Recall the "textbook" RSA public key cryptosystem: Alice chooses large primes $p, q > 10^{150}$ and computes $N = pq$. Let $\varphi(N) = (p-1)(q-1) = N - (p+q) + 1$. Alice chooses $e \in \mathbb{N}$ coprime to $\varphi(N)$. Let $d \in \mathbb{N}$ be such that $ed \equiv 1 \mod \varphi(N)$. The **public key** for Alice is (N, e) and her **private key** is d.
To encrypt $m \in \mathbb{Z}/N\mathbb{Z}$ to Alice we compute $c = m^e \mod N$. Alice decrypts c to get m by computing $c^d \mod N$.

Computational efficiency: One computes $m^e \mod N$ and $c^d \mod N$ using the square-and-multiply method, but it still takes some computation time.

Temptations:

- Choose e small (e.g., 65537) to speed up encryption.
- Choose d small to speed up decryption.

Are there security weaknesses if one does this?

Theorem 22.1. Suppose $p < q < 2p$, $N = pq$, $1 < e < \varphi(N)$ and $1 < d < N^{1/4}/\sqrt{6}$. Then one can compute the private key d using continued fractions.

Exercise 22.2. Show how to speed up RSA decryption using the Chinese remainder theorem. How much of a speedup does this give?

Continued fraction expansion of $\sqrt{d}$.
Let $d \in \mathbb{N}$ be square-free.

Lemma 22.3. Let $m_0 = 0, q_0 = 1$. For $i \geq 0$ set $a_i = \lfloor (m_i + \sqrt{d})/q_i \rfloor$, $m_{i+1} = a_i q_i - m_i$, $q_{i+1} = (d - m_{i+1}^2)/q_i$. Then $q_i \in \mathbb{Z}$ and $[a_0; a_1, \dots]$ is the continued fraction expansion of $\sqrt{d}$.

Exercise 22.4. With notation as in Lemma 22.3, show that $1 \leq q_i \leq d$ and $|m_i| < \sqrt{d}$ for all $i \geq 0$.

Corollary 22.5. The continued fraction expansion of $\sqrt{d}$ is periodic.

Theorem 22.6. The continued fraction expansion of $\sqrt{d}$ has the form

$$[a_0; \overline{a_1, \dots, a_r, 2a_0}]$$

for some $r \in \mathbb{N}$.