1. Page 193, first 5 lines in Section 1.2.1:

1.1 When the output distribution of the filtering function is biased, give an approximation of the keystream length required for distinguishing the sequence produced by FLIP from a random sequence?

1.2 In which practical situation could such a distinguisher help recover some information on the plaintext from the ciphertext?

2. Page 194, Lines 15-16:

2.1 Prove that, for any prime number $p$ and any integer $i$,

$$(1 + X)^{p^i} = 1 + X^{p^i} \mod p.$$ 

2.2 Prove that, for any prime number $p$,

$$\binom{n}{k} = \prod_{i=0}^\ell \binom{n_i}{k_i} \mod p$$

where $n = \sum_{i=0}^\ell n_i p^i$ and $k = \sum_{i=0}^\ell k_i p^i$ with $n_i, k_i \in \{0, \ldots, p-1\}$. This can be deduced from the previous result by expanding $(1 + X)^n$ in two different ways. Recall that, by convention, $\binom{a}{b} = 0$ when $b > a$ and $\binom{0}{0} = 1$.

2.3 Deduce that $\binom{n}{k}$ is even for all $1 \leq k < n$ if and only if $n$ is a power of 2.

3. Page 197, Prop 1:

3.1 The statement of Proposition 1 is not correct. What would be the correct statement?

3.2 Prove by induction that, for any $n$-variable Boolean function $f$ with algebraic normal form $\sum_{u \in \mathbb{F}_2^n} a_u x^u$, we have

$$f(x) = \sum_{u \leq x} a_u \mod 2$$

for all $x \in \mathbb{F}_2^n$ where $u \leq x$ means that $u_i \leq x_i$ for all $1 \leq i \leq n$.

3.3 Deduce that the values of the function $f$ with ANF $\sum_{1 \leq i < j \leq n} x_i x_j$ correspond to

$$f(x) = \frac{w_H(x)}{2} \mod 2.$$ 

3.4 Prove by induction on the number of variables that the function with ANF $\sum_{1 \leq i < j \leq n} x_i x_j$ is bent. [Hint: Compute the Walsh transform of the $(n + 2)$-variable function at point $(a, a_{n+1}, a_{n+2})$ for $a \in \mathbb{F}_2^n$ and for fixed $a_{n+1}, a_{n+2} \in \mathbb{F}_2$.]
4. Page 197, Remark 2:

4.1 Prove that, similarly to the second item in Question 3, the coefficients of the ANF \( \sum_{u \in \mathbb{F}_2^n} a_u x^u \) of a function \( f \) can be deduced from its values by

\[
a_u = \sum_{x \leq u} f(x) \mod 2
\]

for all \( u \in \mathbb{F}_2^n \).

4.2 Let \( \varphi \) be an \( n \)-variable symmetric Boolean function such that \( \varphi(x) = c_w \) for all \( x \) with \( w_H(x) = w \). Compute the coefficients of the ANF of \( \varphi \) as a function of \( c_0, \ldots, c_n \). Deduce that the ANF \( \sum_{u \in \mathbb{F}_2^n} a_u x^u \) of \( \varphi \) is such that \( a_u \) takes the same value when \( u \) varies in \( E_{n,k} \), for any \( k \).

4.3 Recall that the authors proved that an \( n \)-variable function \( f \) has nonlinearity 0 on all \( E_{n,k} \), \( 1 \leq k < n \), if and only if

\[
f(x) = \varphi_0(x) + \sum_{i=1}^{n} \varphi_i(x)x_i
\]

where all \( \varphi_i \) are symmetric Boolean functions (i.e., each \( \varphi_i \) is constant on each \( E_{n,k} \)). Derive from the previous question that any function \( f \) having nonlinearity 0 on all \( E_{n,k}, 1 \leq k < n \), can be written as

\[
f(x) = \ell'_0(x) + \sum_{i=1}^{n} \sigma_i(x)\ell'_i(x)
\]

where all \( \ell'_i \) are affine functions, and \( \sigma_i \) denotes the function whose ANF consists of the sum of all monomials of degree \( i \).

5. Page 200, Proposition 3:

5.1 Line 3 in the proof: explain why, if \( f \) is a linear function with \( d \) monomials, then

\[
w_H(f)_k = \sum_{i \text{ odd}} \binom{d}{i} \binom{n-d}{k-i}.
\]

5.2 Line 6 on Page 201: Explain how it can be deduced that \( w_H(qf)_2 \geq \lfloor n/4 \rfloor \).

6. Page 202, Theorem 2: Based on this theorem, give the ANF of a weightwise perfectly balanced function of 4 variables.

7. Page 218, Table 1: Explain how the number of variables \( N \) of the filtering function is computed from \( (n_1, n_2, nb, h) \).

8. Page 220, first 3 lines after Table 3:

8.1 The attack described by the authors consists in building a linear system and in seeing the key-recovery as a decoding problem. Give the expression of the linear system that the authors have in mind. What is the decoding problem that the attacker has to solve?

8.2 Deduce the minimal number of keystream bits needed in the attack.