

CHAPTER 17 – INFORMATION SCIENCE

Binary and decimal numbers – a short review:

For decimal numbers we have 10 digits available (0, 1, 2, 3, ... 9)

10,000	1000	100	10	1
10^4	10^3	10^2	10^1	10^0

4731 =

For binary numbers we have 2 digits available, 0 and 1.

64	32	16	8	4	2	1
2^6	2^5	2^4	2^3	2^2	2^1	2^0

Express the following binary numbers as decimal numbers:

10101 =

11100010 =

Express the following decimal numbers as binary numbers:

55 =

88 =

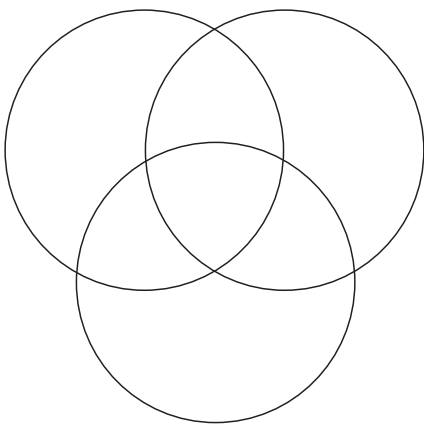
An orbiting satellite can follow 16 different directions that are labeled 0 to 15 in binary (0000 to 1111). Each message is sent as the command along with 3 check digits. The check digits are arranged so that certain sums have even parity. These are called *parity-check sums* where the parity of a number refers to whether a number is even or odd. Even numbers have *even parity* and odd numbers have *odd parity*.

For our satellite, the following sums must be even (0 mod 2)

$$a_1 + a_2 + a_3 + c_1, \quad a_1 + a_3 + a_4 + c_2, \quad \text{and} \quad a_2 + a_3 + a_4 + c_3$$

What are the check digits for command 9?

For this type of parity-check sum, we can use a Venn diagram to help find the check digits or find errors.



Fix the error in the code 1101101 if it is known only one digit has an error.

A set of words composed of 0's and 1's that has a message and parity check sums appended to the message is called a ***binary linear code***. The resulting strings are called ***code words***.

The process of determining the message you were sent is called ***decoding***. If you are sent a message x and receive the message as y , how can it be decoded?

The ***distance between two strings*** of equal length is the number of positions in which the strings differ.

- | | |
|---------------------------------------------|-----------------------------------------------|
| (a) 10101 and
11101
distance of _____ | (b) 111111 and
000000
distance of _____ |
|---------------------------------------------|-----------------------------------------------|

The ***nearest neighbor decoding method*** decodes a message as the code word that agrees with the message in the most positions *provided there is only one such message*.

How good a code is at detecting and correcting errors is determined by the weight of the code. The ***weight of a binary code*** is the minimum number of 1's that occur among all non-zero code words of that code.

Consider a code of weight t ,

- The code can *detect* $t - 1$ or fewer errors.
- The code can *correct* half as many as it can detect (rounded down).
 - $\frac{t-1}{2}$ or fewer errors if t is odd.
 - $\frac{t-2}{2}$ or fewer errors if t is even.

Consider the code $C = \{0000000, 0001111, 1111000, 1111111\}$

- (a) What is the weight of the code?
- (b) How many errors can this code detect?
- (c) How many errors can this code correct?
- (d) Decode the message received as 0001101.

A **compression algorithm** converts data from an easy-to-use format to one that is more compact. jpg photo files use data compression as do most video and audio files.

Delta function encoding uses the beginning value and the differences in one value to the next to encode the data.

The data below is the closing price of the Dow Jones on Oct. 1, 2012 – Oct 5, 2012. Compress the data using delta function encoding and determine how much the data is compressed.

13610 13575 13495 13482 13515

Binary codes can also be compressed by assigning short codes to characters that occur frequently and longer codes to characters that occur rarely.

We have 5 symbols, A, B, C, D, and E. If we give all the symbols a code of the same length, we would need 3 binary digits (000 to 101). So a string of 6 symbols would be $6 \times 3 = 18$ characters long. Can we devise a different binary code if we knew how often each character occurred?

Use *Huffman coding* is a way to assign shorter code words to those characters that occur more often.

Step 1 Arrange these letters from least to most likely.

Step 2 Add the probabilities of the two least likely characters and combine them. Keep the letter with the smaller probability on the left. Arrange the new list from least to most likely.

Step 3 Repeat Step 2 until all the letters have been combined into one group with probability of 1.

Step 4 To assign a binary code to each letter, display the information in a Huffman tree by undoing the process from Steps 2 and 3. Always keep the smaller probability on the left and assign a 0 to that branch. Assign a 1 to the branch with the higher probability.

Step 5 The 0's and 1's for each path determine the code word for that letter. Read from the top of the chart down to the letter.

A	B	C	D	E
0.16	0.19	0.23	0.17	0.25

A	B	C	D	E

Decipher a message that was encoded using this Huffman code:

1011001001111101010010001

The process of disguising data is called encryption. Cryptology is the study of making and breaking secret codes.

A *Caesar cipher* shifts the letters of the alphabet by fixed amount.

EXAMPLE

Create a Caesar cipher that shifts the alphabet by 10 letters and use it to encrypt the message THANKS.

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	

The message HSZSDVCBS was created with a Caesar cipher with a shift of 14. What is the original message?

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	

A *decimation cipher* multiplies the position of each letter by a fixed number k (called the *key*) and then uses modular arithmetic. To use a decimation cipher,

1. Assign the letters A – Z to the numbers 0 – 25.
2. Choose a value for the key, k , that is an odd integer from 3 to 25 but not 13 (why not?)
3. Multiply the value of each letter (i) by the key (k) and find the remainder when divided by 26.
4. To decrypt a message, the encrypted value x needs to be multiplied by the decryption letter j and then the remainder mod 26 is the original letter.

A *Vigenère cipher* uses a *key word* to encode the characters.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z

Use a *Vigenère cipher* with a key word of MINT to encode the message

N E W P R I N T E R

Position																									
Key Word																									
Sum																									
Mod 26																									
Code																									

A *Vigenère cipher* with a key word of LEX was used to encode the message below. Decode it. You will need to subtract rather than add.

D Y M P V J L R

Position																									
Key Word																									
Diff.																									
Mod 26																									
Code																									

To increase security, binary strings can be added together. If the result is even, enter 0. If the result is odd, enter 1. In other words, take the sum of the digits mod 2. *Note:* This is not the same as binary addition.

Add the binary strings:

(a)
$$\begin{array}{r} 10110 \\ + 00111 \\ \hline \end{array}$$

(b)
$$\begin{array}{r} 10110 \\ + 10110 \\ \hline \end{array}$$

SAMPLE EXAM QUESTIONS FROM CHAPTER 17

1. Convert the binary number 11001 to a decimal number.
(A) 3 (B) 25
(C) 6 (D) 31

2. What is the distance between received words 1100101 and 1010111?
(A) 1 (B) 2 (C) 3 (D) 4
(E) more than 4

3. Add the binary strings 1100101 and 1110001. How many 1s digits are in the sum?
(A) 1 (B) 2 (C) 3 (D) 4
(E) more than 4

4. Use delta encoding to compress the data
1834 1831 1831 1825 1850.
By how many characters is the data compressed?
(A) 9 (B) 10
(C) 11 (D) 13

5. Use the Venn diagram method to decode the received word 1101101.
(A) 1001
(B) 0100
(C) 1101
(D) 1011
(E) None of these

Questions 6 and 7 use the code $\{1100, 1010, 1001, 0110, 0101, 0011\}$.

6. What is the weight of this code?

- (A) 0 (B) 1 (C) 2 (D) 3 (E) 4

7. Which one of the following is a true statement about this code?

- (A) This code can detect and correct two errors
(B) This code can detect two errors and correct 1 error
(C) This code can detect and correct one error.
(D) This code can detect one error and correct 0 errors
(E) None of these

8. Given binary codes $A \rightarrow 0, C \rightarrow 10, I \rightarrow 110, S \rightarrow 1110, B \rightarrow 11110$.

(a) Encode the message CASSI

(b) Decode the message 1111001110110101110

9. What is the code word for the message 110, if the code word is the message appended with three check digits found using the parity-check sums $a_1 + a_2 + a_3$, $a_1 + a_3$ and $a_2 + a_3$?

10. Use a Huffman code to assign binary codes to the directions that occur with the probabilities given below.

Up	Down	Left	Right
0.12	0.20	0.25	0.43

11. Use a Caesar cipher with a shift of 3 to encode the word **BINARY**.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z

12. Use a decimation cipher with key 9 to encode the word **CABLE**.

C A B L E

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z

13. Use the Vigenere cipher with the key word **PEN** to encode **BASEBALL**.

B A S E B A L L
