

Schüler-Propädeutikum Mathematik 2020

Vortrag 5: Primzahltests

Andreas Lochmann

4. Dezember 2023

1 Knobelaufgabe von letzter Woche

Primzahlsatz:

Mit $\pi(x)$ bezeichnen wir die Zahl der Primzahlen zwischen 1 und (einschließlich) x . Dann ist $\pi(x)$ ungefähr $x/\ln(x)$, in folgendem Sinn:

$$\lim_{x \rightarrow \infty} \frac{\pi(x)}{\frac{x}{\ln(x)}} = 1.$$

Dabei steht \ln für den natürlichen Logarithmus.

Aufgabe:

Wie viele Primzahlen mit bis zu 100 Stellen gibt es ungefähr?

Wie viele SD-Karten bräuchte man zu ihrer Speicherung etwa?

(SD-Karten haben eine Kapazität von etwa 160 Terabyte pro Kilogramm.)

Lösung:

Wir nähern $\pi(10^{100})$:

$$\pi(10^{100}) \approx \frac{10^{100}}{\ln(10^{100})} = \frac{10^{100}}{100 \cdot \ln 10} \approx \frac{10^{98}}{2,3} \approx 4,3 \cdot 10^{97}$$

Ein Byte kann 256 verschiedene Werte annehmen – wir können also näherungsweise 2 Dezimalstellen mit einem Byte beschreiben und brauchen etwa $2,1 \cdot 10^{99}$ Bytes, bzw. $2 \cdot 10^{87}$ Terabyte. Bei 160 TB/kg entspricht dies etwa 10^{85} Kilogramm.

Zum Vergleich: Das beobachtbare Universum hat nur ungefähr 10^{80} Teilchen, also deutlich weniger als 100-stellige Primzahlen!

Eine Liste aller 100-stelligen Primzahlen ist also aus praktischer Sicht unmöglich.

2 Ein klassischer Primzahltest: Der Lucas-Test

Zur Erinnerung: Kleiner Satz von Fermat:

Wenn p eine Primzahl und $a \in \mathbb{N}$ ist, so ist $a^p \equiv a \pmod{p}$. Wenn p darüber hinaus kein Teiler von a ist, gilt sogar $a^{p-1} \equiv 1 \pmod{p}$.

Lucas-Test (1846): Sei $n \in \mathbb{N}$, $n > 1$. Angenommen, man findet eine Zahl $a \in \mathbb{N}$, $a > 1$, mit:

1. $a^{n-1} \equiv 1 \pmod{n}$
2. $a^m \not\equiv 1 \pmod{n}$ für alle $m = 1, 2, \dots, n-1$.

Dann ist n prim.

Leider benötigt dieser Test in *dieser* Version zu lange, und ist daher für moderne Zwecke ungeeignet. Es gibt aber Verbesserungen:

Verbesserung (1891): Es reichen solche m mit $m \mid n-1$.

Verbesserung (1953): Es reichen solche m mit $\frac{n-1}{m}$ prim.

Verbesserung (1967): Für jedes der m darf man ein eigenes a wählen.

Knobelaufgabe: Versuchen Sie mit einem dieser Tests zu beweisen, dass 31 prim ist.

Mit Tests wie dem Lucas-Test und seinen Varianten werden auch heute noch Rekorde bei der Suche nach sehr großen Primzahlen aufgestellt – für die Zwecke von RSA sind sie aber leider ungeeignet, da auch der Lucas-Test und seine Varianten zu lange brauchen, um in kürzester Zeit Tausende von Primzahlen zu finden.

3 Probabilistische Primzahltests

Fermatscher Primzahltest: Wenn n nicht prim ist, und a zu n teilerfremd ist, kann es sein, dass $a^{n-1} \not\equiv 1 \pmod{n}$ ist. Wenn wir ein passendes a finden, so dass $a^{n-1} \pmod{n} \neq 1$ ist, muss also n zusammengesetzt sein – wir nennen a dann einen *Zeugen* für die Zusammengesetztheit von n . Aber es kann auch sein, dass $a^{n-1} \pmod{n} = 1$ ist, obwohl n zusammengesetzt ist – in dem Fall nennen wir a einen *Lügner* (in Bezug auf den Primzahltest für n).

Die Idee ist: „Befrage“ sehr viele verschiedene a nach der Zusammengesetztheit von n . Wenn immer $a^{n-1} \pmod{n} = 1$ ist, ist n wahrscheinlich prim; ist auch nur einmal das Ergebnis nicht 1, so muss n zusammengesetzt sein.

Leider gibt es zusammengesetzte Zahlen n , für die *alle* teilerfremden Zahlen a Lügner sind – solche Zahlen heißen **Carmichael-Zahlen**. Die kleinste Carmichael-Zahl ist 561.

Frage: 527 und 589 sind beide zusammengesetzt. Welche Zahlen a sind Lügner für 527? Welche für 589? Gibt es auch Zeugen?

Miller-Rabin-Test (1976): Sei n ungerade, $a \in \{2, 3, \dots, n-2\}$ zu n teilerfremd. Seien $d, j \in \mathbb{N}$ derart, dass $n-1 = 2^j \cdot d$ und $2 \nmid d$ ist. (j ist einfach die Zahl der Nullen am Ende der Binärdarstellung von $n-1$; j und d sind also leicht zu berechnen.) Dann prüfe:

Ist $a^d \equiv 1 \pmod{n}$ oder $a^{d \cdot 2^r} \equiv -1 \pmod{n}$ für ein $r \in \{0, 1, \dots, j-1\}$?

Für Primzahlen n ist das immer richtig. Ist n zusammengesetzt, so nennen wir wiederum a einen Lügner, wenn die Aussage stimmt, und einen Zeugen, wenn die Aussage falsch ist.

Es gibt Lügner a für den Miller-Rabin-Test, aber man kann beweisen, dass es immer auch einen Zeugen für die Zusammengesetztheit gibt ... besser noch: Weniger als 25 % aller zu n teilerfremden Zahlen zwischen 1 und n sind Lügner!

Frage: Wie hoch ist im Miller-Rabin-Test die Wahrscheinlichkeit, dass von zehn verschiedenen a s alle zehn Lügner sind?

Wenn man den Miller-Rabin-Test nur genügend oft (also mit genügend vielen a s) ausführt, kann man mit praktischer Sicherheit davon ausgehen, dass eine Primzahl vorliegt, weil die Wahrscheinlichkeit für N Lügner kleiner als 4^{-N} ist und daher exponentiell fällt. Es handelt sich aber um keine 100 % -igen Beweise – es gibt auch beim Miller-Rabin-Test immer eine sehr, sehr kleine Wahrscheinlichkeit, dass man nur auf Lügner getroffen ist. Diese Wahrscheinlichkeit ist halt nur aus praktischer Sicht verschwindend gering.

4 Zwei Faktorisierungsmethoden

Fermat (1643): Sei $n = p \cdot q$ mit unbekanntem $p, q \in \mathbb{N}$. Wir nehmen $p < q$ und $2 \nmid n$ an (dann sind natürlich auch p und q ungerade). Wir setzen

$$r = \frac{q-p}{2} \quad \text{und} \quad s = \frac{q+p}{2}.$$

Dann sind auch r und s natürliche Zahlen, und p und q können aus r und s berechnet werden.

Frage: Wie können p und q mittels r und s berechnet werden?

Dann ist $n = pq = (s - r)(s + r) = s^2 - r^2$. Angenommen, p und q sind etwa gleich groß. Dann wird r relativ klein sein, und s ist ungefähr die Wurzel aus n . Dadurch erhalten wir eine erste Schätzung für s :

Erste Schätzung: s_1 sei \sqrt{n} aufgerundet

Ist $n - s_1^2$ eine Quadratzahl?

Wenn ja: Dann ist $r = \sqrt{n - s_1^2}$: fertig.

Wenn nein: Setze $s_2 = s_1 + 1$ und wiederhole alles.

Ist auch $n - s_2^2$ keine Quadratzahl fährt man mit $s_3 = s_2 + 1$ fort usw.

Beispiel: Wir wollen $n = 527$ faktorisieren. Es ist $\sqrt{n} \approx 22,956\dots$. Wir setzen also $s_1 = 23$. Dann ist $r_1^2 = s_1^2 - n = 23^2 - 527 = 2$: Das ist keine Quadratzahl. Also versuchen wir $s_2 = 24$. Dann ist $r_2^2 = s_2^2 - n = 24^2 - 527 = 49$. Dies ist eine Quadratzahl, also ist $r_2 = 7$, $s_2 = 24$ und damit $p = s_2 - r_2 = 17$ und $q = s_2 + r_2 = 31$; also $527 = 17 \cdot 31$.

Knobelaufgabe: Faktorisieren Sie 345.383 mit Fermats Methode.

Fermats Methode funktioniert am besten, wenn p und q nahe beieinander liegen, denn dann ist r klein und Fermats Methode findet schnell eine Lösung. Daher fordert man im RSA-Verfahren, dass sich p und q um mehrere Größenordnungen voneinander unterscheiden sollen.

Pollards $p - 1$ -Methode (1974): Sei p prim und $a \in \mathbb{N}$ mit $p \nmid a$. Dann ist $a^{p-1} \equiv 1 \pmod{p}$. Wir kennen p nicht. Angenommen, $p - 1$ enthält viele kleine Primfaktoren (die wir natürlich auch allesamt nicht kennen). Man könnte dann viele kleine Zahlen miteinander multiplizieren und so eine (ggbf. sehr große) Zahl M erhalten mit $p - 1 \mid M$. Dann ist also $M = x \cdot (p - 1)$ für ein $x \in \mathbb{N}$ und daher

$$a^M \equiv (a^x)^{p-1} \equiv 1 \pmod{p},$$

also $p \mid a^M - 1$. Ist $p \mid n$, so ist der größte gemeinsame Teiler von $a^M - 1$ und n ebenfalls durch p teilbar. Der ggT lässt sich mit dem euklidischen Algorithmus schnell berechnen, und mit etwas Glück ist der ggT gleich p (statt gleich n).

Beispiel: Wir versuchen erneut, $n = 527$ zu faktorisieren. Wir wählen auf's Geratewohl

$$M = 2 \cdot 2 \cdot 3 \cdot 5 \cdot 7 = 420$$

und $a = 2$. Dann ist $a^M - 1$ die gewaltig große Zahl

2707685248164858261307045101702230179137145581421695874189921465
443966120903931272499975005961073806735733604454495675614232575

(mit 127 Stellen). Mit Computerunterstützung berechnet man schnell den ggT von $a^M - 1$ und 527; das Ergebnis ist 31. Dadurch haben wir einen Faktor von n gefunden. Ob dies funktioniert, hängt aber von einer günstigen Wahl von M und a ab: Wählt man beispielsweise $a = 13$ und $M = 420$, so ist $\text{ggT}(a^M - 1, 527) = 527$, man hätte dann also keinen echten Teiler von 527 gefunden, sondern 527 selbst.

Frage: Wenn man die Rechnung aus dem Beispiel mit dem gleichen M aber verschiedenen a ausprobiert, erhält man zwar oft den Faktor 31, aber niemals den Faktor 17 – warum?

Pollards $p - 1$ -Methode funktioniert am besten, wenn $p - 1$ aus vielen kleinen Primfaktoren zusammengesetzt ist, da man nur dann ein geeignetes M finden kann. Deshalb interessiert man sich im Rahmen von RSA für Primzahlen p , so dass $p - 1$ mindestens einen sehr sehr großen Primfaktor besitzt. Der Extremfall hierfür sind sichere Primzahlen p , wenn also $(p - 1)/2$ prim ist.

5 Ein weiterer klassischer Primzahltest: Der Satz von Wilson

Satz von Wilson:

Sei $n \in \mathbb{N}$. Dann ist n genau dann prim, wenn $(n-1)! \equiv -1 \pmod{n}$ ist. (Alternativ kann man auch $(n-1)! \equiv n-1$ schreiben, denn $-1 \equiv n-1$ modulo n .)

Beweis:

Angenommen, n ist keine Primzahl. Dann werden die Primteiler von n allesamt in den Zahlen 1 bis $n - 1$ auftauchen, das Produkt $(n - 1)!$ enthält also n als Teiler und damit ist es Null modulo n , und nicht -1 .

Nehmen wir jetzt an, n ist eine Primzahl. Modulo n besitzt dann jede der Zahlen $x \in \{1, \dots, n - 1\}$ ein (modulares) Inverses, das ebenfalls in diesem Intervall liegt. Im Produkt $(n - 1)!$ heben sich also (fast) alle Zahlen mit ihrem Inversen gegenseitig auf. Die einzige Ausnahme sind solche Zahlen x , die modulo n ihr eigenes Inverses sind, also Zahlen x mit $x \cdot x \equiv 1 \pmod{n}$. Da diese ihre eigenen Partner sind, tauchen Sie im Produkt $(n - 1)!$ nur einmal auf. Aber aus $x^2 \equiv 1$ folgt $x^2 - 1 \equiv 0$, also $n \mid (x - 1)(x + 1)$. Da n eine Primzahl ist, ist dies nur möglich, wenn $n \mid x - 1$

oder $n \mid x + 1$ ist, unter den Zahlen $n \in \{1, \dots, n - 1\}$ trifft dies nur auf $x = 1$ und $x = n - 1$ zu. Also ist

$$(n - 1)! \equiv 1 \cdot (n - 1) \equiv -1 \pmod{n}.$$

□

Der Satz von Wilson ist im Prinzip ein geeigneter Primzahltest: Man multipliziere alle Zahlen von 1 bis $n - 1$ miteinander, rechne modulo n und prüfe, ob das Ergebnis -1 ist. Wenn ja, handelt es sich in jedem Fall um eine Primzahl, wenn nein, ist es definitiv keine Primzahl.

Aus praktischer Sicht ist der Satz von Wilson nach aktuellem Kenntnisstand aber als Primzahltest für große Primzahlen ungeeignet, da wir kein Verfahren kennen, um $(n - 1)! \pmod{n}$ schnell zu berechnen. Dies mag sich ändern, falls wir in der Zukunft einen passenden Algorithmus finden – es ist aber unklar, ob es einen solchen Algorithmus überhaupt gibt.

Literatur

Zum Thema Kryptographie:

Kryptographie von Dietmar Wätjen

Zum Thema große Primzahlen (abseits von kryptographischen Anwendungen):

Die Welt der Primzahlen von Paulo Ribenboim