Exercise 1:  Bit complexity of the AKS primality test (4 points)

Show that the AKS primality test requires $\tilde{O}(\log^{10.5} n)$ bit operations to decide whether the number $n$ is prime. You can use that $\gcd(a, b)$ of two $\tau$-bit integers $a$ and $b$ can be computed in time $\tilde{O}(\tau)$.

Note that you only need to prove the polynomial time bound with exponent 10.5; a proof of the bound $\tilde{O}(\log^{7.5} n)$ as stated in the lecture is more involved.

Exercise 2:  Chinese remaindering over the ring of integers (4 points)

1. Determine the smallest positive integer $x$ satisfying
   
   \[ x \equiv 4 \mod 7, \quad x \equiv 5 \mod 11, \quad x \equiv 6 \mod 13. \]

2. How many integers $x$ between 0 and $10^6$ are common solutions of the following congruences?
   
   \[ x \equiv 3 \mod 13, \quad x \equiv 4 \mod 15, \quad x \equiv 5 \mod 17. \]

Exercise 3:  Chinese remaindering over polynomial rings (4 points)

1. Consider a Euclidean domain $R$ and elements $a, b, c \in R$. Prove that
   
   \[ a \cdot x \equiv b \mod c \]

   has a solution $x \in R$ if and only if $g := \gcd(a, c)$ divides $b$. Show that, in the latter case, the congruence is equivalent to
   
   \[ \frac{a}{g} \cdot x \equiv \frac{b}{g} \mod \frac{c}{g}. \]

2. Determine the solution $f \in \mathbb{Z}[x]$ of the system of congruences
   
   \[
   \begin{align*}
   f & \equiv 1 \mod x + 3, \\
   x \cdot f & \equiv x + 1 \mod x^2 + 2, \\
   (x + 3) \cdot f & \equiv x^2 + 1 \mod x^3 + 2
   \end{align*}
   
   with the smallest possible degree.
Exercise 4: Computing a small separating linear form for points on integer grids

(4 points + 4 bonus points)

Let \( X = \{x_1, \ldots, x_n\} \subset \mathbb{Z} \) be a set of integers with \(|x_i| < 2^\tau\) and let \( d \) be an integer with \( d \geq 2 \). We consider the problem of computing a separating linear form of small size for \( X^d \). More precisely, compute coefficients \( a_k \) such that the linear map

\[
    s_a : \mathbb{Z}^d \to \mathbb{Z}, \quad x \mapsto a_1x_1 + \cdots + a_dx_d
\]

is injective on \( X^d \), that is

\[
    s_a(x_{i_1}, \ldots, x_{i_d}) = \sum_{k=1}^d a_k \cdot x_{i_k} \neq \sum_{k=1}^d a_k \cdot x_{j_k} = s_a(x_{j_1}, \ldots, x_{j_d})
\]

for all pairs of distinct \( d \)-tuples \((x_{i_1}, \ldots, x_{i_d}) \neq (x_{j_1}, \ldots, x_{j_d})\) in \( X^d \).

“Small size” means that the coefficients \( a_k \) of \( s_a \) have bitsize bounded by \( \tilde{O}(d(\log \tau + \log n)) \).

Give an algorithm which solves this task in a polynomial number (in \( n \), \( d \) and \( \tau \)) of bit operations and provide a runtime analysis.

**Hint:** Determine primes \( p_1, \ldots, p_d \) such that

\[
    (x_{i_1} \mod p_1, \ldots, x_{i_d} \mod p_d) \neq (x_{j_1} \mod p_1, \ldots, x_{j_d} \mod p_d)
\]

for all distinct \( d \)-tuples \((x_{i_1}, \ldots, x_{i_d}) \neq (x_{j_1}, \ldots, x_{j_d})\) in \( X^d \).