Midterm Test 1 Feedback

Question 1 – average mark 61%

a) This part was generally well done

b) Many students correctly designed the tree for optimal encoding but did not write down the encoding for each message – they scored 5 out of 7 marks for this part of the question.

c) Some students simply added up the number of bits in encoding for each message and divided by 5 – this would be correct if each message occurs with the probability $\frac{1}{5}$. However, this is not the case and the correct solution is to multiply the number of bits in encoding for each message by the probability of that message, and ten add them up.

Some other students computed entropy instead the average number of bits for a given encoding. Note that if we construct Huffman codes, they average number of bits per message will be close but not necessarily equal to the entropy.

Question 2 – average mark 72%

This questions was generally well done; many students made a mistake in part b) – as 19 is a prime number many students answer “TRUE” – the correct answer is “FALSE” as 0 does not have a multiplicative inverse.

Another part that was not very well done was c) - many students forgot that multiples of 5 and 7 are also not relatively prime with 35.

Question 3 – average mark 49%

a) Some students provided a formula for either Euler’s theorem, or calculating multiplicative inverse using Euler’s theorem; however, what was required here was the formula for calculating the Euler’s Totient Function $\Phi(n) = \prod_{i=1}^{t} {p_i}^{e_i - 1}(p_i - 1),$ where $n = \prod_{i=1}^{t} {p_i}^{e_i}$.

b) This part was generally well done.

c) This part was not well done. Many students were not on top of this – we will discuss it in the lectures.

Some students defined the Absolute Rate of Language as the amount of information in the language, while the correct answer is the maximum amount of information per character. Also, same students wrote that the Absolute Rate of Language when all characters are equally likely, while the correct answer is when all sequences of characters (of a given length) are equally likely. Please observe the subtle difference between these two statements.

Consider the following example: A language L over alphabet {0,1} contains the following words (messages): 01 and 10. Note that here all characters are equally likely but all sequences are not – 00 and 11 appear with probability 0, while 01 and 10 appear with the probability 0.5 each. The rate of language L the entropy of the message (1 bit) divided by the number of character per message (2) – therefore the rate of language is 0.5bits. Note that this is NOT equal to the Absolute Rate of Language, which is lg 2 = 1bit.

Question 4 – average mark 54%

Many students did not attempt this question. Of those that attempted the question, the most common issue was to get the formula for the multiplicative inverse or Euler's Totient Function incorrect.

Some students noticed that a form of Euler's Theorem is given in Question 5a, but then did not correctly apply the formula given in Question 5a to this question. In the context of $GF(2^n)$ with irreducible polynomial $p(x)$, the formula for finding the multiplicative inverse for a, (i.e. a^{-1}) using Euler's Theorem is given by $a^{-1} = a^{\phi(p(x))-1} \mathrm{mod}\, p(x)$.

Recall that when working in integers, Euler's Totient Function $\phi(n)$ counts the number of positive integers relatively prime to n. When working in $GF(2^n)$, with irreducible polynomial $p(x)$, Euler's Totient Function $\phi(p(x))$ counts the number of elements in the field relatively prime to $p(x)$. Because $p(x)$ is irreducible, this is simply all of the elements, except for the polynomial with all zero coefficients. Thus, you could find $\phi(p(x))$ by enumerating all the elements, or by using the formula $\phi\big(p(x)\big) = 2^n - 1$.

Some students forgot to verify $a \times a^{-1} \bmod p(x) = 1$, and lost marks for that part of the question.

Question 5 – average mark 83%

a) Most issues were in calculating Euler's Totient Function incorrectly or failing to correctly rearrange Euler's Theorem to find the multiplicative inverse. The fast exponentiation algorithm itself was generally done well.

b) The Chinese Remainder Theorem was generally done well, with most issues being arithmetic errors. d_1 and d_2 were generally identified correctly, but a fair number of students made errors in identifying $x_1, x_2, y_1 \text{ or } y_2$ by incorrectly solving the smaller multiplicative inverse equations (e.g. claiming that $7x_1 \bmod 8 \Rightarrow x_1 = 1$). Overall, a good understanding of the CRT was shown. A side note is that several students made the arithmetic harder for themselves by not applying the modulo operator early, only applying the modulo operator at the last step – this appeared to contribute to arithmetic errors.

c) This was generally done well. The main reasons for loss of marks on this question were either arithmetic errors, or not attempting the question.