Applied Number Theory

Hendrik W. Lenstra, Jr.
University of California, Berkeley

1. Primality Testing

A prime number is an integer bigger than 1 that has no factor except 1 and itself. A number that is not prime is called composite. The primality testing problem is to decide whether a given integer is prime or composite. It is considered to be well solved, in contrast to the factoring problem, which asks for the factorization of a given integer into prime factors.

As an example of what can be accomplished with present-day primality testing algorithms, we mention the number

$$(10^{1031} - 1) / 9 = 111 \ldots 111 ,$$

which consists of 1031 ones. It was proved to be a prime number by H.C. Williams. As another example, consider

$$(2^{512} + 1) / 2424833 = 552 \ldots 209 .$$

This number has 148 digits, and with a computer it is very easy to prove that it is composite. However, no factor of this number is known, except the trivial factors, which are 1 and the number itself.

Most modern primality testing methods depend on Fermat's theorem, which was proved by the French mathematician Pierre de Fermat (1601-1665). It states that every prime number n has the property that

$$a^n \equiv a \bmod n$$

for all integers a. Here we make use of the notation "$b \equiv c \bmod n$," which was introduced by Carl Friedrich Gauss in 1801. It is pronounced "b and c are congruent modulo n," and it means that $b - c$ is divisible by n; or equivalently, that b and c leave the same remainder when divided by n.

If one is not certain whether n is prime, then one can pick a few integers a and test whether $a^n \equiv a \bmod n$. It may not be immediately obvious that this is easy to test. Suppose, for example, that n has several hundred digits, which is a realistic size in the context of primality testing. Then even for a small integer a such as $a = 2$ the number a^n is far too large to ever be calculated. Even if the whole universe would consist of ink and the necessary paper were supplied for free, it would be impossible to print the decimal expansion of the number

$$2^{(10^{1031}-1)/9}.$$

Fortunately, we do not need a itself, but only its remainder upon division by n. To illustrate how this remainder can be found, we consider the example $n = 29$. Then we have

$$a^{29} = (((a^2 \times a)^2 \times a)^2)^2 \times a\,,$$

which shows that a 29th power can be computed using four squarings and three multiplications. To prevent the numbers from growing too large, all intermediate results are replaced by their remainders after division by n. Thus, with $a = 6$ we find successively

$$\begin{aligned}
6^2 &= 36 \equiv 7 \bmod 29,\\
6^3 &= 6^2 \times 6 \equiv 7 \times 6 = 42 \equiv 13 \bmod 29,\\
6^6 &= (6^3)^2 \equiv 13^2 = 169 \equiv 24 \bmod 29,\\
6^7 &= 6^6 \times 6 \equiv 24 \times 6 = 144 \equiv 28 \bmod 29,\\
6^{14} &= (6^7)^2 \equiv 28^2 = 784 \equiv 1 \bmod 29,\\
6^{28} &= (6^{14})^2 \equiv 1^2 = 1 \bmod 29,\\
6^{29} &= 6^{28} \times 6 \equiv 1 \times 6 = 6 \bmod 29,
\end{aligned}$$

which is consistent with the fact that 29 is prime. The same method can be used to calculate the remainder of a modulo n for all a, m, and n. Even for numbers of hundreds of decimal digits, it is practically feasible.

If one finds by means of such a computation, that $a^n \not\equiv a \bmod n$ for some integer a, then Fermat's theorem implies that n cannot be a prime number, so that n is composite. One might think that inspection of the proof of Fermat's theorem then leads to a nontrivial factor of n, but it turns out that the computations to make this work are completely unfeasible, even for fairly small values of n. This explains how a number such as

$$(2^{512} + 1) / 2424833$$

can be known to be composite, without an explicit nontrivial factor being known.

For the purposes of primality testing, Fermat's theorem has two shortcomings. The first is that it does not characterize prime numbers. For example, the number $n = 1729$ is composite, since $1729 = (7)(13)(9)$; however,

$$a^{1729} \equiv a \bmod 1729$$

for all integers a. Such numbers n are relatively rare, but one doesn't want a primality test to incorrectly decide that they are prime.

The second shortcoming of Fermat's theorem is that even if it would characterize prime numbers, one would never be able to try out all integers a. The question arises whether it suffices to just try a small set of a's.

The first shortcoming is the less serious one. To remove it, one replaces Fermat's theorem by a strengthened version that does characterize prime numbers, as follows.

Theorem 1. If n is an odd prime number, then

(i) $a^{(n-1)/2} \equiv \pm 1 \bmod n$ for all integers a for which the greatest common divisor, $\gcd(a, n)$, equals 1;
(ii) $a^{(n-1)/2} \equiv -1 \bmod n$ for at least one integer a.

Conversely, if n is an odd integer bigger than 1, for which (i) and (ii) hold, then n is prime.

To see the connection with Fermat's theorem, write

$$a^n - a = a(a^{(n-1)/2} - 1)\,(a^{(n-1)/2} + 1).$$

If n is an odd prime number, then n divides the left hand side, so it must divide one of the three factors on the right. This implies assertion (i) of the theorem. The proof of the other assertions is left to the reader.

We are still faced with the second problem, namely, that it is not feasible to try all integers a. Several methods have been proposed to resolve it. The first is to try a small number of values of a—say, ten or twenty—that are randomly chosen from $\{1, 2, \ldots, n-1\}$, and view them as representative for all integers. If they all satisfy $a^{(n-1)/2} \equiv \pm 1 \bmod n$, and at least one of them satisfies $a^{(n-1)/2} \equiv -1 \bmod n$, then n is declared to be prime, and in all other cases n is declared to be composite. This test may give the wrong answer, but the probability for this to happen is very small, independent of n. Also, the test is very efficient, and it may be said that it solves the primality testing problem for practical purposes. In practice, a variant of this test that was introduced by Michael Rabin (1980) would be used. This variant is more efficient, and it also has the advantage of never declaring a prime number composite, as the above test may do. It may, however, declare a composite number prime, with a very small probability.

The second approach, attributed to Gary Miller (1976), is to try all integers a satisfying $1 < a < 2(\log n)^2$, where "log" denotes the natural logarithm. This is less efficient, but it may lead to more mathematical certainty concerning the correctness of the answer. Namely, if one adds the restriction $1 < a < 2(\log n)^2$ to the conditions (i) and (ii) appearing in Theorem 1, then all assertions of the theorem can still be proved to be valid provided that one assumes the truth of the generalized Riemann hypothesis from analytic number theory. The Riemann hypothesis is a conjectural assertion about the location of the zeros of a certain complex function that arises in the study of the distribution of prime numbers. It was formulated in 1859 by the German mathematician Bernhard Riemann (1826-1866), and its validity would have very attractive consequences in the theory of numbers. To this day it remains unproved. The generalized Riemann hypothesis is a similar assertion that arises during the investigation of the distribution of prime numbers that satisfy certain additional constraints, and it is likewise unproved. If, at some future date, the generalized Riemann hypothesis is proved, then the primality testing problem may be said to have been solved for theoretical purposes as well.

Following are methods that can currently be used to acquire mathematical certainty concerning the primality of a large number n. It turns out that Rabin's test, referred to above, does yield mathematical certainty in cases where it declares the number to be composite. The prime numbers present difficulties: if a number creates the strong impression of being prime (e.g., because Rabin's or Miller's test declares it prime) how is this proven? It should be stressed that this is the only problem, and for this reason, primality testing algorithms are sometimes referred to as primality proving algorithms. The output of the algorithm is not just the statement that n is (or is not) prime, but it is also a proof of the correctness of this statement.

Several modern primality tests attempt, roughly speaking, to prove that a number n is prime by means of the following trivial primality criterion:

an integer $n > 1$ is prime if and only if each factor of n is a power of n.

For example, if n is prime, then its only factors are $1 = n^0$ and $n = n^1$. The proof of the converse may be left to the reader.

The classical approaches to the primality testing problem that originated with Édouard Lucas (1842–1891) and D.H. Lehmer (1905–) can be formulated so as to fit this very imprecise description as illustrated below.

Theorem 2. Let n be an odd positive integer, and suppose that

(i) $a^{(n-1)/2} \equiv \pm 1 \bmod n$ for $a = -1$, $a = 2$, and $a = 3$; and
(ii) $a^{(n-1)/2} \equiv -1 \bmod n$ for at least one integer a.

Then for every factor r of n there exists an integer $i \geq 0$ such that $r \equiv n^i \bmod 24$.

The proof of this theorem utilizes the quadratic reciprocity law, which was proved by Gauss (1801).

In comparing this theorem with the previous one, note first that condition (i) has been substantially weakened. Instead of requiring that $a^{(n-1)/2} \equiv \pm 1 \bmod n$ for all a, it is now only required for $a = -1$, $a = 2$, and $a = 3$ (and the first of these is superfluous, since any power of -1 is ± 1). This makes condition (i) very easy to test in practice. Condition (ii) is the same as before, and it is just there to prevent the theorem from being wrong ($n = 1729$ would again be a counterexample). For seven out of every eight prime numbers n, an integer a as required in (ii) is already encountered in (i), among 1, 2, and 3, and for the remaining ones such an a is generally easy to find by trying a few values.

Another major difference with the previous theorem is in the conclusion; namely, it is not asserted that n is prime, but only that its factors behave in a certain way (more precisely, modulo 24) as if they are powers of n. It may be noted that $i = 0$ or $i = 1$ may be assumed in the conclusion of the theorem, because $n^2 \equiv 1 \bmod 24$.

Theorem 2, as it stands, is not very useful for primality testing. This is because the conclusion $r \equiv n^i \bmod 24$ does not put a very heavy restriction on possible factors of n. It excludes in most cases no more than 75 percent of the numbers which might potentially be factors of n.

In practice, generalizations of Theorem 2 that depend on the higher reciprocity laws from algebraic number theory would be used. Special cases of these higher reciprocity laws, which actually suffice for the application to primality testing, were first proved by Carl Gustav Jacob Jacobi and Ernst Eduard Kummer in the 1840s. A more complete theory was later developed in the context of class field theory. This area took shape under the hands of Weber, Hilbert, Takagi, and Artin, over the period 1891 to 1927.

Inspecting the proof of Theorem 2, which we suppressed, it is seen that the essential property of the number 24 that makes the quadratic reciprocity law applicable is the following:

$$m^2 \equiv 1 \bmod 24$$

for all integers m with $\gcd(m, 24) = 1$.

Additionally, 24 is the largest number with this property. Using higher reciprocity laws the square m^2 can be replaced by a higher power, which allows the number 24 to be replaced by a much larger number. For example,

$$m^{12} \equiv 1 \bmod 65520$$

for all integers m with $\gcd(m, 65520) = 1$, and

$$m^{5040} \equiv 1 \bmod s$$

for all integers m with $\gcd(m, s) = 1$, where s is a number of 53 digits:

$$s = 153219867888544432846626127356636113800104312257712 00.$$

In attempting to prove the primality of a number n, a theorem similar to Theorem 2 can be applied to yield the information that for every factor r of n there exists an integer i such that $0 \le i < 5040$ and

$$r \equiv n^i \bmod s,$$

where s is the number of 53 digits above. This is much stronger information on possible factors of n than before, and if n has no more than 100 digits, it can readily be used to complete the proof that n is prime. For a larger n, use similar results would be used for even larger values of s. The essential ideas of the method described above are attributed to L.M. Adleman, C. Pomerance, and R.S. Rumely (1983). The test has an excellent performance in practice, especially when used in conjunction with the Lucas-Lehmer type tests.

In more recent years a different class of primality testing algorithm has been developed. This depends on the arithmetic theory of elliptic curves, which was started by H. Hasse and M. Deuring in the 1930s and is still of central importance in number theory and algebraic geometry. In 1986, A.O.L. Atkin found a primality test of this nature that in practice is competitive with the test of Adleman et al. sketched above. Still more recently, L.M. Adleman and M.D. Huang proposed a primality test that employs abelian varieties, which are higher dimensional analogues of elliptic curves. Their algorithm does not have practical value, but it can be used to prove that, in the jargon of theoretical computer science, primality can be decided in random polynomial time. The proof employs techniques from algebraic number theory, analytic number theory, and arithmetic algebraic geometry, and it is a veritable tour de force.

2. Telephone Poker

For a number theorist, the respectability of number theory does not depend on its usefulness in real life; and if, within number theory, the opinion has always prevailed that hunting for large primes is no more than a frivolous pastime, we cannot expect this attitude to change over night when an application to cryptology is discovered. Accordingly, we shall, in this section, not discuss the most serious use of large prime numbers in cryptology, but the most frivolous one. It was proposed in 1981 by A. Shamir, R.L. Rivest, and L.M. Adleman.

Two players, Andrew and Beatrix, want to play poker over the telephone. Since it is impossible to send playing cards over the phone line, the entire game must be realized using only spoken messages between the two players. In keeping with the spirit of the game, we assume that neither player is beyond cheating.

The game must begin with a fair deal. To accomplish this, the players exchange a sequence of messages according to some procedure. This procedure should ensure that after the deal is over each player knows which cards are in his hand without having any information about his opponent's hand. In addition, the hands should be disjoint, and all possibilities should be equally likely. Additional procedures must be designed for actually playing the game and for checking at the end that the game was played fairly, but we shall restrict attention to how a fair deal can be accomplished.

Before explaining the role of large prime numbers, we consider a physical analogy in which actual playing cards are used and the telephone line is replaced by the United Parcel Service. In that case the cards can be dealt as follows. Andrew buys 52 indistinguishable boxes, puts 1 card in each box, and locks all boxes with padlocks so that Beatrix will not be able to inspect the contents of the boxes. Next he ships all 52 boxes to Beatrix by UPS. Beatrix selects 5 boxes to return to Andrew for his hand. In addition, she selects five boxes for her own hand, and adds to these her own padlock to the clasp ring. These five boxes are also returned to Andrew. Finally, Andrew removes his padlock from all ten boxes and returns to Beatrix as her hand those that still have her padlock. Although this procedure has practical problems (e.g., how to obtain indistinguishable boxes), it is clear that in principle it can accomplish a fair deal.

We next consider the problem of how to simulate the procedure over the phone line. For simplicity, we shall assume that Andrew and Beatrix have computers at their disposal. We shall also assume that Andrew and Beatrix have agreed on some fixed way to read the card names as numbers. For example, they may use the official numbering proposed by the "International Telephone Poker Union":

$2\spadesuit = 200701$	$2\heartsuit = 200702$	$2\diamondsuit = 200703$	$2\clubsuit = 200704$
$3\spadesuit = 300701$	$3\heartsuit = 300702$	$3\diamondsuit = 300703$	$3\clubsuit = 300704$
...	...	...	...
$K\spadesuit = 1300701$	$K\heartsuit = 1300702$	$K\diamondsuit = 1300703$	$K\clubsuit = 1300704$
$A\spadesuit = 1400701$	$K\heartsuit = 1400702$	$A\diamondsuit = 1400703$	$K\clubsuit = 1400704.$

Next, Andrew and Beatrix choose a big prime number n, of about 150 digits. This can be done by testing a few numbers of that size for primality, until a prime number is found. The messages that Andrew and Beatrix are going to exchange are numbers less than n, for example, card numbers, or card numbers that have been locked away.

How can cards that are numbers be locked away? This is done using a padlock that also consists of numbers. By a padlock we mean a pair of positive integers a and a' that satisfy

$$aa' \equiv 1 \bmod (n-1).$$

Such a padlock is used as follows. Suppose that m is a message that Andrew wishes to lock away; so m is a number between 0 and n. To lock m away, Andrew replaces m with $m^a \bmod n$ (the remainder of m^a after division by n). We saw above that there is a good way to calculate this remainder. Anyone not knowing a, a' is unable to recover m from $m^a \bmod n$. However, Andrew can recover m from $m^a \bmod n$ by using a' in the same way he used a, i.e., by replacing $m^a \bmod n$ with the remainder of its a'-th power after division by n. The claim is that this remainder is m itself:

$$(m^a)^{a'} \equiv m \bmod n$$

so that Andrew indeed recovers m. To prove the claim, we remark that $(m^a)^{a'} = m^{aa'} = m^{1+t(n-1)}$ for some integer t, and that from a repeated application of Fermat's theorem it follows that

$$\begin{aligned} m &\equiv m^n = m^{1+(n-1)} = m \cdot m^{n-1} \\ &\equiv m^n m^{n-1} = m^{1+2(n-1)} = m \cdot m^{2(n-1)} \\ &\equiv m^n m^{2(n-1)} = m^{1+3(n-1)} = m \cdot m^{3(n-1)} \\ &\equiv \ldots \equiv m^{1+t(n-1)} \bmod n, \end{aligned}$$

for all m and t.

Before the deal starts, Andrew gets himself a padlock a, a'. To do this, he tries a few values for a, until one is found for which $\gcd(a, n-1) = 1$. This gcd can be calculated by means of an algorithm that goes back to Euclid. The same algorithm can also be used to calculate an integer a' with $aa' \equiv 1 \bmod (n-1)$, which is the other half of the padlock. Andrew keeps his padlock secret from Beatrix; in particular, he should choose a large m, so that Beatrix cannot guess it. Likewise, Beatrix gets herself a padlock b,b', which she keeps secret from Andrew.

After these preparations, Andrew and Beatrix proceed exactly as in the physical analogy presented above.

Andrew takes the 52 card numbers $m_1 = 200701, \ldots, m_{52} = 1400704$, locks them away by calculating $(m_1)^a$ mod $n, \ldots, (m_{52})^a$ mod n, and sends these to Beatrix. Beatrix selects five of these numbers for Andrew, and five for herself. The latter five she also locks away herself, i.e., those five $(m_i)^a$ mod n she replaces with $((m_i)^a)^b$ mod n. She returns the ten numbers to Andrew, who uses a' to remove his own lock from all ten numbers. The first five are now card numbers, and they form his hand. The other five still have Beatrix's lock. They are sent to Beatrix, who uses b' to find which cards m_i she dealt herself. This concludes the description of the deal.

For the deal as described to be fair, it must be supposed that Beatrix, knowing both the card numbers $m_1, \ldots, m_{52}$ and their a-th powers modulo n, is unable to compute a, since otherwise she could simply select good cards for herself and bad cards for Andrew. There is no guarantee that Beatrix cannot do this. All we know is that no one has so far been able to come up with a method that computes a in the given situation within a reasonable period of time, say less than a year. So if Beatrix wishes to cheat in this way she must be more clever than everyone who worked on this problem until now. The probability of this is for Andrew to decide.

There are a few other obvious ways in which one could conceivably cheat. However, it turns out that these give rise to computational problems in number theory that, given the current state of knowledge, are equally intractable.

There are also a few less obvious ways to cheat, by which the players can obtain information about each other's hands that they are not supposed to have. These lead to the recommendation not to use the above procedure in practice without taking some rather drastic precautions, and perhaps even not to use it at all. There are similar but more complicated procedures that do not have this shortcoming, and it may well be that the only use of the scheme that we described is for exposition.

3. Number Theory and Cryptology

Cryptography used to be the art of providing secure communication over insecure channels, cryptanalysis the art of breaking into such communications, and cryptology the combined arts of cryptography and cryptanalysis. In more recent times one would consider cryptology not only as an art, but also as a science. In addition, the most recent techniques of cryptology have applications that reach far beyond the establishment of secure communication over insecure channels. Here is a list, neither complete nor systematic, of seemingly impossible problems for which modern cryptology offers solutions.

Public key cryptography. Andrew wants to send confidential information over an insecure channel to Beatrix, but they did not agree upon a secret key beforehand. How to proceed? A public key cryptosystem allows Andrew to check a public directory to find Beatrix's key, and to use this key for encrypting his message. Even though a potential eavesdropper knows both the key and the encrypted message, it is computationally unfeasible for him to recover the original message. Only Beatrix can do so, by using information that she alone knows.

Authentication and digital signatures. The purpose of authentication is to convince Beatrix that the message she receives comes actually from Andrew and not from someone else who pretends to be Andrew. If Andrew digitally signs his message, afterwards he cannot deny that he sent it.

User identification. Currently, a would-be user of an automatic teller machine proves his identity by means of his PIN code. For this system to be invulnerable, it is desirable that the machine can validate this PIN code without having access to a table of PIN codes, since the very existence of such a table is a threat to the system. In addition, it is desirable that the machine is never told the PIN code; for example, to prevent others from seeing it keyed in. Modern cryptology can satisfy these seemingly paradoxical requirements.

Showing credentials without identification, protection of privacy. Business and government organizations sometimes need to see statements about individuals that are issued by other organizations: for example, passports or driver's licenses. Can an individual show his credentials without disclosing his identity? There exists a system making this possible that does away with all manner of personal data about individuals being stored in computerized files. It is claimed to protect both the individual's rights and the legitimate interests of organizations better than the current system.

Minimum disclosure proofs. Andrew has proved Fermat's Last Theorem, and he wants to convince others of this fact without giving away any hint as to how he did it. With cryptological techniques he can do this, except that he will give information about the length of his proof.

Coin flipping. Andrew and Beatrix have just divorced, and live in different cities. To decide who gets the car they want to flip a coin by telephone. How can this be done fairly?

The millionaires' problem. Two millionaires wish to know which is the richest. Can they find out without either knowing the other's wealth?

A common property of the solutions to all these problems is that they depend on number theory in two ways. First, good methods exist that can be used to solve certain computational problems in number theory by computer. We encountered the two main examples already: the primality testing problem and the problem to calculate $a^m \bmod n$ for given a, m, and n. By means of these methods, one can efficiently both construct an encryption algorithm and apply it to the encryption of messages. Second, for certain other, apparently closely related, computational problems no feasible solution is known. The two main examples are the factoring problem, mentioned in Primality Testing, and the discrete logarithm problem, which given a, n, and $a^m \bmod n$ asks for m. The intractability of these problems is important for the security of the system; someone who can factor numbers or compute discrete logarithms in an efficient manner can break many systems that have been proposed in cryptology.

It may be dangerous to base all of cryptology on just a few computational problems from number theory; but currently number theory is the only branch of mathematics that offers the combination of hard and easy problems that cryptology needs. Widespread introduction of modern cryptographic schemes will transform the area of computational number theory from a puzzle corner in a confirmed pure discipline into a branch of science that is of critical importance for the well-being of the "information based society."

Number theory plays a crucial role in many unsolved problems in cryptology. Of vital importance is the problem of finding evidence of the security of a cryptographic system.

In the days when cryptology was only an art and not yet a science, this was a very weak spot. A cryptographic scheme was considered secure if a relatively small team of specialists, whose work was classified, could not break it. Nowadays cryptology is studied in the open, and it is on its way to becoming a recognized academic discipline. This multiplies, in the first place, the amount of intellectual energy brought to bear on the problem by a sizable factor, which leads to a corresponding increase in the confidence one may have in the security of a system. Also, if a system is broken, given the current state of affairs, it is much more likely than before to be immediately publicized, so that the use of the compromised system can be discontinued and the damage done is minimized. Second, and more importantly, modern cryptology develops a conceptual framework in which notions such as unbreakability and security are given precise mathematical meanings, so that in principle it becomes possible to prove that a system is secure as far as its mathematical aspects are concerned.

It is important to realize that this program is at the moment still faced with considerable theoretical difficulties, and that it is not yet beyond the state of wishful thinking and daydreams. For the completion of this program, it would first be necessary to prove the famous $P \neq NP$ conjecture from theoretical computer science. Even that would not be enough. The edifice of theoretical cryptology is being erected on quicksand.

Although the situation leaves much to be desired, it is also clear that it is much better than before. As an example, a typical rigorously proved result from present-day cryptology reads as follows: if a certain cryptographic scheme is unsafe, then an allegedly difficult number theoretic problem such as factoring admits a solution. Thus, rather than letting the evidence for the security of a system depend on the ingenuity of a handful of individuals whose efforts to break it remain fruitless, one now establishes a link to a classical and well-studied problem, for which generations of number theorists have been unable to find a solution.

Is there good evidence that factoring is an intrinsically hard problem for which a good solution will never be found? The only available evidence is of a historical nature. Many people worked on it for a long time, and the best that was found is not good enough. There is no consensus among experts on factoring that this is convincing evidence, and the evidence looks far too weak to let the hardness of the factoring problem serve as essentially the only basis for a branch of science whose consequences are as far-reaching as is the case with cryptology.

One of the main weaknesses of the historical argument is that during a long and important period in the development of number theory (from the early nineteenth century to about ten years ago), the factoring problem did not have good standing among leading number theorists. The people who did work on the problem formed an isolated group, and until after the Second World War, they did not have the perspective that is offered by the existence of electronic computers. Only relatively recently have techniques from the full spectrum of contemporary number theory been used for computational purposes: analytic number theory, algebraic number theory, and algebraic geometry, as illustrated in section 1, Primality Testing. When, in the early 1980s, a geometric technique that Minkowski introduced in number theory in the late nineteenth century was brought to the attention of computer scientists, it instantly led to the solution of several computational problems that had been open for some time. One of these was the problem of breaking a certain cryptographic scheme, others belonged to the areas of operations research, combinatorial optimization, and number theory itself. Often in number theory, a problem on which many of the finest minds had been working for a long time finally found a solution. The history of number theory does not give a good reason to be surprised if the same happens to the factoring problem.

For these reasons, the truly pure mathematician will see the large-scale introduction of current cryptographic schemes as a means to invoke the help of all of mankind for the solution of the factoring problem. If, in the year 2020, a small Spanish girl finds a good method to factor integers into primes, she will do so to the delight of all number theorists, who will rejoice on the ashes of the information based society.

Bibliography

Brassard, G. 1988. Modern cryptology, a tutorial. Lecture Notes in Computer Science. Vol. 325. New York: Springer-Verlag.

Knuth, D.E. 1981. The Art of Computer Programming. Vol. 2, second edition. Reading: Addison-Wesley.

Lenstra, A.K. and H.W. Lenstra, Jr. In press. Algorithms in number theory. Handbook of Theoretical Computer Science, J. van Leeuwen et al., eds. Amsterdam: North-Holland.

Lenstra, H.W., Jr. 1983. Fast prime number tests. Nieuw Archief voor Wiskunde. Ser. 4, 1:133-144.

Lenstra, H.W., Jr., and R. Tijdeman. 1984. Computational Methods in Number Theory. Vols. 1 and 2. Amsterdam: CWI.

Shamir, A., R.L. Rivest, and L.M. Adleman. 1981. Mental poker. *In* The Mathematical Gardner, D.A. Klarner, ed. Belmont: Wadsworth International.