Section 1.1

3. **b.** 001 011 1 decodes as '='.

d. 101 011 1 decodes as 'Parity Error'.

4. **b.** 000 was transmitted.

d. 111 was transmitted.

6. **a.** Any two of the following code words are acceptable:

1010110	$
0110110	.
0000110	3
0010010	9
0010111	=

001 111 0 and 001 010 0 are not code words.

b. As each of the 5 code words above differ from the received code word in only one digit, it is equally likely that an error could occur in any of them to yield our received word. Therefore, there is no reason that one would be preferred over the others as a candidate for the transmitted code word.

8. To detect up to triple errors, we encode 0 as 0000 and 1 as 1111. On the receiving end, we decoded 0000 as 0, 1111 as 1, and any other string of 4 binary digits as an error message. Thus, if at most 3 errors occur, say, in the transmission of 0000, then there will be at most three 1's and at least one 0 in the received string. Since there would be 0's and 1's in the received string, it would be decoded as an error, and the coding scheme detects up to 3 errors.

11. Assume that we are trying to add 1011 + 1101 in binary. The sum is 11000. Suppose that the program makes an error in the addition. Suppose that after adding the right-most column, it forgets to carry the 1 to the next column. Then the result would be 10110, which is wrong in 3 digits. Thus, a single error can lead to many incorrect digits.

Section 1.2

1. b. The Hamming distance between 00000 and 10010 is 2.

d. The Hamming distance between 111000 and 100101 is 4.

2. b.

	101	011	000
101	–	2	2
011	2	–	2
000	2	2	–

$d = 2$.

d.

	01000	00110	11111	10001
01000	–	3	4	3
00110	3	–	3	4
11111	4	3	–	3
10001	3	4	3	–

$d = 3$.

3. b. The Hamming distance of 01100 to

$$\begin{array}{rl} 00110: & 2 \\ 01011: & 3 \\ 10001: & 4 \\ 11100: & 4 \end{array}$$

The most likely transmitted code word is 00110.

d. The Hamming distance of 01011 to

$$\begin{array}{rl} 00110: & 3 \\ 01011: & 0 \\ 10001: & 3 \\ 11100: & 4 \end{array}$$

The most likely transmitted code word is 01011.

5. Suppose we change the same bit place in every code word. Let a and b be two code words before the bit was changed, and a' and b' the code words after the change. If the bit place in question was the same in a and b, then it will also be the same in a' and b'. Likewise, if it was different in a and b, then it will be different in a' and b', but swapped. Since none of the other bit places change, the total number of different bits between a and b stays the same. $H(a,b) = H(a',b')$. Since all of the Hamming distances are the same, the minimum Hamming distance is the same as well.

8. Let a and b be code words such that $H(a,b) = 3$. a and b exists because $d = 3$. Let r be a word that differs from a in exactly one of the places that b differs from a and nowhere else. Then r has a one bit place difference from a and a two bit place difference from b. Thus, r can be received if a is transmitted and a single error occurs. If the encoding scheme corrects all single errors, then it must decode r as a. However, r can also be received when b is transmitted and two errors occur. The encoding scheme cannot detect this double error because it must decode r as a.

11. Let a, b, and c be binary n-tuples.

i. By definition, $H(a,b)$ is the number of bit places in which a and b differ. This obviously cannot be negative. Thus, $H(a,b) \geq 0$.

ii. Suppose $H(a,b) = 0$. This means that a and b agree in every bit place. Thus, $a = b$.

iii. The number of bit places in which a and b differ is the same as the number of bit places in which b and a differ. Therefore, $H(a,b) = H(b,a)$.

iv. $H(a,b)$ is the minimum number of bit places that need to be changed to get from a to b. You can change $H(a,c)$ bit places to get from a to c, and then change $H(c,b)$ bit places to get from c to b. Therefore, you can get from a to b through c in $H(a,c) + H(c,b)$ changes. Since $H(a,b)$ is the minimum number of changes required, we must have $H(a,b) \leq H(a,c) + H(c,b)$.