2	3	5	7	11	13	17	19	23	29
31	37	41	43	47	53	59	61	67	71
73	79	83	89	97	101	103	107	109	113
127	131	137	139	149	151	157	163	167	173
179	181	191	193	197	199	211	223	227	229
233	239	241	251	257	263	269	271	277	281
283	293	307	311	313	317	331	337	347	349
353	359	367	373	379	383	389	397	401	409
419	421	431	433	439	443	449	457	461	463
467	479	487	491	499	503	509	521	523	541
547	557	563	569	571	577	587	593	599	601
607	613	617	619	631	641	643	647	653	659
661	673	677	683	691	701	709	719	727	733
739	743	751	757	761	769	773	787	797	809
811	821	823	827	829	839	853	857	859	863
877	881	883	887	907	911	919	929	937	941
947	953	967	971	977	983	991	997	1009	1013
1019	1021	1031	1033	1039	1049	1051	1061	1063	1069
1087	1091	1093	1097	1103	1109	1117	1123	1129	1151
1153	1163	1171	1181	1187	1193	1201	1213	1217	1223
1229	1231	1237	1249	1259	1277	1279	1283	1289	1291
1297	1301	1303	1307	1319	1321	1327	1361	1367	1373
1381	1399	1409	1423	1427	1429	1433	1439	1447	1451
1453	1459	1471	1481	1483	1487	1489	1493	1499	1511
1523	1531	1543	1549	1553	1559	1567	1571	1579	1583
1597	1601	1607	1609	1613	1619	1621	1627	1637	1657
1663	1667	1669	1693	1697	1699	1709	1721	1723	1733
1741	1747	1753	1759	1777	1783	1787	1789	1801	1811
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

Primzahl-Rekordjagd

von Günter M. Ziegler

Der Dezember 2003 beschert uns mehrere Primzahl-Rekorde. So wurde unter der Regie von Jens Franke (Bonn) das RSA-576 *Entschlüsselungsproblem gelöst: die Faktorisierung einer 174-stelligen Dezimalzahl.*
Die größte bekannte Primzahl ist ebenfalls neu, eine Mersenne'sche Primzahl mit insgesamt 6.320430 Stellen: $M = 2^{20.996.011} - 1$. *Die Medien (unter anderem Spiegel-Online vom 3. Dezember) schreiben die Entdeckung einem Studenten der Verfahrenstechnik an der Michigan State University namens Michael Shafer zu – aber das ist nur ein Teil der Wahrheit.*

Mersenne'sche Zahlen

Seit Januar 1996 läuft im Internet eine Suche nach immer größeren Mersenne'schen Primzahlen. In dem verteilten Rechenprojekt unter dem Titel GIMPS ("Great Internet Mersenne Prime Search", `www.mersennse.org`), können Freiwillige über's Internet die GIMPS-Computerprogramme abrufen und „ihre" Zahlen zum Testen zugeteilt bekommen, ihre PCs damit Sklavenarbeit leisten lassen, und die Rückmeldung über's Internet abliefern.

Zur Erinnerung: zu Ehren des französischen Mönches Marin Mersenne (1588–1648) heißen die Zahlen der Form $M_n = 2^n - 1$ *Mersenne'sche Primzahlen* – wenn sie prim sind. Dafür ist notwendig (schöne Übungsaufgabe aus der elementaren Zahlentheorie), dass n selbst prim ist. Aber hinreichend ist das nicht: $n = 11$ liefert das erste Gegenbeispiel. Im Jahr 1644 behauptete Mersenne, dass M_n für $n = 2$, 3, 5, 7, 13, 17, 19, 31, 67, 127 und 257 prim sei, aber für keine andere Primzahl unter 257 (womit er exakt fünfmal danebengelegen hat). Mersenne'sche Primzahlen sind ziemlich selten: man weiß nicht, ob es unendlich viele gibt, und man kennt inzwischen die ersten 38 davon, und nur zwei weitere, darunter die neu gefundene $M_{20.996.011}$, die auch die größte bekannte Primzahl überhaupt ist.

Dass man Zahlen mit mehr als 6 Millionen Stellen effektiv auf Primalität testen kann, ist die eigentliche wissenschaftliche (und programmiererische) Höchstleistung hinter dem neuen Rekord – dass $n = 20.996.011$ prim sein muss, ist ja nur eine klitzekleine Aufwärmübung für den neuen Rekord.

Primalitätstests

Nun weiß man seit Kurzem, dass es exakte Primzahltests gibt, die in Polynomzeit laufen – siehe *DMV Mitteilungen* 4–2002, S. 14–21. Diese stellen einen theoretischen Durchbruch dar, sind aber für den Einsatz in der Praxis (noch) nicht geeignet. Im GIMPS-Projekt wird für jedes prime n eine Kaskade von klassischeren Tests durchlaufen, die unter `www.mersenne.org/math.htm` sehr schön und kapierbar beschrieben werden:[1]

In *Phase I* sucht man nach kleinen Primteilern q von $2^n - 1$. Diese müssen (wieder eine hübsche Übungsaufgabe) $q \equiv 1 \bmod 2n$ und $q \equiv \pm 1 \bmod 8$ erfüllen. Mithilfe eines auf solche Faktoren zugeschnittenen „Sieb des Eratosthenes" werden dann Primteiler von M_n bis ca. 40.000 erkannt. Dabei kann ausgenutzt werden, dass Teilbarkeitstests für Zahlen vom Typ $2^n - 1$

1 Zur algorithmischen Primzahltheorie empfehlen die Experten Richard Crandell & Carl Pomerance: "Prime Numbers. A Computational Perspective", Springer-Verlag, New York 2001.
Aus Computeralgebra-Perspektive finden sich Primzahltests (und sehr viel mehr Spannendes) in Joachim von zur Gathen & Jürgen Gerhard: "Modern Computer Algebra", Cambridge University Press, 2. Auflage 2003.

Marin Mersenne, 1588–1648 (Quelle: `http://www-groups.dcs.st-and.ac.uk/~history/PictDisplay/Mersenne.html`)

in Binärarithmetik sehr effektiv durchgeführt werden können.

In *Phase II* wird dann ein Spezialfall der sogenannten $(p-1)$-Methode von Pollard (1974) verwendet, mit der man Faktoren $q = 2kn + 1$ finden kann, für die $q - 1 = 2kn$ aus vielen kleinen Primfaktoren besteht, oder aber (in einer verbesserten Version) bis auf einen etwas größeren Primfaktor stark zusammengesetzt ist: Wenn man q sucht, so dass alle Primfaktoren kleiner als B sind, so bildet man dafür das Produkt $E := \prod_{p<B} p$ aller Primzahlen, die kleiner als B sind, und berechnet dann $x := 3^{E2n}$. Im ggT von $x - 1$ und $2^n - 1$ fängt man dann den gesuchten Teiler von $2^n - 1$.

Erst in *Phase III* verwendet man dann ein Verfahren, mit dem man sicher entscheiden kann, ob $2^n - 1$ prim ist, den sogenannten Lucas–Lehmer Test (1878, 1930/1935) für Mersenne-Zahlen: M_n ist genau dann prim, wenn $\ell_{n-1} \equiv 0 \bmod M_n$ gilt, wobei die ℓ_k durch $\ell_1 = 4$ und $\ell_n = \ell_{n-1}^2 - 2$ rekursiv definiert werden. Um das effektiv zu berechnen, muss man riesige Zahlen schnell modulo $2^n - 1$ quadrieren. Dazu werden die Zahlen in große Blöcke unterteilt, und dann arbeitet man mit Spezialversionen einer schnellen Fourier Transformation ("Fast Fourier Transform", FFT), in diesem Fall mit einer FFT bezüglich einer irrationalen Basis, die von Richard Crandell und Barry Fagin (*Mathematics of Computation* 1994) eingeführt wurde. Auf den WWW-Seiten des *Mathematica*-Projekts `mathworld.wolfram.com`, die die aktuelle Rekordmeldung verbreiten, wird suggeriert, GIMPS würde mit einer *Mathematica*-Implementierung arbeiten, aber das ist eine arge Dehnung der Tatsachen. (Es hat nur Crandell die Methode auch mal für die Primzahltests von *Mathematica* implementiert.) In der Tat arbeitet GIMPS mit hochoptimiertem Assembler-Code, aus Prozessorarchitekturgründen in Gleitkommaarithmetik, deren Fehler getrennt erkannt und aufgefangen werden müssen.

Primalität und Faktorisierung

Phasen I und II des GIMPS-Verfahrens spucken also im Fall von zusammengesetztem M_n wirklich Teiler aus – *wenn* sie welche finden –, die dritte und entscheidende Phase aber nicht mehr. Die Antwort heißt da dann nur noch „zusammengesetzt!", ohne einen expliziten (Prim-)Teiler als Beweis. Es wird also ein Primalitätstest durchgeführt, aber kein vollständiges Faktorisierungsverfahren.

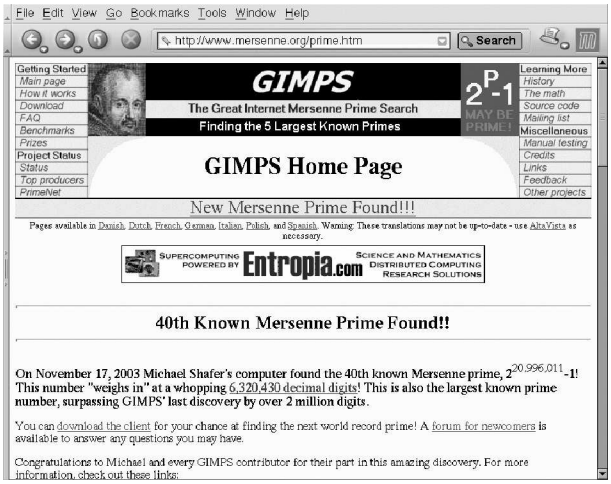
Und das ist auch gut so: Nicht einmal für den Spezialfall von Mersenne-Zahlen kennt man effektive Verfahren zum Faktorisieren. Ein Verfahren, mit dem man *beliebige* Zahlen mit ein paar Hundert Stellen faktorisieren könnte wäre interessant und bedrohlich, weil die kryptographischen Verfahren, die die Sicherheit von Online-Banking und Internet garantieren sollen, darauf beruhen, dass das Faktorisieren und verwandte Probleme (wie die Berechnung von „diskreten Logarithmen") offenbar schwer sind.

RSA

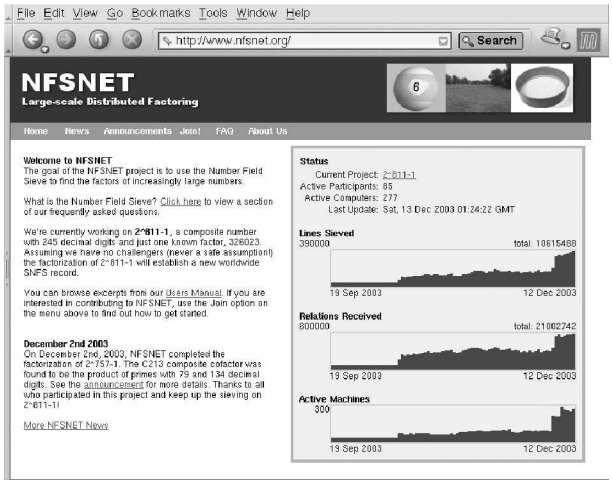
Ein Beispiel dafür ist das von Ron Rivest, Adi Shamir und Leonard Adleman 1978 publizierte Verschlüsselungsverfahren „mit öffentlichen Schlüsseln", das sich inzwischen in fast jedem elementaren Zahlentheorie-Lehrbuch findet, gleichzeitig aber auch in der Praxis vielfältig zum Einsatz kommt – siehe die Homepage `http://www.rsasecurity.com` der Firma von Rivest, Shamir und Adleman. Die Sicherheit des Verfahrens gegen unerlaubtes Entschlüsseln hängt davon ab, dass es mit heutiger Technologie sehr schwer ist, Produkte von Zahlen mit 150-200 Stellen in ihre Primfaktoren zu zerlegen. Die Firma "RSA Securities" hat sogar Preise auf Beispielprobleme[2] ausgesetzt. Der erste davon ist/war ein Preis von 10.000 Dollar für das Faktorisieren der Zahl „RSA-576"

188198812920607963838697239461650439807163563
379417382700763356422988859715234665485319060
606504743045317388011303396716199692321205734
031879550656996221305168759307650257059

2 `http://www.rsasecurity.com/rsalabs/challenges/factoring/numbers.html`



http://www.mersenne.org/prime.htm



http://www.nfsnet.org/

mit 174 Dezimalziffern, bzw. 576 Binärziffern (bits). Und dieses Problem hat Jens Franke von der Universität Bonn jetzt geknackt, wie *Heise Online* am 8. Dezember gemeldet hat: die Zahl hat Faktoren

398075086424064937397125500550386491199064362342526708406385189575946388957261768583317

und

472772146107435302536223071973048224632914695302097116459852171130520711256363590397527

(mit je 87 Ziffern), und die sind prim – was wiederum mit den aktuellen Methoden ganz leicht zu zeigen ist. Franke verwendete dabei das „General Number Field Sieve (GNFS)“. Dieses wurde von Lenstra, Lenstra, Manasse & Pollard 1990 eingeführt, und hat eine Laufzeit von $\exp(O(\sqrt[3]{n \log n}))$ für n-stellige Zahlen; es ist also nicht ganz polynomial, aber *fast*. Unter Verwendung des GNFS wurden auch schon die kleineren Testprobleme von RSA-100 bis RSA-512 geknackt (letzteres im August 1999).

Und es gibt noch mehr aktuelle Rekorde, die sich ebenfalls auf's Faktorisieren beziehen: Unter anderem versucht man eben Mersenne-Zahlen nicht nur auf Primalität zu untersuchen, sondern auch vollständig in Primfaktoren zu zerlegen. NFSNET (`http://www.nfsnet.org` ist auch ein Internet-Projekt, dem es nun (Erfolgsmeldung vom 2. Dezember) in Internet-Gemeinschaftsarbeit gelang, die Mersenne'sche Zahl $2^{787} - 1$ vollständig zu faktorisieren: Die Primfaktoren 9815263 und 561595591 dieser Zahl kannte man schon länger, aber der 212-stellige Rest war ein hartes Stück Arbeit: er wurde jetzt in die Primfaktoren

572213702200206782424822797509585774915131282780938840696234625318212891696459 3

und

240338216409835080887362734030059654466890023563443321305650666431938139011197710904242694120545430727149147426656777742473252923275 59

zerlegt. Dieser Erfolg basiert auf dem “Special Number Field Sieve (SNFS)” – einer schnelleren Spezialversion des GNFS, die nur für spezielle Zahlen, etwa vom Typ $b^n \pm 1$, anwendbar ist.

Rekordjagd

Die Rekordjagd geht weiter. Die „Electronic Frontier Foundation“ (`http://www.eff.org/`) hat schon im Jahr 2000 einmal 50.000 Dollar für die erste Primzahl mit einer Million Stellen ausgezahlt. Für die Identifikation einer Primzahl mit mehr als 10 Millionen Dezimalstellen hat sie 100.000 Dollar ausgesetzt. Dies heizt die Stimmung an, und das GIMPS-Projekt sucht Mitstreiter, die ihre Computer für die Rekordjagd einsetzen wollen.

Genauso ist man natürlich hinter den größeren RSA-Problemen hinterher; als nächstes wartet da RSA-640, eine Zahl mit 193 Dezimalstellen, auf ihre Zerlegung. Darauf sind 20.000 Dollar ausgesetzt.

Und die nächste Mersenne-Zahl auf der Abschuss- bzw. Zerlegungsliste von NFSNET ist $2^{811} - 1$. Auch für dieses Projekt werden noch Mitstreiter gesucht.

Viele arme kleine PCs werden also mit Zahlen gefüttert und mit Primzahltests und mit Zerlegungsverfahren gequält werden, nur damit Herrchen vielleicht einen Teil des Ruhms (und des Preisgeldes) einkassieren kann.

Adresse des Autors
Prof. Günter M. Ziegler
Institut für Mathematik
MA 6-2
Technische Universität Berlin
Straße des 17. Juni 136
10623 Berlin
`ziegler@math.tu-berlin.de`