15 Number Theory

15.10 The Continued Fraction Method for Factorization (8 units)

This project is related to material in the Part II course Number Theory.

1 Factor bases

In this project N will be a (usually large) integer, that we would like to factor, and B will be a finite set of (usually small) primes. We call B the factor base. Sometimes it is convenient to allow -1 as an element of B.

> **Question 1** Write a program, using trial division, to test whether N is a product of primes in B (we say that N is B-smooth), and if so to give the prime factorization. Use your program to estimate, for a suitable range of d, the probability that a d-digit integer is B-smooth where B is the set of primes less than 50.

The integers that may be accurately represented in your chosen language may be limited to 10^{15} or similar. For example in MATLAB numbers are represented by default as doubles, meaning that they are stored to 16 significant (decimal) figures. For integers larger than 10^{15} functions such as `mod` and `int2str` may give incorrect answers. If your language is able to handle larger integers, then you are still expected to comment where appropriate on how you would manage if you were restricted to 10^{15}.

On a modern computer it is practical to factor integers of the size considered in this project by trial division. We study a factoring method that remains practical for much larger values of N. To enable comparisons, we therefore make the following artificial restriction: the factor base B is only allowed to contain primes that are less than 50.

2 Continued fractions

The continued fraction algorithm applied to a real number $x_0 = x$ forms a sequence of partial quotients a_n by the transformation

$$\begin{aligned} a_n &= \lfloor x_n \rfloor \\ x_{n+1} &= \frac{1}{x_n - a_n} \end{aligned}$$

where as usual $\lfloor x \rfloor$ denotes the greatest integer $\leqslant x$. The algorithm terminates if $x_n = a_n$; this happens if and only if the initial x is rational.

The convergents P_n/Q_n are defined for $n \geqslant 0$ by

$$\begin{aligned} P_n &= a_n P_{n-1} + P_{n-2} \\ Q_n &= a_n Q_{n-1} + Q_{n-2} \end{aligned}$$

with initial conditions $P_{-2} = 0$, $P_{-1} = 1$ and $Q_{-2} = 1$, $Q_{-1} = 0$.

Question 2 Show that if $x = \sqrt{N}$ for some positive integer N then each x_n may be written in the form $(r + \sqrt{N})/s$ with r, s integers and $s \mid (r^2 - N)$.

Write a program to develop the continued fraction expansion of $\sqrt{N}$. Your program should work with integers as far as possible, so that there is no risk of rounding errors. In some programming languages it is best to use an integer type, but there is no particular advantage in doing this if you are using MATLAB.

Tabulate the partial quotients of $\sqrt{N}$ for $1 \leqslant N \leqslant 50$ and comment on the results. Investigate how large r and s can become (in terms of N).

For a fixed value of the positive integer N, the equation $x^2 - Ny^2 = 1$, in integer unknowns x and y, is called Pell's equation. The negative Pell equation is $x^2 - Ny^2 = -1$.

Question 3 Tabulate the quantities $P_n^2 - NQ_n^2$ for some values of N and hence comment on the use of continued fractions to solve Pell's equation, and the negative Pell equation. Can you see any simple condition on N which ensures that the negative Pell equation is insoluble?

Write a procedure that given $x, y, N \leqslant 10^{15}$ tests whether $x^2 - Ny^2 = \pm 1$, being careful to avoid integer overflow. (Hint: try working mod p for several primes p.)

Use your observations to write a program to find non-trivial solutions to Pell's equation. Tabulate the solutions found for each N in the ranges $1 \leqslant N \leqslant 100$ and $500 \leqslant N \leqslant 550$, when such a solution exists. You may find a few values of N, such as 509, beyond the capacity of your program; you do not need to correct for this. You should however make sure that all answers you do give are correct.

3 Factorization

By "factorization" we mean the task of finding a non-trivial factor of a composite integer N. We will not be concerned with finding the complete factorization of N into primes. You should assume from now on that N is odd.

Question 4 Suppose we are given a supply of integers x, y with $x^2 \equiv y^2 \bmod N$. How might this help us factor N? Estimate the complexity of the steps involved. If N is composite, can we be sure that suitable x and y exist?

One source of integers x, y with $x^2 \equiv y^2 \bmod N$ arises from the convergents of the continued fraction expansion of $\sqrt{N}$.

Question 5 Modify your programs to compute $P_n \bmod N$ and $P_n^2 \bmod N$ for N up to 10^{10}. Explain how you avoid integer overflow. (Hint: a multiplication mod N can be done in two pieces.) Run your program for $N = 2012449237$, 2575992413 and 3548710699.

Question 6 Let A be a matrix over the field $\mathbb{F}_2 = \mathbb{Z}/2\mathbb{Z}$. Write a program using Gaussian elimination to determine whether there exists a non-zero column vector $\mathbf{v}$ with $A\mathbf{v} = 0$, and to find one if there is.

Question 7 Implement the continued fraction method for factorization, and discuss briefly how it works. Your program should be capable of giving details of the intermediate steps involved. Give some detailed worked examples, including the values of N in Question 5. Investigate the number of convergents typically required for factorization.

Discuss the choice of factor base B. What improvements could be made to your method to make it more efficient for very large values of N?

References

[1] Davenport, H., *The higher arithmetic.*

[2] Koblitz, N., *A course in number theory and cryptography.*

[3] Riesel, H., *Prime numbers and computer methods for factorization.*