強すぎる）
  - 正当な入力に対し有限回ステップの実行で結果を出力
    - » 計算可能性 (Computability)、計算量 (Computational Complexity)
  - 正しい結果を出す
    - » 正当性 (Correctness)

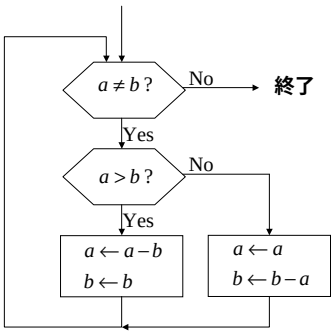
1.3 アルゴリズムの例 (ユークリッドの互除法)

- アルゴリズム1.1 (ユークリッドの互除法)
- 入力：ゼロでない二つの正整数MとN
- 出力：MとNの最大公約数
- 内容
  - (1) MをNで割った余りRを求める
  - (2) Rがゼロなら Nを出力して終了
  - (3) Nを新たにM、RをNとして (1) へ

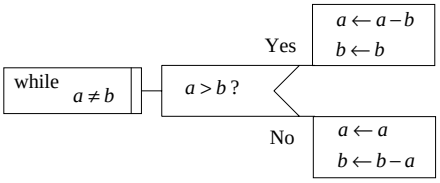
ユークリッドの互除法（数学的には）

$$M = q_0 N + R_1 \qquad (0 < R_1 < N),$$
$$N = q_1 R_1 + R_2 \qquad (0 < R_2 < R_1),$$
$$R_1 = q_2 R_2 + R_3 \qquad (0 < R_3 < R_2),$$
$$\vdots$$
$$R_{n-2} = q_{n-1} R_{n-1} + R_n \qquad (0 < R_n < R_{n-1}),$$
$$R_{n-1} = q_n R_n.$$

ユークリッドの互除法（その2）



ユークリッドの互除法（その3）



ユークリッドの互除法とアルゴリズムの5つの性質

- 個別のステップが明確に定義（自明）
- ステップ間の制御のつながりや実行の順序も明確に定義　→　説明
- 正当な入力の範囲が明確（自明）
- 正当な入力に対してステップの実行が有限回で結果（出力）を出す　→　説明
- 正しい結果を出す　→　証明

ステップ間の制御のつながりと実行順序

（1）から（3）は、何もなければこの順で実行
（2）では、判定結果によって実行の終了か（3）を続けて実行するかを指示
（3）では、次に（1）へ戻って実行することを指示

実行順序でのあいまいさが全くない

有限回での実行終了

- （1）から（3）を実行する（1回ループする）たびに余りが必ず小さくなる
- 余りRはゼロ以上
- 有限回のループでRが必ずゼロになる（実行が終了）

正しい結果を出す（アルゴリズムの正当性）

- ユークリッドのアルゴリズムで二つのゼロでない正整数MとNの最大公約数が求まる
  - Rがゼロになったときの N が、入力されたMとNの最大公約数gcd(M,N)になる

    以下が証明できれば「ユークリッドのアルゴリズムは正しい」

$$\begin{aligned} \gcd(M,N) &= \gcd(N,R_1) \\ &= \gcd(R_1,R_2) \\ &\quad \cdots\cdots \\ &= \gcd(R_{j-1},R_j) \\ &= R_j \end{aligned} \qquad (1)$$

ただし、$R_{j+1} = 0$

ユークリッドの互除法の正当性

- $M = N \times q_1 + R_1 \qquad (0 \leq R_1 < N)$
  - $R_1 = 0$ のとき、Nが最大公約数
    - $M = N \times q_1$ なので、$\gcd(M,N) = N$
  - $R_1 \neq 0$ のとき、$\gcd(M,N) = \gcd(N,R_1)$
    - {MとNの公約数全体} ＝ {NとR1の公約数全体}
      - {MとNの公約数全体} ⊆ {NとR1の公約数全体}
      - {MとNの公約数全体} ⊇ {NとR1の公約数全体}
  - 以上を余りがゼロになるまで繰り返すことで、前頁の（1）が証明できた