コンピュータ・サイエンス入門：課題３
配布 2015.12.3

課題３：アルゴリズム入門（その２）

講義ノート

1. アルゴリズムの計算量

同じ計算をするプログラムでも，アルゴリズムの工夫次第で，かなり計算速度に差が出る場合が多い．ただし，つねに速くなる場合ばかりではない．そのような場合も含め，アルゴリズムの良し悪しを評価する尺度が「計算量」である．この講義・演習では，計算速度を議論するための「時間計算量」について考える．

情報処理で求められているのは，入力データに対して何らかの処理をして目標の答えを出すような計算問題である．つまり，1 つの計算問題でも，実際には無限個の入力データに対する計算を考えねばならない．たとえば,「最短経路長問題を解く」といっても，ある特定の 1 つのグラフを処理すればよいのではなく，どのようなグラフ（正確には隣接距離配列）に対しても計算できるようなプログラムを設計することが求められる．つまり，対象となるグラフは無限個ある．したがって,「このグラフに対して速かった・遅かった」で一喜一憂していても意味がない．そこで登場するのが時間計算量である．

一般に，計算量は入力サイズの増加にともなう計算コストの変化を示す関数である．入力データは通常，そのデータ量を表わすパラメータが考えられる．それを入力サイズという．たとえば最短経路長問題の場合，グラフの頂点数 n が入力サイズだ．入力サイズは問題ごとに異なる．場合によっては複数のパラメータになることもある．重要なのは，実際の応用場面でのデータ量の増減を的確に表わすパラメータを入力サイズとして使うことである．といっても大抵は常識的に決められるのでそう悩む必要はない．アルゴリズムの時間計算量とは，この入力サイズの増加に対して，その計算時間がどのように増えていくかを表わす関数である．

同じ入力サイズのデータに対してでも計算時間が異なる場合がある．そのような場合，通常は「最悪の入力データ」を考える．つまり，そのサイズのデータの中で最も計算時間のかかる入力である．それはプログラムごとに異なるかもしれない．たとえば，ある計算問題を解くプログラム P を考えよう．この計算問題の入力サイズを表わすパラメータを n とする．たとえばサイズ $n = 10$ の入力データの中で，P が最も苦手で計算時間のかかる入力データに対する計算時間が t_{10} 秒だったとする．同様に，$n = 20$ での最悪の入力データに対する計算時間が t_{20} 秒，$n = 30$ では t_{30} 秒 $\cdots$ だったとすると，プログラム P の計算量 $\mathrm{Time}_{\mathrm{P}}$ は，

$$\mathrm{Time}_{\mathrm{P}}(10) = t_{10},\quad \mathrm{Time}_{\mathrm{P}}(20) = t_{20},\quad \ldots$$

という値をとる関数なのである．

このような計算量の決め方は最悪時基準とよばれる．これに対して,「これでは悲観的すぎる．ほとんどの場合で効率よく解ければ十分だ」という考え方もあるだろう．そのような観点から平

均計算時間に基づく計算量を用いる場合もある．実際，最短経路長問題に対しても，最悪時ではあまり良くないのだが，平均的に考えると非常に高速に計算するアルゴリズムもある．

ところで，時間計算量を議論する際に実際の計算時間を用いるのは現実的でない場合が多い．プログラムにする前に，アルゴリズムの良し悪しを検討する場合，実際の計算時間まで考えられないからである．また計算時間は使うコンピュータによってもかなり変わってくる．そこで，アルゴリズムの良し悪しを議論する場合，通常は，計算に必要な「代表的な演算」の回数を見積もって，それを用いてアルゴリズムの計算量を定義する方法がとられている．

2. 最短経路長計算アルゴリズム `dist1`, `dist2` の計算量

「じゃあ，何が代表的な演算なの？」と言われると，統一的な答えはない．アルゴリズムごとに異なるからである．一言でいえば，そのアルゴリズム中で（とくに最悪時の入力の際に）最も実行されている演算である.（曖昧ですみません！大抵の場合には，何が代表的かはすぐわかります．ただし，アルゴリズムが複雑になると，そうもいかなくなるのですが ...）

ということで，最短経路長問題に対して我々が実験する 2 つのアルゴリズム `dist1`, `dist2` について考えてみよう.（我々は，実際のプログラム `dist1.rb`, `dist2.rb` をすでに持っているが，ここでは，そのプログラムをアルゴリズム `dist1`，`dist2` とみなして議論する．多くの場合には，アルゴリズムを，これほど厳密に書いていなくても計算量の解析はできる.）

さて，この各々で，もっとも実行されそうな演算は何だろうか？それを追求していくと ...

では，その演算は何回くらい行われるだろうか？その数を，この問題の入力サイズパラメータ（グラフの頂点数 n でしたよね！）を用いて表わしてみよう．それが `dist1`, `dist2` の時間計算量 $\mathrm{Time}_{\texttt{dist1}}$，$\mathrm{Time}_{\texttt{dist2}}$ である.（見積りは結構大ざっぱでもよい.）

$$\begin{aligned} \mathrm{Time}_{\texttt{dist1}}(n) &= n^a \text{ に比例} \qquad (a \text{ を求めよう}) \\ \mathrm{Time}_{\texttt{dist2}}(n) &= t(n) \text{ に比例} \qquad (t(n) \text{ を求めよう}) \end{aligned}$$

このように「○○に比例する」というような形で評価することが多い．この場合，たとえば「$\mathrm{Time}_{\texttt{dist1}}(n) = O(n^a)$」と書き，`dist1` の計算量は「オーダー n^a である」という．

こうしたオーダーによる見積りをアルゴリズムの 漸近的評価 という．もちろん，実際の応用では本当の計算時間が重要だ．そうした実時間に基づく計算量を求めるには，実際にアルゴリズムに沿ってプログラミングし，使用するコンピュータ上で実測実験をすればよい．そのデータに基づき，たとえば n^a に対する比例係数を決め，オーダーによる計算量を詳細化すればよいのである．