1. Prove that if $C$ is permutation-equivalent to $C'$, then $C^\perp$ is permutation-equivalent to $C'^\perp$.

2. **End of Page 4**: Explain why the two notions of equivalence defined in Definitions 2.1 and 2.2 coincide in the binary case.

3. **Page 5, Definition 2.4**: Prove that the code defined by

$$C_J = (C + E_J) \cap E_{I_\setminus J}$$

consists of all elements of $C$ where the coordinates indexed by $J$ are replaced by zeroes.

4. **Page 5, (b):**
   - Prove that, if $J \cap \text{Supp}(B) = \emptyset$, then $(B + C)_{\setminus J} = B_{\setminus J} + C_{\setminus J}$.
   - Exhibit an example of $B$ and $C$ for which $(B_{\setminus J} + C_{\setminus J}) \neq (B + C)_{\setminus J}$.

5. **Page 6, (d):**
   - Prove that, for any two linear spaces $U$ and $V$ in $F_q^n$,
     $$\left( U \cap V \right)^\perp = \left( U^\perp + V^\perp \right).$$
   - Deduce that $(C_J)^\perp = (C^\perp)_{\setminus J} \oplus E_J$.

6. **Page 10**: Explain why this algorithm is named *support splitting algorithm*.

7. **Page 14, Definition 5.1**: What is the dual of the hull of $C$?

8. **Pages 14-15**: Explain why, in the algorithm, the weight distribution of the hull is considered instead of that of the whole code?

9. **Page 14**:  
   - Prove that the signature $i \mapsto W(C_i)$ is self-dual (in the sense of Definition 3.7), where $W(C)$ denotes the weight distribution of $C$.
   - Does the same property hold when $C_i$ is replaced by $(\mathcal{H}(C))_i$? Why?

10. **Pages 17 and 18**: Explain why all polynomials corresponding to the weight distributions in Tables 2-5 contain only monomials of even degree.

11. **Pages 17-18**: What are the positions which are not discriminated after the first refinement? And after the third one?

12. **Section 5.2**: If $C$ is a cyclic code, will be the signature of Section 5.2 discriminant? How to address this issue?

13. **Page 21, “When the hull has dimension zero”**: Explain why, when the hull has dimension zero, there exists a unique pair $(x^{(i)}, y^{(i)}) \in C \times C^\perp$ such that $x^{(i)} - y^{(i)} = e^{(i)}$.

14. Explain why the support splitting algorithm applied to Reed-Muller codes has a complexity which is exponential in the code dimension.

*Turn the page please.*
15. Let $\mathbb{F}_q$ be a fixed finite field and $n$ an integer with $n < q$. Consider the McEliece scheme instantiated with Reed-Solomon codes $RS(x)$ where the vector $x = (x_1, ..., x_n)$ is the secret key. A brute-force key-recovery attack consists in enumerating all the possible secret keys $x'$ and compare the codes $RS(x')$ with the public code. How could the support splitting algorithm improve this attack?