Error Correcting Codes: Combinatorics, Algorithms and Applications Spring 2010

HOMEWORK 0

The problems in this homework are to help you get more comfortable with the background material. Most of these questions would be useful in solving the real homework problems. Please do **not** submit answers to these questions. A warning: I have not tried to formally solve all the problems, so some of them might be tougher than they look. Also I'll try to make the problem statements verbose so that they are somewhat self-sufficient.
Unlike the real homework problems, for these problem, feel free to refer to any source that might help you.
This document will be updated with more problems as the semester moves on.

---

1. (**System of Linear Equations**) In this problem we will look at the problem of solving a system of linear equations over $\mathbb{F}_q$. That is, one needs to solve for unknowns $x_1, \ldots, x_n$ given the following m linear equations (where $a_{i,j}, b_i \in \mathbb{F}_q$ for $1 \leq i \leq m$ and $1 \leq j \leq n$):

$$a_{1,1}x_1 + a_{1,2}x_2 + \cdots + a_{1,n}x_n = b_1.$$

$$a_{2,1}x_1 + a_{2,2}x_2 + \cdots + a_{2,n}x_n = b_2.$$

$$\vdots$$

$$a_{m,1}x_1 + a_{m,2}x_2 + \cdots + a_{m,n}x_n = b_m.$$

   (a) (*Warm-up*) Convince yourself that the above problem can be stated as $A \cdot \mathbf{x}^T = \mathbf{b}^T$, where A is an $m \times n$ matrix over $\mathbb{F}_q$, $\mathbf{x} \in \mathbb{F}_q^n$ and $\mathbf{b} \in \mathbb{F}_q^m$.

   (b) (*Upper Triangular Matrix*) Assume $n = m$ and that A is upper triangular, i.e. all diagonal elements ($a_{i,i}$) are non-zero and all lower triangular elements ($a_{i,j}$, $i > j$) are 0. Then present an $O(n^2)$ time[1] algorithm to compute the unknown vector $\mathbf{x}$.

   (c) (*Gaussian Elimination*) Assume that A has *full rank* (or equivalently a *rank*[2] of n.)

   - Prove that the following algorithm due to Gauss converts A into an upper triangular matrix. By permuting the columns if necessary make sure that $a_{1,1} \neq 0$. (Why can one assume w.l.o.g. that this can be done?) Multiply all rows $1 < i \leq n$ with $\frac{a_{1,1}}{a_{i,1}}$ and then subtract $a_{1,j}$ from the (i,j)th entry $1 \leq j \leq n$. Recurse with the same algorithm on the $(n-1) \times (n-1)$ matrix A' obtained by removing the first row and column from A. (Stop when $n = 1$.)
   - What happens if A does not have full rank? Show how one can modify the algorithm above to either upper triangulate a matrix or report that it does not have full rank. (Convince yourself that your modification works.)

[1] For this problem, any basic operation over $\mathbb{F}_q$ takes unit time.

[2] A rank of an $n \times n$ matrix is the maximum number of linearly independent rows. It can be shown that this is the same as the maximum number of linearly independent columns. This definition also works for general $m \times n$ matrices.

- Call a system of equations $A \cdot \mathbf{x}^T = \mathbf{b}^T$ *consistent* if there exists a solution to $\mathbf{x} \in \mathbb{F}_q^n$. Show that there exists an $O(n^3)$ algorithm that finds the solution if the system of equations is consistent and A has full rank (and report "fail" otherwise).

(d) ($m < n$ *case*) Assume that A has full rank, i.e. has a rank of m. In this scenario either the system of equations is inconsistent or there are q^{n-m} solutions to $\mathbf{x}$. Modify the algorithm from above to design an $O(m^2 n)$ time algorithm to output the solutions (or report that the system is inconsistent).

- Note that in case the system is consistent there will be q^{n-m} solutions, which might be much bigger than $O(m^2 n)$. Show that this is not a problem as one can represent the solutions as system of linear equations. (I.e. one can have $n-m$ "free" variables and m "bound" variables.)

(e) ($m > n$ *case*) Assume that A has full rank, i.e. a rank of n. In this scenario either the system of equations is inconsistent or there is a unique solution to $\mathbf{x}$. Modify the algorithm from above to design an $O(m^2 n)$ time algorithm to output the solution (or report that the system is inconsistent).

(f) (*Non-full rank case*) Give an $O(m^2 n)$ algorithm for the general case, i.e. the $m \times n$ matrix A need not have full rank. (The algorithm should either report that the system of equations is inconsistent or output the solution(s) to $\mathbf{x}$.)

2. (**Geometric Series Sum for Finite Fields**) Let q be a prime power. Let $x \in \mathbb{F}_q$ such that $x \notin \{0, 1\}$. Then prove that for any $n \leq q - 1$:

$$\sum_{i=0}^{n} x^i = \frac{x^{n+1} - 1}{x - 1}.$$

Hint: Use induction on n. Note that for any $\alpha \in \mathbb{F}_q \setminus \{0\}$, $\alpha^0 = 1$