3.3 Der Primzahltest von Miller

Wie ist das im vorigen Abschnitt entwickelte Kriterium, der strenge Pseudoprimzahltest zu einer oder genügend vielen Basen, praktisch verwertbar? Dazu ist zunächst der Algorithmus für eine Basis a zu formulieren und sein Aufwand zu bestimmen.

Da a^{n-1} sowieso nach dem binären Potenzalgorithmus berechnet wird, ist es effizienter, gleich die ganze Folge der Potenzen ab a^r zu bestimmen; dann ist der Aufwand nicht wesentlich größer als für den „schwachen“ Pseudoprimzahltest. Der strenge Pseudoprimzahltest zur Basis a sieht dann so aus:

Prozedur sPPT(a)
[Strenger Pseudoprimzahltest zu einer Basis a.]
Eingabeparameter:
n = die zu prüfende Zahl (ungerade ≥ 3),
s = Zweierordnung von $n-1$ (vorberechnet),
r = ungerader Teil von $n-1$ (vorberechnet),
a = Basis (im Bereich $[2 \ldots n-1]$).
Ausgabeparameter:
zus = ein Boolescher Wert mit der Bedeutung
TRUE: n ist sicher zusammengesetzt,
FALSE: die Prüfung gab kein definitives Ergebnis
[d. h., n ist strenge Pseudoprimzahl zur Basis a].
Anweisungen:
Bestimme $b = a^r \bmod n$.
Setze $k = 0$.
[Schleife: Am Eingang ist $b = a^{2^k r} \bmod n$;
die Boolesche Variable ‘Ende’, vorbesetzt mit FALSE, entscheidet über das nochmalige Durchlaufen der Schleife.]
Solange nicht Ende:
Falls $b = 1$: Setze Ende = TRUE;
falls $k = 0$, setze zus = FALSE,
sonst setze zus = TRUE. [1 ohne vorherige -1]
Falls $b = n-1$ und $k < s$:
Setze zus = FALSE, Ende = TRUE.
Falls $k = s$ und $b \neq 1$:
Setze zus = TRUE, Ende = TRUE.
In allen anderen Fällen $[k < s, b \neq 1, b \neq n-1]$
ersetze b durch $b^2 \bmod n$,
ersetze k durch $k+1$.

Der Aufwand läßt sich in Einzelschritte aufbrechen, in denen jeweils zwei Zahlen $\bmod n$ multipliziert werden. Zur Berechnung von $a^r \bmod n$ sind

höchstens $2 \cdot {}^2\log(r)$ solcher Schritte nötig. Bei den höchstens s Schleifendurchläufen wird noch je einmal quadriert. Da ${}^2\log(n-1) = s + {}^2\log(r)$, sind also insgesamt höchstens $2 \cdot {}^2\log(n)$ Quadrate $\bmod n$ zu bilden. Jedes solche Quadrat erfordert höchstens N^2 „primitive“ Ganzzahl-Multiplikationen, wobei N die Stellenzahl von n in der verwendeten Basis des Zahlsystems ist. Die Bestimmung von r bedeutet s Divisionen durch 2 und kann hier vernachlässigt werden. Der Gesamtaufwand ist also, grob geschätzt, $\mathrm{O}(\log(n)^3)$.

Der Primzahltest von MILLER ist nun einfach die Aneinanderreihung der strengen Pseudoprimzahltests zu den Basen $2, 3, 4, 5, \ldots$. Das sieht zunächst nicht effizient aus, denn wenn man tatsächlich eine Primzahl testet, muss man scheinbar alle Basen $< n$ durchlaufen. MILLER hat aber gezeigt, dass man mit entscheidend weniger auskommt – *vorausgesetzt, die erweiterte* RIEMANN*sche Vermutung ist wahr.* Hierzu im nächsten Abschnitt einige Erläuterungen ohne vollständige Beweise.