Exercise 9.1 (Pollard’s $p - 1$ method). We consider Pollard’s $p - 1$ method from the lecture.

- **How can one find all primes $\leq B$, where $B \in \mathbb{N}$?**
  
  **Solution.** Sieving techniques like the sieve of Eratosthenes.

- **Assume we want to factor $n = 12827$ using Pollard’s $p - 1$ method. The bound $B$ is chosen as $B = 6, 13, 27$ respectively. Discuss the impact of these choices on the outcome of the algorithm.**
  
  **Solution.** We have $n = 101 \cdot 127$ and $100 = 2^2 \cdot 5^2$, $126 = 2 \cdot 3^2 \cdot 7$. We consider the following cases:

  - $B = 6$: The prime powers are $2^3, 3^8, 5^5$ and $g = 101$.
  - $B = 13$: The prime powers are $2^{13}, 3^8, 5^5, 7^4, 11^3, 13^3$ and $g = 12827 = n$.
  - $B = 27$: The prime powers are $2^{13}, 3^8, 5^5, 7^4, 11^3, 13^3, 17^3, 19^3, 23^3$ and $g = 12827 = n$.

- **Discuss in general the cases where the algorithm fails to find a factor of $n$.**
  
  **Solution.** Discussed in the lecture.

Exercise 9.2. The public key of an RSA cryptosystem is given by $(n, e) = (247, 17)$.

- **Encrypt the message $m = 101$ using the public key.**
  
  **Solution.** We have $m^e \pmod{n} = 225 \pmod{274}$.

- **Compute the private key.**
  
  **Solution.** This can be done by factoring $n = 13 \cdot 19$ and computing $\varphi(n) = 12 \cdot 18 = 216$. Now compute

  $$e^{-1} \pmod{\varphi(n)} = 89 \pmod{216} =: d$$

  Thus the private key is $d = 89$. 
Decrypt the message $m = 42$ using the private key you computed.

**Solution.** We have $m^d \pmod{n} = 100 \pmod{274}$.

Assume that the exponentiation with the secret exponent $d$ are performed with the CRT. Calculate $d \pmod{p-1}$ and $d \pmod{q-1}$, $N_p, N_q$ (notation as in C.35).

**Solution.** We have $d \pmod{p-1} = 89 \pmod{12} = 5 \pmod{12}$ and $d \pmod{q-1} = 89 \pmod{18} = 17 \pmod{18}$. Now we are looking for the unique $N_p \in \mathbb{Z}_n$ such that $N_p \equiv 1 \pmod{p}$ and $N_p \equiv 0 \pmod{q}$. In other words: We have $k, \ell \in \mathbb{N}$, such that $N_p - 1 = kp$ and $N_p = \ell q$. This is equivalent to $\ell q - kp = 1$. Since $\gcd(p, q) = 1$ the values $k, \ell$ can be found using the extended Euclidean algorithm. Carrying out the computation we find $N_p = 209$ and $N_q = 39$.

Decrypt the message $m = 42$ using the CRT.

**Solution.** We compute $x_p := m^d \pmod{(p-1)} \pmod{p} = 42^5 \pmod{13} = 9$ (mod 13) and $x_q := m^d \pmod{(q-1)} \pmod{q} = 42^{17} \pmod{19} = 5$ (mod 19). We have $m^d \equiv N_p x_p + N_q x_q \equiv 209 \cdot 9 + 39 \cdot 5 \equiv 100 \pmod{247}$, which was what we expected.

**Exercise 9.3** (Homework: Multiplicativity of RSA). (6 points)

Let $m_1, m_2, m_3$ be three messages with known signatures $m_1^d \pmod{n}$, $m_2^d \pmod{n}$ and $m_3^d \pmod{n}$. Let $m := m_1 \cdot m_2 \cdot m_3 \pmod{n}$. Compute $m^d \pmod{n}$.

**Remark:** Hash functions prevent the aimed construction of meaningful messages $m$ to exploit the multiplicity of the RSA algorithm. Also padding has positive influence since finding such relations is more difficult for larger integers.

**Solution.** We have $m_1^d \cdot m_2^d \cdot m_3^d \equiv (m_1 \cdot m_2 \cdot m_3)^d \equiv m^d \pmod{n}$.

**Exercise 9.4** (Homework: RSA). (6 points)

Let $n = pq \in \mathbb{N}$ with $p, q$ prime be an RSA modulus, $e \in \mathbb{Z}_n^*$ and $d = e^{-1} \pmod{\varphi(n)}$. Prove:

$$(x^e)^d = x = (x^d)^e \pmod{n}$$

**Hint:** CRT!

**Solution.** Two solutions possible: Either you apply Euler’s theorem on the equation modulo the composite number $n$ (you have to take special care if $x$ is a multiple of $p$ or $q$!). Or you apply two times Fermat’s little theorem on the equations mod $p$ and mod $q$, respectively.
Exercise 9.5 (Homework: Pollard $p - 1$). (8 points)

Here you have the choice which task you want to solve:

- Either: Implement Pollard’s $p - 1$ algorithm (presented in class) and factor the number $n = 504380101$. Hand in the (commented) source code, the search bound $B$ you used as well as the factors.

- Or: Factor $n = 289593956703807855037$ with a computer algebra system of your choice. Use this knowledge to propose an appropriate smoothness bound $B$ for Pollard’s $p - 1$ algorithm that yields a successful attack. Justify your proposal. Estimate the number of modular squarings and multiplications that are necessary for one run of Pollard’s algorithm.

Solution. We have $289593956703807855037 = p \cdot q$ with $p = 15728234383$ and $q = 18412362739$. Further $p - 1 = 2 \cdot 3^2 \cdot 7 \cdot 124827257$ and $q - 1 = 2 \cdot 3^2 \cdot 7 \cdot 11 \cdot 41 \cdot 457 \cdot 709$. Thus an appropriate smoothness bound would be $B = 709$. 