BU CLA CS 530: Analysis of Algorithms

Spring 1999

Handout 1

A Historical Note

Algorithms have existed for a long time, although they were not always called by their proper name and mostly referred to number-theoretic algorithms. The word "algorithm" is a deformation of the last name of a 9th century Persian mathematician, Abu Ja'far Mohammad ibn Musa al-Khawarizmi (in Arabic: "Father of Ja'far, Mohammad, son of Moses, native of Khawarizm"). An early German dictionary, *Vollständiges Mathematisches Lexicon* (Leipzig 1747), gives the following definition for *Algorithmus*: "Under this designation are combined the notions of the four types of arithmetic calculations, namely addition, multiplication, subtraction, and division." By the middle of this century the word "algorithm" was still not very much in use, and mostly associated with what is perhaps the oldest (if not the most famous) algorithm in the history of mathematics, namely, Euclid's algorithm for finding the greatest common divisor of two numbers. The Euclidian algorithm was originally written in terms of two operations, subtraction and exchange of two numbers, and first appeared in Euclid's *Elements* (Book 7, Propositions 1 and 2). Euclid was a Greek mathematician who lived and worked in Alexandria, Egypt, around 300 b.c., which thus makes the Euclidian algorithm nearly 2,300 years old!

The Euclidian algorithm (original form). It describes a method for finding the greatest common divisor (gcd) of two positive integers m and n.

(1) $A \leftarrow \min(m, n)$, $B \leftarrow \max(m, n)$.

(2) Subtract A from B as many times as needed to produce a number C smaller than A.

(3) If $C = 0$, the algorithm terminates, A is the answer.
If $C \neq 0$, let $B \leftarrow A$, $A \leftarrow C$, and go back to step (2). ■

(Check that the algorithm is correct!) Of course Euclid did not write the algorithm exactly in this form. For one thing he wrote in ancient Greek. Moreover there are no extant manuscripts written by Euclid himself. His writings have had to be reconstructed from numerous commentaries and translations by later writers.

The Euclidian algorithm (modern form). Identical to the original form, except that step (2) is replaced by (2′):

(2′) $C \leftarrow B \bmod A$. ■

(Check that steps (2) and (2′) are equivalent!) Although the two versions of the algorithm above, the original and the modern, are mathematically the same, their running times are markedly different — this is where algorithm analysis matters! We will have more to say on this on a later occasion.

Just as number-theoretic algorithms have a long history, so too investigation of their efficiency long predates the advent of modern computers. Formal analysis of the properties of the Euclidian algorithm dates back to at least the first half of the 19th century. In a 1841 paper, the French mathematician Binet showed that the number of division steps taken by the Euclidian algorithm (in its modern form) to calculate the gcd of two positive integers is less than 10/3 of the number of decimal digits in the smaller integer. A few years later, at the session of the *Académie des Sciences* on 28 October 1844, another French mathematician, Lamé, apparently unaware of Binet's work, presented a note in which he gave a similar bound, with 5 in place of 10/3. At a session of the *Académie* a week later, on 4 November 1844, Binet mentioned his own previous work and gave a different derivation for a lemma by Lamé on the rate of growth of the Fibonacci numbers.

This is all very modern stuff and very much in the spirit of what we do in algorithm analysis! The results by Binet and Lamé may be the first explicit demonstrations that a particular algorithm can be executed in polynomial time. As you probably know already, an algorithm is *polynomial-time* if the computation time it requires on a given input is bounded by a polynomial function of the size of the input (measured by, for example, the number of binary or decimal digits for the case of an input consisting of positive integers). Lamé cites earlier and unnamed "traités d'Arithmétique" where it is shown that the number of steps in the Euclidean algorithm is bounded by half the smaller argument, which of course is exponential in the size of the input – and, as he points out, not even true for very small arguments. (Check the various claims in this last sentence!)

The historical information above was collected from several sources:

- *Mathematical Thought from Ancient to Modern Times*, by M. Kline, Oxford 1972.
- *The Art of Computer Programming*, Vols 1 and 2, by D. Knuth, Addison Wesley 1973.
- "Review of *A Course in Computational Algebraic Number Theory*" by H. Trotter, *Bulletin of the AMS*, Vol 31, Oct 1994, pp 312-318.

Exact references for the papers by Binet and Lamé can be found in the review by Trotter. A detailed analysis of the Euclidean algorithm, with a proof of Lamé's lemma on which it is based, is given in Section 33.2 of the book by Cormen *et al.*

Assaf Kfoury
Created: 95.01.20
Modified: 99.01.06