1. Recall the protocol by which Alice commits herself to a bit \( x \in \{0, 1\} \) without revealing \( x \) to Bob. Namely, Alice first chooses two large random prime numbers \( P \) and \( Q \), one of which ends in a ‘7’ if and only if \( x = 1 \). She then computes their product \( N = PQ \) and sends \( N \) to Bob, but keeps the factors \( P \) and \( Q \) to herself. To reveal the value of \( x \) later, Alice sends \( P \) and \( Q \) to Bob, whereupon Bob checks that (i) \( P \) and \( Q \) encode the claimed value of \( x \), (ii) \( P \) and \( Q \) are indeed prime numbers, and (iii) \( PQ = N \). Suppose Bob forgets to check that \( P \) and \( Q \) are prime. Does the protocol still work correctly, and if not, what can go wrong?

2. Recall Euclid’s algorithm for computing \( \text{GCD}(A, B) \) for positive integers \( A \geq B \), which is given by the following recursive pseudocode:

   ```
   if B divides A then return B
   else return GCD(B, A mod B)
   ```

   Show that, if initialized on \( n \)-bit integers \( A \geq B \), Euclid’s algorithm halts after at most \( 2^n \) iterations. [Hint: Let \( A_t \geq B_t \) be the arguments to the GCD function at the \( t \)th iteration, so that \( A_1 = A \) and \( B_1 = B \). What can you say about the decrease of \( A_t \), as a function of \( t \)?]

3. Show that any language \( L \) containing only finitely many strings is regular.

4. Show that, if \( L_1 \) and \( L_2 \) are any two regular languages, then \( L_1 \cap L_2 \) is also a regular language.

5. Let \( L = \{x \in \{a, b\}^* : x \) does not contain two consecutive \( b \)'s\}. Write a regular expression for \( L \).

6. Let \( L \subseteq \{a, b\}^* \) be the language consisting of all palindromes: that is, strings like \( abba \) that are the same backwards and forwards. Using the pigeonhole principle, show that \( L \) is not regular.

7. Concatenation of regular languages

   (a) Let \( L \subseteq \{a, b, c\}^* \) be the language consisting of all strings \( w \) that can be expressed as \( w_1 \circ w_2 \), where \( w_1 \) contains an even number of \( b \)'s, \( w_2 \) contains a number of \( c \)'s that is divisible by 3, and \( \circ \) denotes string concatenation. Show that \( L \) is regular, by constructing an NDFA that recognizes \( L \).

   (b) Let \( L \subseteq \{a, b\}^* \) be the language consisting of all strings \( w \) that can be expressed as \( w_1 \circ w_2 \), where \( w_1 \) contains an even number of \( b \)'s and \( w_2 \) contains a number of \( b \)'s that is divisible by 3. Construct a DFA that recognizes \( L \). [Hint: You could do this by first constructing an NDFA and then using the simulation of NDFA’s by DFA’s, but that’s working way too hard!]

   (c) Generalize part a. to show that, if \( L_1 \) and \( L_2 \) are any two regular languages, then

   \[
   L = \{w_1 \circ w_2 | w_1 \in L_1, w_2 \in L_2 \}
   \]

   is also a regular language.