These solutions aren't intended to be comprehensive. Make sure to ask me if you have any questions or find any typos. In a few cases you might have gotten full credit if your answers didn't quite match what's here as long as you demonstrated the required knowledge in a later part of the problem.

(1) **(i):** There are four choices for each of $f(a)$, $f(b)$ and $f(c)$, so $4^3 = 64$ overall.
(ii): An injective function has 4 choices for $f(a)$, only 3 choices for $f(b)$, and 2 choices for $f(c)$, so $4 \cdot 3 \cdot 2 = 24$ overall.
(iii): None; with three inputs and four outputs, a surjective function is impossible.
(iv): False; it's at a minimum if a $p_i = 1$.
(v): True

(2) **(i):** Many correct answers were possible, depending on the choices you made when constructing your tree. Here's my code:

$$f(A)00$$
$$f(B)01$$
$$f(C)10$$
$$f(D)110$$
$$f(E)1110$$
$$f(F)1111$$

(ii): The entropy is

$$\begin{aligned} H(X) &= -\frac{1}{4}\log_2\frac{1}{4} - \frac{1}{4}\log_2\frac{1}{4} - \frac{1}{4}\log_2\frac{1}{4} - \frac{1}{8}\log_2\frac{1}{8} - \frac{1}{16}\log_2\frac{1}{16} - \frac{1}{16}\log_2\frac{1}{16} \\ &= \cdots = \frac{19}{8} \end{aligned}$$

The average word length is the *weighted* average of the lengths, or

$$\frac{1}{4}\cdot 2 + \frac{1}{4}\cdot 2 + \frac{1}{4}\cdot 2 + \frac{1}{8}\cdot 3 + \frac{1}{16}\cdot 4 + \frac{1}{16}\cdot 4 = \frac{19}{8}$$

The Noiseless Coding Theorem tells us to expect that $H(X) \leq$ average word length[1] which is true for our code. (In fact, they're equal, which means this is an optimal code.)

(3) **(i):** The minimum Hamming distance between any two of these words is 3.
(ii): Minimum Distance Decoding means that, after receiving y, you compute the distances $d(y, x)$ for all valid codewords x and decode y to whichever word is closest.

- 00110 decodes to $b = 00111$; $d(00110, 00111) = 1$.
- 10001 decodes to $c = 11001$; $d(10001, 11001) = 1$.
- 11001 decodes to $c = 11001$; it *is* c, and $d(11001, 11001) = 0$.

(iii): No; we proved this in class, first for a minimum distance of 3, and then a minimum distance of $d = 2k+1$. It was an application of the fact that Hamming distance satisfies

[1] At least in the binary case; otherwise $H(X)$ is replaced with $\frac{H(X)}{\log_2 |\Sigma|}$.

the triangle inequality:

$$d(x, y) \leq d(x, z) + d(z, y)$$

In this case, y can't have a distance of 1 to two codewords x and z because then

$$d(x, z) \leq d(x, y) + d(y, z) = 2$$

but $d(x, z)$ must be at least 3 (because that's the minimum distance of our code).

(4) **(i):** Your friend is lying, because the Kraft Inequality is not satisfied:

$$\frac{1}{3^1} + \frac{1}{3^1} + \frac{1}{3^2} + \frac{1}{3^2} + \frac{1}{3^2} + \frac{1}{3^3} = \frac{28}{27} > 1$$

(ii): Such a code exists, because the McMillan Inequality is satisfied:

$$\frac{1}{3^2} + \frac{1}{3^2} + \frac{1}{3^2} + \frac{1}{3^2} = \frac{4}{9} \leq 1$$

(iii): Your answers shouldn't change, because of the curious fact that the Kraft Inequality and the McMillan Inequality are exactly the same. We discussed this in class; if you can find a uniquely decipherable code with given lengths, then you can also find an *instantaneous* uniquely decipherable code with those same lengths.

NOTE: Many people said that their answers *would* change; they said the only requirement for a code to be uniquely decipherable is that it's an injective function, and in part (i) there are 3 words of length one, $3^2 = 9$ words of length 2, and $3^3 = 27$ words of length three, so there are clearly enough words for the code to be injective.

There's more to uniquely decipherable that simply being injective, but the book glosses over this point, so I gave full credit for this answer if it was explained well enough. *I won't give full credit for this on the final, though*, so please learn the following two examples:

Consider the code $f(A) = 0$, $f(B) = 1$, $f(C) = 01$. This is definitely not instantaneous, because $f(A)$ is a prefix of $f(C)$. It is an injective function because $f(A) \neq f(B)$, etc., but it's **NOT** uniquely decipherable. Suppose I receive the following transmission.

0101010101

How should I decode this? As $CCCCC$? $ABABABABAB$? $ABCABCC$? There are many other choices as well; this transmission does not decode uniquely.

So that code is neither instantaneous nor uniquely decipherable. It's actually fairly tricky to create a code which is uniquely decipherable but not instantaneous. Here's one example:

$$\begin{aligned} f(A) &= 0 \\ f(B) &= 01 \\ f(C) &= 011 \\ f(D) &= 111 \end{aligned}$$

This definitely isn't an instantaneous code, but it's hard to see that it is uniquely decipherable. You can't find any two codewords (strings made of of A, B, C, or D) which encode to the same string of 0's and 1's.

Although it's uniquely decipherable, it's hard to decode. You might have to wait until the entire transmission is completed before you can decode it (as opposed to decoding each symbol instantaneously as you receive it). For example, if we've received

$$000\ldots$$

then this might be AAA, or AA and the first part of B, or AA and the first part of C. Once we receive a 1 then things become clear:

$$000110111$$

You should check that the only possible decoding is $AACAD$.

(5) **(i):**

$$\begin{aligned} C &= 1 + p\log_2 p + (1-p)\log_2(1-p) \\ &= \frac{1}{2} + \frac{3}{4}\log_2\frac{3}{4} \end{aligned}$$

This is the highest rate of information we can expect to transmit across the channel with any given code; if a code has a rate R which is greater than C, we cannot expect reliable transmission of data.

(ii):

$$\begin{aligned} R &= \frac{\log_2 \#\text{ of CWs}}{\text{max'm length of CW}} \\ &= \frac{\log_2 8}{4} = \frac{3}{4} \end{aligned}$$

An even parity check digit will detect 1 or 3 bit errors, so all we have to check is the probability of exactly 2 errors:

$$\binom{4}{2}p^2(1-p)^2 = 6\cdot\left(\frac{1}{4}\right)^2\left(\frac{3}{4}\right)^2 = 6\frac{1}{16}\frac{9}{16} = \frac{54}{256}$$

See the calculations on pp63-65 or ask me for help if you don't follow this.