Exercises 2WC12 week 3

Exercise 1

Fermat's method of factoring n works as follows:

- start with $X = \lfloor \sqrt{n} \rfloor$
- do
  - increase X by 1
  - compute $Y = \sqrt{X^2 - n}$

  until Y is an integer
- recover factors p, q from X, Y

a) Show how to recover the factors p, q from X, Y. (Note that in general they are not necessarily primes; explain what happens if n itself is prime.)

b) Assuming that n is an RSA-modulus, i.e. is a product of two different primes p, q, show that Fermat's method does indeed factor n in a finite number of steps.

c) Show that if $|p - q| < n^{1/4}$ then this method is extremely efficient (count how many times the do-loop is executed). Hint: $0 < X - \sqrt{n} = (X^2 - n)/(X + \sqrt{n}) < \ldots$.

d) Provide a nice example where $|p - q|$ is slightly larger than $n^{1/4}$ (but not too large, because the complexity very quickly grows out of hand).

Exercise 2

a) Suppose an attacker knowing an RSA-modulus n but not its prime factors p, q can easily recover them if he also knows $\phi(n)$.

b) Let the RSA-modulus $n = pq$ be balanced, i.e. p, q have the same number of bits. Then show that n and $\phi(n)$ share about the first half of their bits, so that the attacker 'only' has to find the last half of the bits of $\phi(n)$ in order to factor n.