CS 70 Discrete Mathematics and Probability Theory
Fall 2016 Seshia and Walrand

HW 7

1 Sundry

Before you start your homework, write down your team. Who else did you work with on this homework? List names and email addresses. (In case of hw party, you can also just describe the group.) How did you work on this homework? Working in groups of 3-5 will earn credit for your "Sundry" grade.

Please copy the following statement and sign next to it:

I certify that all solutions are entirely in my words and that I have not looked at another student's solutions. I have credited all external sources in this write up.

2 Problems

1. **Error-Correcting Codes**

   (a) Recall from class the error-correcting code for erasure errors, which protects against up to k lost packets by sending a total of $n+k$ packets (where n is the number of packets in the original message). Often the number of packets lost is not some fixed number k, but rather a *fraction* of the number of packets sent. Suppose we wish to protect against a fraction α of lost packets (where $0 < \alpha < 1$). What is the total number of packets that we need to send (as a function of n and α)?

   (b) Repeat part (a) for the case of general errors.

2. **Berlekamp-Welch for general errors**

   Suppose that Hector wants to send you a length $n = 3$ message, m_0, m_1, m_2, with the possibility for $k = 1$ error. For all parts of this problem, we will work mod 11, so we can encode 11 letters as shown below:

A	B	C	D	E	F	G	H	I	J	K
0	1	2	3	4	5	6	7	8	9	10

   Hector encodes the message by finding the degree ≤ 2 polynomial $P(x)$ that passes through $(0, m_0)$, $(1, m_1)$, and $(2, m_2)$, and then sends you the five packets $P(0), P(1), P(2), P(3), P(4)$ over a noisy channel. The message you receive is

$$\text{DHACK} \Rightarrow 3, 7, 0, 2, 10 = r_0, r_1, r_2, r_3, r_4$$

   which could have up to 1 error.

   (a) First, let's locate the error, using an error-locating polynomial $E(x)$. Let $Q(x) = P(x)E(x)$. Recall that

$$Q(i) = P(i)E(i) = r_i E(i), \quad \text{for} \quad 0 \leq i < n + 2k$$

   What is the degree of $E(x)$? What is the degree of $Q(x)$? Using the relation above, write out the form of $E(x)$ and $Q(x)$ in terms of the unknown coefficients, and then a system of equations to find both these polynomials.

   (b) Solve for $Q(x)$ and $E(x)$. Where is the error located?

   (c) Finally, what is $P(x)$? Use $P(x)$ to determine the original message that Hector wanted to send.

   Hint: The message refers to a US federal agency.

3. **Why Work with Primes?** In class, you learned about erasure codes and error correcting codes, and prime numbers played a central role in both kinds of codes – since all calculations were supposed to be done modulo a *prime number*. In this problem, we will see why this is a crucial requirement, and explore what happens if this requirement is relaxed in a naive manner.

For this problem, assume that Alice wants to send n packets to Bob, across an "erasure channel" (Check detailed definition below). Let us say all calculations are done modulo $N = 12$ (note that this is *not* a prime number).

Erasure Channel: Let us say Alice sends $n+1$ packets to Bob, and Bob receives at least n of these packets intact. That is, the channel can erase at most 1 packet, and if it does so, Bob gets to know which packet was erased (although he does not know the contents of the erased packet).

(a) Suppose $n = 1$. That is, Alice wants to send only 1 packet to Bob (plus one redundant packet to compensate for erasure). Would the scheme discussed in class work with $N = 12$? What are all the possible 2-packet lists that Alice could transmit? In each case, would Bob be able to recover Alice's message in spite of a possible erasure? Would Alice or Bob face any problems because they are doing their calculations modulo 12?

(b) Now suppose $n = 2$. That is, Alice now wants to send 2 packets to Bob (plus one redundant packet to compensate for erasure). Now, would there be any problems because $N = 12$?

(c) Now let $n = 3$ (3 packets plus one additional packet to compensate for erasure). Assume that Alice wants to encode messages into "systematic" codewords (with the first few evaluations of the polynomial being the message itself). Prove that Alice can no longer send arbitrary messages of her liking to Bob, by showing that it would be impossible for Alice to send the message $(11, 6, 2)$. Find 2 other examples of messages that Alice cannot send to Bob.

4. **To Infinities and Beyond**

Show whether each of the following sets is finite, countably infinite, or uncountable:

(a) $\mathbb{N}$ (the set of all natural numbers)

(b) $\mathbb{Z}$ (the set of all integers)

(c) $\mathbb{Q}$ (the set of all rational numbers, i.e., numbers that can be expressed in the form a/b, where $a, b \in \mathbb{Z}$ and $b \neq 0$)

(d) $\mathbb{R}$ (the set of all real numbers)

(e) $\mathbb{C}$ (the set of all complex numbers)

(f) $\{0,1\}^*$ (the set of all finite-length binary strings)

(g) $\{0,1,2\}^*$ (the set of all finite-length ternary strings)

(h) The set of all primes.

(i) The set of all graphs

5. **More Countability**

Given:

- A is a countable set, non-empty set. Forall $i \in A$, S_i is an uncountable set.

- B is an uncountable set. Forall $i \in B$, Q_i is a countable set.

For each of the following, decide if the expression is "Always Countable", "Always Uncountable", "Sometimes Countable, Sometimes Uncountable."

For the "Always" cases, prove your claim. For the "Sometimes" case, provide two examples – one where the expression is countable, and one where the expression is uncountable.

(a) $\cup_{i \in A} S_i$

(b) $\cap_{i \in A} S_i$

(c) $\cup_{i \in B} Q_i$

(d) $\cap_{i \in B} Q_i$

(e) $A \cap B$

6. **Printing All x Where $M(x)$ Halts**

Prove that it is possible to write a program P which:

- takes as input M, a java program
- runs forever, and prints out strings to the console
- for every x, if M(x) halts, then P(M) eventually prints out x
- for every x, if M(x) does NOT halt, then P(M) never prints out x