IV. Update to the Introduction for the Second Edition*.

The tables of the first edition contained all factors known to the authors on October 23, 1982. Since then more than two thousand new factorizations have been discovered. Appendix C lists the smallest composite cofactors in the tables. In the first edition it contained numbers with 51 to 64 digits. Now it gives numbers of 80 to 100 digits. The lists of "wanted" factorizations in the first edition had 25 numbers with 52 to 71 digits. They have all been factored. Other "wanted" lists have since been issued and many of their entries have been factored. The "wanted" lists (of the second edition) now contain numbers with 86 to 291 digits. All remaining numbers $b^n \pm 1$ with exponent $n < 100$ now appear on the "wanted" lists. All but nine of the numbers for base $b > 2$ considered in [**11**] have been completely factored!

The smallest probable prime (PRP) in Appendix A of the first edition had 54 digits. Prime proofs have now been completed for all numbers up to 221 digits. In this edition we have updated the tables and appendices to June 22, 1987, and reviewed the developments in technology, factorization and primality testing which have produced the recent advances. We also include a few references to recent work which, though it has not contributed to this edition, may produce results in the future.

Since some of the first edition tables had very few composite entries, and since now most of the Aurifeuillians in those tables with base > 2 have been factored, we decided to extend the higher base tables in the second edition. The numbers we have added to these tables have been factored with about the same effort that was applied to numbers in the first edition. The factoring for these extensions was done mostly by Robert Silverman, Peter Montgomery and SSW. Some factors of $10^n \pm 1$ came from Samuel Yates [**262**] and his updates.

The tables of the first edition were found to be nearly free of errors. The most interesting error was the composite number 1223165341640099735851, which was listed as a *prime* factor of $6^{175} - 1$. A. O. L. Atkin found that this number is 34840572551.35107498301. Other errors were the line references in the parentheses of the first lines of 3,399+ and 11,209+. The first digit of k for the second factor of F_7 was missing. These errors have been corrected in the present edition.

The format of the tables and appendices has been changed a little in this edition. In the first edition the decimal digits of a prime factor appeared in the main tables if it had no more than 25 digits; otherwise it was placed in Appendix A. In the first edition no penultimate prime factor had more than 25 digits. In the present edition many penultimate prime factors have more than 25 digits and they are given in full in the main tables. Only final prime factors are given in Appendix A. On the other hand, final prime factors with 21 to 25 digits have been placed in Appendix A when a line could be saved in the main tables. At the beginning of each table final factors are given in full as long as they fit, so as to enhance the pleasing triangular shape of the beginning of each table.

*The text of this part of the Introduction to the second edition is essentially that used in the second edition. In this section the word "now" means, "on June 22, 1987." A few typographical errors were corrected and the old status report was deleted. The new text for the third edition appears in Section **V**.

The inclusion of so many large new prime factors forced many more factorizations to be split into two lines in the main tables. When it was necessary to break a factorization we aligned the second line as follows, unless it was very long: If the factorization was incomplete, put the "C" in the column for "C"'s. The last digit of a broken complete factorization appears two columns to the left of the "C"'s. (In the first edition we tried to align a dot in the second line with a dot in the first line.)

A. Developments Contributing to the Second Edition.

In this section we list some of the advances in technology, factoring algorithms and primality testing which made the second edition better than the first one.

1. Developments in Technology.

The use of personal computers and supercomputers for factoring has continued as has the construction of special machines designed for factoring. The use of networks of microcomputers to factor numbers is a new development. More memory has become available in modern computers due to its dramatically reduced cost. Meaningful error messages are now generated when an error occurs during the execution of overlapping instructions.

(a) *Supercomputers.* Davis and Holdridge [**215**] at Sandia National Laboratories used a Cray-1 and a Cray X-MP to obtain all of the "Ten Most Wanted" factorizations of the first edition. They made the first implementation of the quadratic sieve factoring algorithm on a supercomputer.

McCurdy and Wunderlich [**260, 261**] have programmed the continued fraction algorithm on the MPP computer. With this machine, they factored the 62-digit composite divisor of $5^{171} + 1$.

Herman te Riele et al. [**229**] have programmed the quadratic sieve algorithm on a Cyber 205. They factored the 82-digit number $(7^{104} + 1)/(7^8 + 1)$ and several smaller numbers.

Young and Buell [**263**] used a Cray-2 to determine that the Fermat number F_{20} is composite. They checked this calculation with a Cray X-MP.

Many more general purpose supercomputers of various designs are being built, which should make it possible to factor even larger numbers.

(b) *The Extended Precision Operand Computer.* With the assistance of many students at the University of Georgia, J. W. Smith and SSW [**240, 249, 253**] built a special processor, the Extended Precision Operand Computer (EPOC), to factor numbers with the continued fraction algorithm of **III B 2**(d). This machine has a 128-bit word length and several remaindering units (the "Mod Squad") to perform the trial divisions quickly. Its results include the factoring of the 62-digit primitive part of $3^{204} + 1$. The elliptic curve method (see **B 2**(d) below) is presently being programmed on the EPOC.

(c) *The Dubner Processor.* Dubner and Dubner [**218**] built a special computer which rapidly performs arithmetic with large integers. Their greatest success has been in finding small divisors of large numbers by Brent's variation [**204**] of Pollard's "Rho" method and by the elliptic curve method. (See [**257**] for an account of how this machine assisted in the proof that $(10^{1031} - 1)/9$ is prime.)

(d) *Small Machines.* Hiromi Suyama continues to factor numbers from the Cunningham Project using his microcomputer. Yûji Kida found the 30-digit prime

divisor of $7^{127} - 1$ with his NEC PC-9801VM2 personal computer. There is little doubt that with the increase in power of small computers many factorizations will continue to be discovered on these machines.

The host computer for the EPOC (see (b) above) is an IBM PC. The host prepares the factor base and initializes the continued fraction expansion of $\sqrt{mN}$. It transmits this information to the EPOC and starts the EPOC. When the EPOC factors a Q (see **2**(c) below), it sends the pair A, Q to the host to be stored. After the host has collected enough of these pairs, it computes the null space of a large matrix and factors N.

(e) *A Distributed Network of Small Computers.* Silverman [**247, 248**] has factored many large numbers using the quadratic sieve algorithm running on a network of SUN microcomputers at MITRE Corporation. The master processor assigns a different polynomial to each SUN. After the SUN sieves the range of this polynomial, it reports the results back to the master processor, which then determines whether there is likely to be enough information to factor the number. The most difficult number Silverman has factored so far is the 87-digit number $(5^{128} + 1)/514$. At present, this is the largest number ever factored by a general purpose factoring algorithm, i.e., one which can factor all numbers of a given size in about the same time.

2. Developments in Factorization.

Several methods which were discussed in the first edition have been improved. They include Pollard's methods and the continued fraction method. The quadratic sieve method was just mentioned in the first edition, because it had hardly been used at that time, although it had been used a great deal by M. Kraitchik as a hand method and was cited in the first edition in **III B 1**(a). It has now been programmed and has been much advanced in the past five years. A completely new method, which uses elliptic curves, has had a major effect on the tables of this edition. (See [**208**] for a comparison of the best factoring methods and for recommended choices of their parameters.) Some good general references for recent progress in factoring and prime testing are [**217, 225, 236, 242, 243, 256**].

(a) *More about Pollard's Methods.* As was predicted in **III B 2**(e), Pollard's $p - 1$ [**80**] and "Rho" [**81**] methods have been of great importance in factoring numbers in these tables.

Baillie has completed his search for factors for all numbers in the Cunningham Project using the two-step $p - 1$ method with limits 200000 and 10200000. The largest factor he found was $p = 1744633866571915160339326144 01$, which divides $2^{740} + 1$. (Note that $p - 1 = 2^8.5^2.17.37.1627.5387.68111.152081.477361$.) Baillie's efforts have contributed hundreds of factors to this book.

See Williams [**255**] for the $p + 1$ analogue of the $p - 1$ method. (Cf. **III B 2**(e).)

See [**204, 206**] for an account of Brent's improvement of Pollard's Rho method, which factored F_8 just before the first edition went to press. Dubner [**218**] has since factored many numbers by the Rho method.

Montgomery [**231**] has proposed other variations of Pollard's methods.

(b) *More about the Continued Fraction Method.* Just before the first edition was published, Carl Pomerance [**237**] made a substantial improvement to the continued fraction method by optimizing the "early abort" strategy in it. (An "early abort"

strategy determined when to stop the trial divisions of a given residue part way through the factor base if not enough factors have been found at that point.) His analysis predicted more precisely where in the factor base one should examine the progress made so far and how small the remaining cofactor should be if one is to continue work on this residue. SSW programmed this strategy on an IBM computer at the University of Georgia and factored many numbers in the 50 to 54-digit range. The same algorithm now runs on the EPOC (See **A 1**(b) above). For factoring numbers in the 50 to 60-digit range, the early abort strategy cuts the running time by about an order of magnitude. (See [**241**] for a good choice of the parameters.)

One way to implement the continued fraction factoring algorithm on a parallel computer is to compute terms in widely-spaced intervals in the continued fraction expansion of $\sqrt{mN}$. After this is done, each processor of the parallel computer can work independently on its own section of the continued fraction expansion. Williams and Wunderlich [**259**] explain how to do this by jumping ahead in the expansion.

(c) *Kraitchik's Method (The Quadratic Sieve Algorithm).* Only one factorization reported in the first edition was obtained by the quadratic sieve algorithm, namely that obtained by Joseph Gerver [**219**] who factored the 47-digit composite divisor of $3^{225} - 1$ on an HP3000. A hand version of this method had been used extensively by M. Kraitchik [**34, 35, 37**] to factor many numbers in older tables. Although the method with its variants was well-known to JB and DHL, no one had used it in modern times to factor numbers, until Carl Pomerance [**237, 238**] re-discovered the method, analyzed it theoretically and found it was a powerful method. Others have re-discovered some of the techniques of Kraitchik, such as matching large cofactors of the residues, and have contributed new ideas and programming techniques to the point where now this algorithm has factored most of the composite numbers with 54-79 digits which remained in the first edition. (Some numbers in this range were factored by other algorithms before the quadratic sieve method was used.)

Davis and Holdridge [**216**] used it to factor several numbers having 53 to 69 digits on a Cray-1. They also factored $(10^{71} - 1)/9$ with it on a Cray X-MP. Silverman [**247, 248**] implemented the algorithm first on a VAX and then on a network of SUN's. He factored hundreds of numbers of 54 to 87 digits from this project. His effort was the major force which advanced the lower limit of Appendix C from 54 to 80 digits. Niebuhr, te Riele and SSW have factored a few numbers with this algorithm.

The quadratic sieve algorithm is similar to the continued fraction algorithm in that both algorithms generate pairs A, Q with $A^2 \equiv \pm Q \pmod{N}$, where N is the number to be factored. In both algorithms some of these congruences with Q factored are multiplied to construct congruences $X^2 \equiv Y^2 \pmod{N}$ which yield factors of N as $\mathrm{GCD}(X+Y, N)$. (This is Kraitchik's [**36, pp. 147–151**] method of "cycles".) The A-Q pairs in the continued fraction algorithm arise in the continued fraction expansion of $\sqrt{mN}$ and have $0 < Q < 2\sqrt{mN}$. The small size of the Q's improves their chance of being factored completely by trial division by the primes in the factor base. The Q's in the quadratic sieve algorithm are numbers in the range of one or more quadratic polynomials. Although most of them are larger

than $2\sqrt{mN}$, this disadvantage is more than offset by the fact that they may be factored by sieving rather than by trial division.

Gerver [**219**] used just the single polynomial $Q(x) = (x + \lfloor\sqrt{N}\rfloor)^2 - N$, sieving it over the interval $-499999999 < x < 400000000$ in blocks of 10000 x's. However, $|Q(x)|$ becomes large when x is far from 0. Davis and Holdridge began by using this polynomial, but found [**214**] a way to construct other polynomials whose values are divisible by a specified prime beyond the factor base. They used this technique to match some otherwise unmatched "large primes". (See [**207, p. 42**].) Then Montgomery (see [**238**]) found an elegant way to construct many polynomials with slightly smaller average values. Silverman [**247**] implemented Montgomery's version.

(d) *The Elliptic Curve Factoring Method.* On February 14, 1985, H. W. Lenstra, Jr., announced the first factoring algorithm to use twentieth century mathematics. This method, called the Elliptic Curve Method (ECM), computes a high multiple of a point on a random elliptic curve modulo the number N to be factored. During this calculation one hopes to encounter a noninvertible denominator modulo N and thereby discover a proper factor of N in the unsuccessful extended GCD computation. Like the Pollard methods, ECM tends to find small prime factors of N before large ones. However, for most primes Lenstra's method is even faster in practice than those of Pollard. (See [**228, 202, 205, 210, 231, 253**] for more details about ECM.)

The choice and parametrization of the curves are important issues. Chudnovsky and Chudnovsky [**210**] and Suyama have suggested ways to select and parametrize elliptic curves having special properties which accelerate the algorithm. Montgomery [**231**] considered several versions of a second step for ECM, analogous to that of the $p - 1$ method. Although the second step does not speed ECM theoretically, it has important practical value, as most interesting factors are discovered during the second step. Brent [**205**] suggested a "birthday paradox" variation of the second step of ECM.

Several researchers have programmed ECM. The most effective version so far has been that of Montgomery [**231**] which has produced hundreds of factors for the tables in this book. Although Montgomery found most of them, Silverman and SSW found some others with Montgomery's VAX program. Silverman has also written an ECM program for SUN workstations. Brent has a version of ECM, too. Suyama and Kida have found a number of factors by ECM on their own microcomputers.

(e) *Other New Factoring Methods.* Schnorr and Lenstra [**246**] have published a factoring algorithm which requires little storage. It will factor N efficiently if the class number $h(-mN)$ is free of large prime divisors for some small multiplier m. Buell [**209**] has investigated the probability that $h(-mN)$ will have this property. Atkin has factored several numbers in this book by a practical class number algorithm he calls "SPAR".

The continued fraction method and Kraitchik's method have a step in which one computes the null space of a huge matrix over $\mathbf{GF}(2)$. This time-consuming elimination step limits the size of the factor base which may be used. Several researchers [**235, 254**] have suggested techniques for speeding up this step.

Multiplication of numbers modulo n occurs frequently in factoring and prime testing programs. Montgomery [**230**] has found a way to compute $ab \pmod n$

quickly when a, b and n are large numbers and the computer's divide instruction is slow compared to its multiplication instruction.

3. Developments in Primality Testing.

In the first edition we mentioned a new, but unused, primality testing method which is (a) below. The elliptic curve methods in (b) were developed after the first edition was published. In **III B 3**(b) we mentioned Williams' primality theory which utilizes factors of $\Phi_k(N)$, for $k = 1$, 2, 3, 4 and 6, to prove that N is prime. H. W. Lenstra, Jr., [**226, 227**] has extended this theory to all cyclotomic polynomials and has related it to the APR primality test described below.

(a) *The Method of Adleman, Pomerance and Rumely.* These three researchers [**201**] invented a new method for testing a number for primality. Later, Cohen and H. W. Lenstra, Jr. [**212**] transformed this test into a practical primality test. Cohen and A. K. Lenstra [**211**] implemented this test on various computers. With the version of the program which A. K. Lenstra left at Bell Laboratories, Odlyzko has proved the primality of all numbers in Appendix A up to 210 digits. He stopped at 210 digits because larger numbers would require much larger values of the parameters E and I (mentioned in the next paragraph) and hence much longer running times. The proof for a 200-digit prime takes only a few minutes on a Cray-1.

The Adleman-Pomerance-Rumely algorithm (APR) begins by subjecting N to a series of tests similar to the usual probable prime tests. If N fails any one of these tests, then N is composite. But if N passes them all, then N either is prime or is divisible by one of the numbers $N^j \bmod E$ for $j = 0, 1, \ldots, I-1$, where E is an integer slightly greater than $\sqrt{N}$ and I is an integer $< (\log N)^{c \log\log\log N}$ for some constant c. The algorithm concludes by checking that N is not a multiple of one of these numbers. The only way to verify an APR proof of primality is to repeat all of its steps. No information about the proof will shorten the work needed to verify it. By contrast, the hints in Appendix B (other than search limits) are real short-cuts to checking proofs of the type described in **III B 3**. (Actually, some versions of the APR test can use divisors of $N^2 - 1$ to reduce the size of I. Hints like the ones in Appendix B would reduce the work needed to check this type of APR proof.) Nothing was added to Appendix B to indicate APR proofs. Numbers listed as "P" (instead of "PRP") in Appendix A, but for which there is no proof in Appendix B, were proved prime by APR, mostly by Odlyzko. Using another version of his program, A. K. Lenstra has proved primality of some primes $>$ 210 digits. (See Rumely's excellent survey paper [**245**] for a concise overview of the APR method. See [**211, p. 120**] for the details of the APR proof of the primality of the 247-digit divisor of $2^{892} + 1$.)

(b) *Elliptic Curve Primality Tests.* Several researchers [**203, 210, 220**] have invented primality tests which use the theory of elliptic curves. The basic idea for showing that N is prime by these tests is to show that, for any prime factor $p \le \sqrt{N}$ of N, there is an elliptic curve defined over $\mathbf{Z}/p$ which has more points than allowed by the Hasse-Weil theorem, that is, more than $p + 2\sqrt{p} + 1$ points.

Atkin (see 4.12 of [**225**]) has developed a practical primality test based on elliptic curves. He has used it to prove the primality of several large numbers in Appendix A, but as yet has not published his method.

(c) *Other Changes and Additions to Appendix B of the Second Edition.* Because of the power of the new primality tests mentioned in (a) and (b), we lack rigorous

prime proofs for only 35 numbers in Appendix A (marked "PRP" there). The smallest one has 222 digits.

Since only the APR test was used on all the new primes smaller than 211 digits that were added to Appendix A since the first edition, some quite small numbers would have no proof summary even though it would be possible to find one easily by the methods of **III B 3**. We decided to provide such proofs where we could do so with little effort. Because the new factoring algorithms mentioned above are so powerful we could construct such proofs for nearly all primes in Appendix A up to 100 digits. At the same time, in order to shorten the work of those who might check the proof summaries in Appendix B, we simplified many of the old proofs, especially those with a search limit $> 10^6$. Since we wrote a new program to generate these proofs, we did not confine ourselves to just the PPL and CMB proofs constructed by the DOWNRUN program of **III B 3**(b). We also used the powerful Theorem 7 of [**7**], which is abbreviated BLS7 in Appendix B. It is the same as Theorem 11 of **III B 3**(a)(11).

The proof summaries in Appendix B are not guaranteed to be as short as possible.

B. Acknowledgements for the Second Edition.

We wish to thank the following people who have contributed to the Second Edition:

New factors of numbers in the main tables were discovered by A. O. L. Atkin, R. J. Baillie, R. P. Brent, J. A. Davis, H. Dubner, D. B. Holdridge, W. Keller, Y. Kida, K. McCurdy, P. L. Montgomery, W. Niebuhr, N. W. Rickert, R. Silverman, J. W. Smith, H. Suyama, H. J. J. te Riele, M. C. Wunderlich and SSW. The new factors of Fermat numbers were found by R. J. Baillie, G. B. Gostin, W. Keller and H. Suyama.

J. W. Tanner wrote a program which checked the new factors and inserted them into the tables. M. Senn wrote a program to format the tables. The new prime proofs were supplied by A. O. L. Atkin, A. K. Lenstra, A. Odlyzko, H. Suyama, D. Tormey and SSW.

The new results of the second edition required hundreds of thousands of hours of computer time. We are grateful to the directors and staffs of the following computer centers which provided this time: Australian National University Computer Centre; Bell Telephone Laboratories, Murray Hill; Dutch Organization for the Advancement of Pure Research and Control Data Corporation, Netherlands; MITRE Corporation's research computer facility and its Bedford Computer Center; MPP computer facility at Goddard Space Flight Center, Greenbelt, MD; Purdue University Department of Computer Sciences; Rechenzentrum der Universität Hamburg; Sandia National Laboratories, Albuquerque, NM; Supercomputing Research Center, Lanham, MD and their Cray hardware engineers; Unisys Research and Development Computer Facility, Santa Monica and Advanced Research Center, Huntsville; University of Georgia Office of Computing and Information Services; University of Illinois Computer-Based Education Research Laboratory; and the University of Illinois at Chicago Computer Center.

SSW gratefully acknowledges the support of the National Science Foundation in the preparation of this edition and of the annual updates.

References

The first edition had references 1-118; they appear in **III E**.

201. L. M. Adleman, Carl Pomerance and R. S. Rumely, *On Distinguishing Prime Numbers from Composite Numbers*, Ann. of Math. **117** (1983), 173–206, MR 84e:10008.
202. Eric Bach, *Lenstra's Algorithm for Factoring with Elliptic Curves, Exposé*, Computer Sciences Department, University of Wisconsin, Madison, February, 1985.
203. W. Bosma, *Primality Testing Using Elliptic Curves*, Report 85–12, Mathematisch Instituut, Universiteit van Amsterdam, 1985.
204. R. P. Brent, *An Improved Monte Carlo Factorization Algorithm*, BIT **20** (1980), 176–184, MR 82a:10007.
205. R. P. Brent, *Some Integer Factorization Algorithms Using Elliptic Curves*, Research Report CMA-R32-85, The Australian National University, Canberra, September, 1985.
206. R. P. Brent and J. M. Pollard, *Factorization of the Eighth Fermat Number*, Math. Comp. **36** (1981), 627–630, MR 83h:10014.
207. John Brillhart, *Fermat's Factoring Method and its Variants*, Congressus Numerantium **32** (1981), 29–48, MR 84f:10009.
208. John Brillhart, Peter L. Montgomery, and Robert D. Silverman, *Tables of Fibonacci and Lucas Factorizations*, Math. Comp. **50** (1988), 251–260, MR 89h:11002.
209. Duncan A. Buell, *The Expectation of Success Using a Monte Carlo Factoring Method—Some Statistics on Quadratic Class Numbers*, Math. Comp. **43** (1984), 313–327, MR 85k:11068.
210. D. V. Chudnovsky and G. V. Chudnovsky, *Sequences of Numbers Generated by Addition in Formal Groups and New Primality and Factorization Tests*, Research report RC 11262 (#50739), IBM Research Center, Yorktown Heights, July, 1985.
211. H. Cohen and A. K. Lenstra, *Implementation of a New Primality Test*, Math. Comp. **48** (1987), 103–121, MR 88c:11080.
212. H. Cohen and H. W. Lenstra, Jr., *Primality Testing and Jacobi Sums*, Math. Comp. **42** (1984), 297–330, MR 86g:11078.
213. D. Coppersmith, A. M. Odlyzko and R. Schroeppel, *Discrete Logarithms in* $\mathbf{GF}(p)$, Algorithmica **1** (1986), 1–15, MR 87g:11167.
214. J. A. Davis and D. B. Holdridge, *Factorization Using the Quadratic Sieve Algorithm*, *Advances in Cryptology, Proceedings of Crypto 83*, David Chaum, ed., Plenum Press, New York, 1984, pp. 103–113, MR 86j:11128.
215. J. A. Davis and D. B. Holdridge, *Most Wanted Factorizations Using the Quadratic Sieve*, Sandia National Laboratories Report SAND 84-1658, August, 1984.
216. J. A. Davis, D. B. Holdridge and G. J. Simmons, *Status Report on Factoring* (*at the Sandia National Labs*), *Advances in Cryptology, Proceedings of EUROCRYPT 84*, T. Beth, N. Cot and I. Ingemarsson, eds., Springer-Verlag Lecture Notes in Computer Science vol. 209, 1985, pp. 183–215, MR 87f:11105.
217. John D. Dixon, *Factorization and Primality Tests*, Amer. Math. Monthly **91** (1984), 333–352, MR 87c:11121a.
218. H. Dubner and R. Dubner, *The Development of a Powerful, Low-Cost Computer for Number Theory Applications*, J. Rec. Math. **18** (1986), 81–86.
219. J. L. Gerver, *Factoring Large Numbers with a Quadratic Sieve*, Math. Comp. **41** (1983), 287–294, MR 85c:11122.
220. S. Goldwasser and J. Kilian, *Almost All Primes Can Be Certified Quickly*, Proc. Eighteenth Annual ACM Symp. on the Theory of Computing (STOC), Berkeley, May 28-30, 1986, 316–329.
221. G. B. Gostin and P. B. McLaughlin, *Six New Factors of Fermat Numbers*, Math. Comp. **38** (1982), 645–649, MR 83c:10003.
222. G. McC. Haworth, *Primality Testing Mersenne Numbers* (II), Abstract 86T-11-57, Abstr. Amer. Math. Soc. **7** (1986), 224–225.
223. Guy Haworth, *Mersenne Numbers*, Reading, Berkshire, 1987 (privately published notes).
224. Wilfrid Keller, *Factors of Fermat Numbers and Large Primes of the Form* $k.2^n + 1$, Math. Comp. **41** (1983), 661–673, MR 85b:11117.
225. A. K. Lenstra and H. W. Lenstra, Jr., *Algorithms in Number Theory*, Technical Report 87-008, The University of Chicago, May, 1987.

226. H. W. Lenstra, Jr., *Primality Testing Algorithms (after Adleman, Rumely and Williams)*, *Séminaire Bourbaki*, Springer-Verlag Lecture Notes in Mathematics vol. 901, Berlin-New York, 1981, pp. 243–257, MR 83g:10002.
227. H. W. Lenstra, Jr., *Galois Theory and Primality Testing*, *Orders and Their Applications*, I. Reiner and K. Roggenkamp, eds., Springer-Verlag Lecture Notes in Mathematics, vol. 1142, Heidelberg, 1985, pp. 169–189, MR 87g:11171.
228. H. W. Lenstra, Jr., *Factoring Integers with Elliptic Curves*, Ann. of Math. **126** (1987), 649–673, MR 89g:11125.
229. Walter M. Lioen, Dik T. Winter and Herman J. J. te Riele, *Factoring with the Quadratic Sieve on Large Vector Computers*, J. Comput. Appl. Math. **27** (1989 pages 267–278), MR 90h:11111.
230. Peter L. Montgomery, *Modular Multiplication without Trial Division*, Math. Comp. **44** (1985), 519–521, MR 86e:11121.
231. Peter L. Montgomery, *Speeding the Pollard and Elliptic Curve Methods of Factorization*, Math. Comp. **48** (1987), 243–264, MR 88e:11130.
232. Thorkil Naur, *Integer Factorization*, Report DAIMI PB-144, Matematisk Institut, Aarhus Universitet, May, 1982.
233. Thorkil Naur, *New Integer Factorizations*, Math. Comp. **41** (1983), 687–695, MR 85c:11123.
234. A. M. Odlyzko, *Discrete Logarithms in Finite Fields and their Cryptographic Significance*, *Advances in Cryptology, Proceedings of EUROCRYPT 84*, T. Beth, N. Cot and I. Ingemarsson, eds., Springer-Verlag Lecture Notes in Computer Science vol. 209, 1985, pp. 224–314, MR 87g:11022.
235. D. Parkinson and M. Wunderlich, *A Compact Algorithm for Gaussian Elimination over* **GF**(2) *Implemented on Highly Parallel Computers*, Parallel Computing **1** (1984), 65–73.
236. Carl Pomerance, *Recent Developments in Primality Testing*, Math. Intelligencer **3** (1981), 97–105, MR 83h:10015.
237. Carl Pomerance, *Analysis and Comparison of Some Integer Factoring Algorithms*, *Computational Methods in Number Theory, Part 1*, H. W. Lenstra, Jr. and R. Tijdeman, eds., Math. Centrum Tract 154, Amsterdam, 1982, pp. 89–139, MR 84i:10005.
238. Carl Pomerance, *The Quadratic Sieve Factoring Algorithm*, *Advances in Cryptology, Proceedings of EUROCRYPT 84*, T. Beth, N. Cot and I. Ingemarsson, eds., Springer-Verlag Lecture Notes in Computer Science vol. 209, 1985, pp. 169–182, MR 87d:11098.
239. Carl Pomerance, J. W. Smith and Randy Tuler, *A Pipe-line Architecture for Factoring Large Integers with the Quadratic Sieve Algorithm*, SIAM J. Comput. **17** (1988), 387–403.
240. Carl Pomerance, J. W. Smith and S. S. Wagstaff, Jr., *New Ideas for Factoring Large Integers*, *Advances in Cryptology, Proceedings of Crypto 83*, David Chaum, ed., Plenum Press, New York, 1984, pp. 81–85, MR 86f:94001.
241. Carl Pomerance and S. S. Wagstaff, Jr., *Implementation of the Continued Fraction Integer Factoring Algorithm*, Congressus Numerantium **37** (1983), 99–118, MR 85c:11124.
242. Hans Riesel, *Prime Numbers and Computer Methods for Factorization*, Birkhäuser, Boston, 1985, MR 88k:11002.
243. Hans Riesel, *Modern Factorization Methods*, BIT **25** (1985), 205–222, MR 87c:11122.
244. W. G. Rudd, Duncan A. Buell and Donald M. Chiarulli, *A High Performance Factoring Machine*, *Proceedings of the Eleventh International Symposium on Computer Architecture*, 1984.
245. Robert Rumely, *Recent Advances in Primality Testing*, Notic. Amer. Math. Soc. **30** (1983), 475–477, MR 85b:11122.
246. C.-P. Schnorr and H. W. Lenstra, Jr., *A Monte Carlo Factoring Algorithm with Linear Storage*, Math. Comp. **43** (1984), 289–311, MR 85d:11106.
247. Robert D. Silverman, *The Multiple Polynomial Quadratic Sieve*, Math. Comp. **48** (1987), 329–339, MR 88c:11079.
248. Robert D. Silverman, *Parallel Implementation of the Quadratic Sieve*, The Journal of Supercomputing **1** (1988), 273–290.
249. J. W. Smith and S. S. Wagstaff, Jr., *An Extended Precision Operand Computer*, Proceedings of the Twenty-First Southeast Region ACM Conference (1983), 209–216.
250. Hiromi Suyama, *Searching for Prime Factors of Fermat Numbers with a Microcomputer*, bit (Tokyo) **13** (1981), 240–245 (in Japanese), MR 82c:10012.

251. Hiromi Suyama, *The Cofactor of F_{15} is Composite*, Abstr. Amer. Math. Soc. **5** (1984), 271–272.
252. Hiromi Suyama, *Large Primes and Prime Divisors of Fermat Numbers*, bit (Tokyo) **17** (1985), 136–143 (in Japanese).
253. S. S. Wagstaff, Jr. and J. W. Smith, *Methods of Factoring Large Integers*, *Number Theory, New York, 1984-85*, D. V. Chudnovsky, G. V. Chudnovsky, H. Cohn and M. B. Nathanson, eds., Springer-Verlag Lecture Notes in Mathematics, vol. 1240, Berlin, 1987, pp. 281–303.
254. Douglas H. Wiedemann, *Solving Sparse Linear Equations over Finite Fields*, IEEE Trans. Info. Theory **32** (1986), 54–61, MR 87g:11166.
255. H. C. Williams, *A $p + 1$ Method of Factoring*, Math. Comp. **39** (1982), 225–234, MR 83h:10016.
256. H. C. Williams, *An Overview of Factoring*, *Advances in Cryptology, Proceedings of Crypto 83*, David Chaum, ed., Plenum Press, New York, 1984, pp. 71–80, MR 86f:94001.
257. H. C. Williams and Harvey Dubner, *The Primality of R1031*, Math. Comp. **47** (1986), 703–711, MR 87k:11141.
258. H. C. Williams and C. D. Patterson, *A Report on the University of Manitoba Sieve Unit*, Congressus Numerantium **37** (1983), 85–98, MR 84g:10003.
259. H. C. Williams and M. C. Wunderlich, *On the Parallel Generation of the Residues for the Continued Fraction Factoring Algorithm*, Math. Comp. **48** (1987), 405–423, MR 88i:11099.
260. Marvin C. Wunderlich, *Factoring Numbers on the Massively Parallel Computer*, *Advances in Cryptology, Proceedings of Crypto 83*, David Chaum, ed., Plenum Press, New York, 1984, pp. 87–102, MR 86f:94001.
261. Marvin C. Wunderlich, *Implementing the Continued Fraction Factoring Algorithm on Parallel Machines*, Math. Comp. **44** (1985), 251–260, MR 86d:11104.
262. Samuel Yates, *Repunits and Repetends*, Delray Beach, FL, 1982, MR 83k:10014.
263. Jeff Young and Duncan A. Buell, *The Twentieth Fermat Number is Composite*, Math. Comp. **50** (1988), 261–263, MR 89b:11012.