Suggested Reading

Andreas Klappenecker

We have discussed the Euclidean algorithm in its subtractive form and in its more modern form. We also derived the extended Euclidean algorithm. You should read Knuth, *The Art of Computer Programming*, Section 1.1, for some historical context. You should also read the book that started it all: Euclid, *Elements*, Book VII; see, for instance,

`http://aleph0.clarku.edu/ djoyce/java/elements/bookVII/bookVII.html`

The worst case performance of the Euclidean algorithm is attained if two subsequent Fibonacci numbers are provided as input. You find more interesting information about Fibonacci numbers in Knuth, *Art of Computer Programming*, Section 1.2.8.

A much more detailed analysis of the performance of the Euclidean algorithm is contained in Knuth, *The Art of Computer Programming*, Section 4.5.3. Knuth first discusses the relation to continued fractions. He shows the slightly more precise result that the number of steps required by the division with remainder version of the Euclidean algorithm takes $\lfloor \log_\phi (3-\phi)N \rfloor$ steps if the input (a, b) is bounded by $0 \leq b < N$. The worst case analysis of often of little value, Knuth's book also contains a thorough discussion of an average case analysis of Euclid's algorithm.

A more detailed discussion of RSA is contained in any book on cryptography; see, or instance, Buchmann, *Introduction to Cryptography*, Springer, 2001. This book contains also more details about the Chinese Remainder Theorem. Buchmann also gives an analysis of the performance of the quadratic sieve factoring algorithm.

The chapter on number-theoretic algorithms in our textbook [CLRS] is mandatory reading material. You should also review the material of Part I of this book to refresh your memory about the mathematical foundations (you should known this material from Math 302 and CPSC 311, but some of the questions indicate that you might have to review that material).