2015. 4.22

アルゴリズムとデータ構造

Ibaraki Univ. Dept of Electrical & Electronic Eng.

Keiichi MIYAJIMA

質問および授業に関する情報

授業に関する質問は、E-mailでも受け付けます。

質問がある場合は、下記のアドレス宛にメールを送って下さい。

kmiyaji@mx.ibaraki.ac.jp

授業に関する情報は、下記のホームページを見てください。

http://fm.ee.ibaraki.ac.jp/index.html

アルゴリズムと
その解析

計算モデル

–なぜ、アルゴリズムを学ぶのか？–

●アルゴリズムとは何か？

コンピュータで問題を解くとき、その問題を解くための手順を人間がコンピュータに与えなくてはならない。

このような**機械的に実行可能な手順**

アルゴリズム

計算モデル

−なぜ、アルゴリズムを学ぶのか？−

●アルゴリズムの善し悪し

一つの問題に対し、アルゴリズムは複数存在する。

人間はアルゴリズムを適切に選ぶ必要がある。

↓

計算時間や**使用した記憶領域**が**小さければ小さいほど良い**アルゴリズム

計算モデル

–なぜ、アルゴリズムを学ぶのか？–

●アルゴリズムの善し悪し（例1）

多項式 $p(x) = a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0$ の値を計算する

但し、$n, a_n, a_{n-1}, \cdots, a_1, a_0, x$ の値は与えられているものとする。

もし、左側から順番通りに計算をしていったとすると･･･

$a_n x^n$: n回の乗算

$a_{n-1} x^{n-1}$: n-1回の乗算

$a_{n-2} x^{n-2}$: n-2回の乗算

加算

加算

⋮

$a_1 x$: 1回の乗算

合計で $\left(n^2 + 3n\right)/2$ 回の基本演算（乗算と加算）が必要

計算モデル　−なぜ、アルゴリズムを学ぶのか？−

●アルゴリズムの善し悪し（例1）

多項式 $p(x)=a_n x^n + a_{n-1}x^{n-1} + \cdots + a_1 x + a_0$ の値を計算する

但し、$n, a_n, a_{n-1}, \cdots, a_1, a_0, x$ の値は与えられているものとする。

同じ問題を以下のように計算すると･･･

$$p(x) = (\cdots((a_n x + a_{n-1})x + a_{n-2})x + \cdots + a_1)x + a_0$$

項が一つ増える毎に乗算と加算が1回ずつ増える。

合計で $2n$ 回の基本演算（乗算と加算）ですむ

計算モデル
–なぜ、アルゴリズムを学ぶのか？–

●アルゴリズムの善し悪し（例1）

多項式 $p(x) = a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0$ の値を計算する

但し、$n, a_n, a_{n-1}, \cdots, a_1, a_0, x$ の値は与えられているものとする。

初めの方法では $\left(n^2 + 3n\right)/2$ 回の基本演算

後者の方法では $2n$ 回の基本演算

このような、どの命令や演算も一定の単位時間で実行できると仮定する評価基準：**一様コスト基準**

計算量

- 時間計算量と領域計算量

アルゴリズムにしたがって計算を実行したときの計算時間

時間計算量

計算中に使用した記憶領域の量

領域計算量

実際の計算では**時間計算量の方が重要**

「効率の良いアルゴリズム」とは時間計算量が小さいアルゴリズムのこと

計算量

- 最大計算量（最悪計算量）

入力の規模nが大きくなるのにしたがって、計算時間がどのような割合で増加するか？

入力の規模nに対するアルゴリズムの計算時間を関数 $f(n)$ で表し、このように定義したアルゴリズムの計算量を

最大計算量（最悪計算量）

先ほどの例1の場合、

前者の方法: $f(n)=\left(n^2+3n\right)/2$

後者の方法: $f(n)=2n$

計算量

●漸近的計算量(評価)

入力の規模(サイズ)nを大きくしていったときの計算量 $f(n)$ のふるまい

アルゴリズムの善し悪しを計る尺度。

しかし、厳密に $f(n)$ を求める必要はない

先ほどの例1の場合、

前者の方法: $f(n) = \left(n^2 + 3n\right)/2 \boxed{= O(n^2)}$

後者の方法: $f(n) = 2n \boxed{= O(n)}$

なぜか？

計算量

厳密でなくて良い理由

例：以下計算量となる2種類のアルゴリズムA，Bがあったとする

$$f_{\mathrm{A}}(n) = 2000n$$
$$f_{\mathrm{B}}(n) = n^2$$

一つの命令が 10^{-8} 秒で実行されるとすると、n＜2000ではBの方が早いが、n＞2000でAの方が早くなる。

n＝10000となると、Aは0.2秒で終わり、Bは1.0秒となる。

さらに、n＝500000までくると、Aは10秒で計算を終えるが、Bはなんと40分以上計算が必要となる。

このようにアルゴリズムの善し悪しをはかる場合、定数倍部分は無視しても良い。

計算量

- 指数時間アルゴリズムと多項式時間アルゴリズム

計算量が入力サイズnの指数関数（例えば 2^n や 3^n, $n!$等も含める）であるようなアルゴリズム

指数時間アルゴリズム（exponential time algorithm）

計算量がnの多項式（ n や n^2 ）であるようなアルゴリズム

多項式時間アルゴリズム（polynomial time algorithm）

指数関数アルゴリズムは、ごく小さなサイズの入力をのぞけば実用的とは言い難い

本講義で取り扱うものは多項式時間アルゴリズムのみ

本日のまとめ

- アルゴリズムとは？

  アルゴリズムの評価方法

- 計算量

  時間計算量、漸近的計算量（評価）
  指数時間アルゴリズム、多項式時間アルゴリズム

本日の課題

1．計算量 $f(n)$ が次のような式で表されるとき、漸近的計算量はいくらか。

(a) $f(n) = n + \log n + 0.5^n$

(b) $f(n) = n^{2.05} + n^2 \log n$

(c) $f(n) = n^3 + 2^{n/3}$

2．関数 $n\log n, n^2, n^3, n!, n^n$ について、n＝10, 20，30，・・・としたときそれぞれの関数のグラフを描け。（Excel等を利用して良い）

レポートの〆切と提出先

レポート提出先

E2棟（旧システム棟）6F606室（宮島教員室）前

レポートBOX

レポート〆切:

木曜日　PM5：00頃

レポートはA4の紙に学籍番号と氏名を記入し、課題1については直接自筆で記入、課題2についてはExcel等で描いたグラフをプリントアウトして提出すること。

次回について

- 次回は5月1日（金）です。

  ゴールデンウィークで水曜日が連続で休みになるため、5月1日金曜日を水曜授業で行います。