Chapter 17
Information Science

Chapter Outline

Chapter Summary

Data that are stored or transmitted can be corrupted in a variety of ways. Mathematics can be employed to detect and correct the resulting errors. Mathematical methods are also useful in data compression (which helps reduce transmission time and storage space) and *cryptography* (which ensures secure transmission of data by protecting it from "eavesdroppers").

The circuitry of digital computers makes the binary system the most convenient system to use for representing data. Mathematics (particularly abstract algebra) provides us with methods for systematically introducing redundancy into our data so that we can detect and in certain instances, correct errors that occur during storage or transmission. Ideally, a code will correct up to some fixed number of errors in a code word, regardless of where those errors occur.

Codes are constructed in different ways. The *binary linear codes* use code words consisting of strings of n 0's and 1's. Typically, the first k digits are the "message part" and encode a piece of our data (perhaps a single symbol). The remaining $n-k$ digits are *parity-check digits*. The value of each check digit is chosen so that the sum of certain components of the code word is even (a parity-check sum). The set of check digits in a code word is uniquely determined by the message part and represents the redundancy in the message. The check digits ensure that the number of code words ($2k$) is but a small fraction of the total number of n-tuples of 0's and 1's ($2n$). By carefully choosing the parity-check sums, we can further guarantee that the distance between any pair of code words (the number of components in which they disagree) is never less than a certain value t.

So constructed, our binary linear code can detect up to $t-1$ errors and correct up to $\frac{t-1}{2}$ errors (if t is odd) or $\frac{t-2}{2}$ errors (if t is even), although it may not be able to do both simultaneously. The decoding of messages is done by the *nearest-neighbor rule*, as follows: if we receive word v, we decode it as the code word w lying closest to the word v. The word w will be unique if the number of errors that occurred during transmission does not exceed the correcting capability of the code. Otherwise, there may be several choices for w, or we may not be able to tell any errors occurred. If error correction is our goal, we should choose a code that will correct at least as many errors as are likely to occur. Error detection is clearly easier: if the word received is not a code word, then errors have occurred.

The binary linear codes are fixed length in that all code words have the same length. Some codes, such as the Morse code, use *variable-length code* words. The number of dots and dashes used for frequently occurring letters is small and is somewhat larger for less frequently occurring ones. On average, this allows more information to be transmitted with a given number of symbols than would be possible with a fixed-length code. Variable-length codes are therefore useful in *compressing data* for speedier transmission or more efficient storage.

A form of coding in 1951 was developed by David Huffman. *Huffman coding* assigns short code words to characters with higher probabilities of occurring. Even with this relatively recent development of coding, the need to transmit secure messages has a long history. One of the earliest codes (although hardly a secure one) is attributed to Julius Caesar. The *Caesar cipher* involves a substitution of letters based on a shift of the alphabet. The *Vignère cipher* involves choosing a key word and applying modular arithmetic. In the 1970s, a method known as public key cryptography was discovered by Rivest, Shamir, and Adelman. Utilizing modular arithmetic, the security of the system is due to the difficulty in factoring very large numbers. The method of digital signatures is an important application of public key cryptography.

Finally, in this age of Internet usage, this chapter addresses how *Boolean logic* is used to make search engines more efficient. Connectives and the construction of *truth tables* are examined.

Skill Objectives

1. Know what a binary code is.

2. Use the diagram method to determine or verify a code word, given the message.

3. Use the diagram method to decode a received message.

4. Be able to compute check digits for code words given the parity-check sums for the code.

3. Be able to determine the distance between two n-tuples of 0's and 1's.

4. Be able to determine the weight of a code word and the minimum weight of the nonzero code words in a code (for binary codes, the minimum weight is the same as the minimum distance between code words).

5. Know what nearest-neighbor decoding is and be able to use it for decoding messages received in the Hamming code of Table 17.2.

6. Be able to encode and decode messages that have symbols (such as letters of the alphabet) expressed in binary form.

7. Be able to make observations regarding frequently (or infrequently) occurring letters.

8. Be able to decode using a Huffman code and be able to create a Huffman code given a table of probabilities.

9. Be able to encode and decode messages using the Caesar and Vigenère ciphers.

10. Be able to add binary strings.

11. Be able to perform calculations using modular arithmetic.

12. Understand how the RSA public key encryption scheme works.

13. Be able to complete a truth table given 2 or 3 statements and the connectives NOT, OR, and AND.

14. Be able to determine if two statements are logically equivalent.

Teaching Tips

1. The Hamming code listed in Table 17.2 can correct one error and detect two, but it can't do both simultaneously. This is not a universal defect. There are linear codes that can do both at the same time (we need t to be 4 rather than 3; add a fourth check digit that is an overall parity check to the code of Table 17.2).

2. Nearest-neighbor decoding is another example of enlisting the aid of probability. We decode to the nearest code word because a smaller number of errors is more likely than a larger one. (This type of decoding is also known as maximum-likelihood decoding.)

3. Stress geometric thinking. It can be an important intuitive aid.

4. The main set of ideas is that error detection is impossible if every n-tuple is a code word, yet it gets easier if the fraction of n-tuples that are code words is small. Finally, the ability to correct is enhanced if code words are kept far apart.

5. The idea of secret codes (cryptography) can be fascinating to students. The security of the RSA public-key cryptosystem is based on the computational difficulty of finding prime factors of very large integers. This topic provides a good excuse for discussing primes and the fundamental theorem of arithmetic in a context that emphasizes a practical application of material students may have found esoteric or boring (or both).

6. Students may need to be reminded in various encoding/decoding methods that when the alphabet is correlated to a set of 26 positions, they can begin at 0 or 1, depending of the method.

Research Paper

Ask students to investigate the history of the Internet and search engines in general. Students may describe several different search engines and how they prepare the retrieved information. Also, students can investigate whether the results change (and by how much) if connectives are used or altered from one search to another (searching for the same type of information).

Collaborative Learning

Caesar Cipher

The Caesar cipher can serve as a good icebreaker for the subject of coding and decoding. Break the class up into pairs, and have each student construct a message of about 10 words, and then encode the message using the Caesar cipher, each choosing the size of the shift. Then have the students exchange coded messages and attempt to decode them.

Solutions

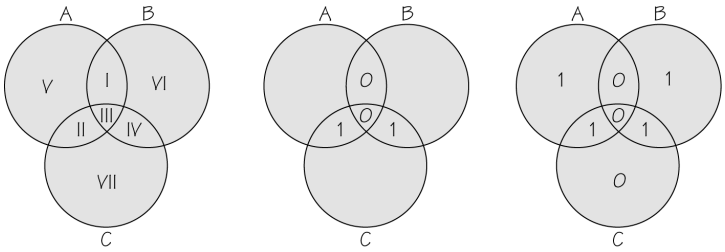
Skills Check:

1. b 2. b 3. b 4. c 5. a 6. b 7. a 8. c 9. b 10. b

11. a 12. b 13. c 14. a 15. c 16. b 17. c 18. c 19. a 20. a

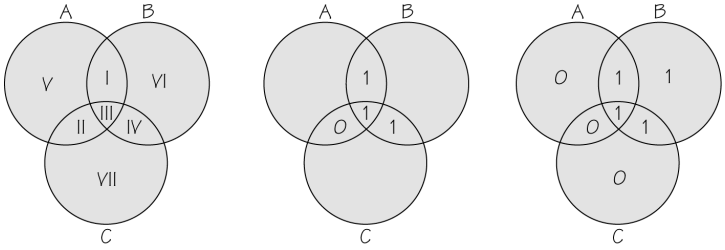
Exercises:

1. For 0101:



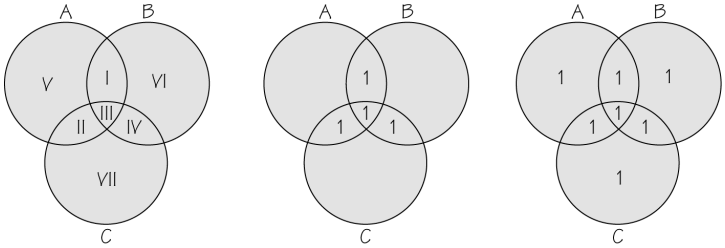
The code word given by regions I, II, III, IV, V, VI, VII is 0101110.

For 1011:



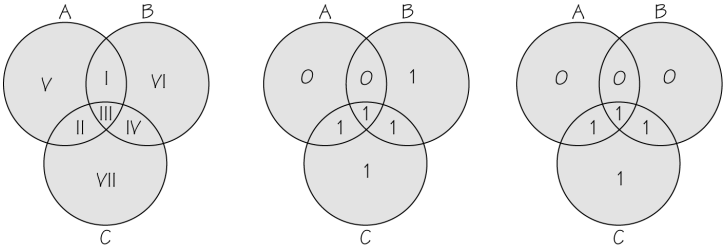
The code word given by regions I, II, III, IV, V, VI, VII is 1011010.

For 1111:



The code word given by regions I, II, III, IV, V, VI, VII is 1111111.

2. For 0111011:

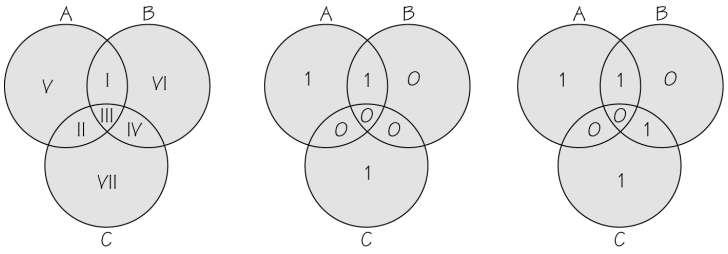


The decoded word given by regions I, II, III, IV, V, VI, VII is 0111001.

Continued on next page

2. continued

For 1000101:



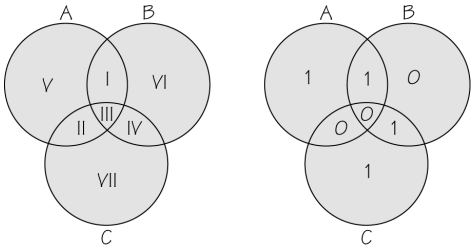
The decoded word given by regions I, II, III, IV, V, VI, VII is 1001101.

3. (a) When you compare 1**1011011** and 1**0100110**, they differ by 6 digits. Thus, the distance is 6.

(b) When you compare **0**11**10**100 and **1**11**01**100, they differ by 3 digits. Thus, the distance is 3.

4. For 0000110, 1000110 only differs by 1 digit. Thus, 0000110 is decoded as 1000110.
For 1110110, 1110100 only differs by 1 digit. Thus, 1110110 is decoded as 1110100.

5. There would be no change to 1001101 since the total number of 1's in each circle is even.



6. No errors were made in transmission or at least two errors were made in transmission.

7. Consider the following table.

	$a_2 + a_3$	c_1	$a_1 + a_3$	c_2	$a_1 + a_2$	c_3	Code word
000	0	**0**	0	**0**	0	**0**	**000000**
100	0	**0**	1	**1**	1	**1**	**100011**
010	1	**1**	0	**0**	1	**1**	**010101**
001	1	**1**	1	**1**	0	**0**	**001110**
110	1	**1**	1	**1**	2	**0**	**110110**
101	1	**1**	2	**0**	1	**1**	**101101**
011	2	**0**	1	**1**	1	**1**	**011011**
111	2	**0**	2	**0**	2	**0**	**111000**

Thus, the binary linear code is 000000, 100011, 010101, 001110, 110110, 101101, 011011, 111000.

8. Since the weight of the code is 4, it will correct any single error or detect any 3 errors.

9. Consider the following table.

	$a_2+a_3+a_4$	c_1	a_2+a_4	c_2	$a_1+a_2+a_3$	c_3	Code word
0000	0	**0**	0	**0**	0	**0**	**0000000**
1000	0	**0**	0	**0**	1	**1**	**1000001**
0100	1	**1**	1	**1**	1	**1**	**0100111**
0010	1	**1**	0	**0**	1	**1**	**0010101**
0001	1	**1**	1	**1**	0	**0**	**0001110**
1100	1	**1**	1	**1**	2	**0**	**1100110**
1010	1	**1**	0	**0**	2	**0**	**1010100**
1001	1	**1**	1	**1**	1	**1**	**1001111**
0110	2	**0**	1	**1**	2	**0**	**0110010**
0101	2	**0**	2	**0**	1	**1**	**0101001**
0011	2	**0**	1	**1**	1	**1**	**0011011**
1110	2	**0**	1	**1**	3	**1**	**1110011**
1101	2	**0**	2	**0**	2	**0**	**1101000**
1011	2	**0**	1	**1**	2	**0**	**1011010**
0111	3	**1**	2	**0**	2	**0**	**0111100**
1111	3	**1**	2	**0**	3	**1**	**1111101**

Thus, the binary linear code is 0000000, 1000001, 0100111, 0010101, 0001110, 1100110, 1010100, 1001111, 0110010, 0101001, 0011011, 1110011, 1101000, 1011010, 0111100, 1111101. No, since 1000001 has weight 2.

10. The nearest neighbors of 11101 are 11100 and 11001, so we do not decode; 01100 is decoded as 11100. The intended code word for the received word 11101 cannot be determined since there are two code words that have distance 1 from it.

11. Consider the following table.

	a_1+a_2	c_1	a_2+a_3	c_2	a_1+a_3	c_3	Code word
000	0	**0**	0	**0**	0	**0**	**000000**
100	1	**1**	0	**0**	1	**1**	**100101**
010	1	**1**	1	**1**	0	**0**	**010110**
001	0	**0**	1	**1**	1	**1**	**001011**
110	2	**0**	1	**1**	1	**1**	**110011**
101	1	**1**	1	**1**	2	**0**	**101110**
011	1	**1**	2	**0**	1	**1**	**011101**
111	2	**0**	2	**0**	2	**0**	**111000**

Thus, the binary linear code is 000000, 100101, 010110, 001011, 110011, 101110, 011101, 111000. 001001 is decoded as 001011; 011000 is decoded as 111000; 000110 is decoded as 010110; 100001 is decoded as 100101.

12. Consider the following table.

	Weight	Append	Code word
0000000	0	**0**	**00000000**
0001011	3	**1**	**00010111**
0010111	4	**0**	**00101110**
0100101	3	**1**	**01001011**
1000110	3	**1**	**10001101**
1100011	4	**0**	**11000110**
1010001	3	**1**	**10100011**
1001101	4	**0**	**10011010**
0110010	3	**1**	**01100101**
0101110	4	**0**	**01011100**
0011100	3	**1**	**00111001**
1110100	4	**0**	**11101000**
1101000	3	**1**	**11010001**
1011010	4	**0**	**10110100**
0111001	4	**0**	**01110010**
1111111	7	**1**	**11111111**

The extended code is 00000000, 00010111, 00101110, 01001011, 10001101, 11000110, 10100011, 10011010, 01100101, 01011100, 00111001, 11101000, 11010001, 10110100, 01110010, 11111111. The code will detect any three errors or correct any single error.

13. Consider the following table.

	Weight	Append	Code word
0000000	0	**0**	**00000000**
0001011	3	**1**	**00010111**
0010111	4	**0**	**00101110**
0100101	3	**1**	**01001011**
1000110	3	**1**	**10001101**
1100011	4	**0**	**11000110**
1010001	3	**1**	**10100011**
1001101	4	**0**	**10011010**
0110010	3	**1**	**01100101**
0101110	4	**0**	**01011100**
0011100	3	**1**	**00111001**
1110100	4	**0**	**11101000**
1101000	3	**1**	**11010001**
1011010	4	**0**	**10110100**
0111001	4	**0**	**01110010**
1111111	7	**1**	**11111111**

The extended code is 00000000, 00010111, 00101110, 01001011, 10001101, 11000110, 10100011, 10011010, 01100101, 01011100, 00111001, 11101000, 11010001, 10110100, 01110010, 11111111. The code will detect any three errors or correct any single error.

14. It will correct any two errors or detect any five errors.

15. There are $2^5 = 32$ possible messages of length 5. There are $2^8 = 256$ possible received words.

16. Such a binary linear code has 8 code words including 000000. For such a code to correct all double errors, the weight of the code would have to be 5. This means that each of the 7 nonzero code words must have at most one 0. Thus, the code would have to be as follows.

$$\{000000,\ 011111,\ 101111,\ 110111,\ 111011,111101,\ 111110,\ 111111\}$$

One mistake in 111111 could not be detected.

17. Consider the following table.

	$a_1 + a_2$	$c_1 = (a_1 + a_2) \bmod 3$	$2a_1 + a_2$	$c_2 = (2a_1 + a_2) \bmod 3$	Code word
00	0	**0**	0	**0**	**0000**
10	1	**1**	2	**2**	**1012**
20	2	**2**	4	**1**	**2021**
01	1	**1**	1	**1**	**0111**
02	2	**2**	2	**2**	**0222**
11	2	**2**	3	**0**	**1120**
22	4	**1**	6	**0**	**2210**
21	3	**0**	5	**2**	**2102**
12	3	**0**	4	**1**	**1201**

Thus, the ternary code is 0000, 1012, 2021, 0111, 0222, 1120, 2210, 2102, 1201.

18. 1211 would be decoded as 1201.

Code word	1211 differs by
0000	4 digits
1012	2 digits
2021	3 digits
0111	2 digits
0222	3 digits
1120	3 digits
2210	2 digits
2102	4 digits
1201	**1 digit**

19. There are $3^4 = 81$ possible messages of length 5. There are $3^6 = 729$ possible received words.

20. 0, 10, 0, 0, 111, 110, 0, 0, 1, 0 would be written as 010001111100010.

21. 001100001111000 can be written as 0, 0, 110, 0, 0, 0, 111, 10, 0, 0. Thus we have, AATAAAGCAA.

22. 0, 1111, 0, 0, 1110, 10, 0, 0, 10, 110, 10 would be written as 01111001111010001011010. 01000110100011111110 can be written as 0, 10, 0, 0, 110, 10, 0, 0, 1111, 1110. Thus we have, *ABAACBAAED*.

23. 1111, 0, 10, 0, 0, 1110, 0, 10, 10 would be written as 111101000111001010. 001000110011110111010 can be written as 0, 0, 10, 0, 0, 110, 0, 1111, 0, 1110, 10. Thus we have *AABAACAEADB*.

24. $A \to 0,\ B \to 10,\ C \to 110,\ D \to 1110,\ E \to 11110,\ F \to 111111.$

25. *t*, *n*, and *r* would be the most frequently occurring consonants. *e* would be the most frequently occurring vowel.

26. Many of the occurrences of *H* are due to the word "the" which is typically omitted in telegrams.

27. In the Morse code, a space is needed to determine where each code word ends. In a fixed-length code of length k, a word ends after each k digits.

28. Any guess between 15% and 20% is an excellent guess. The approximate value is 18.6%.

29. Given the following:

2015	2015
2057 – 2015	42
2079 – 2057	22
2060 – 2079	–19
2050 – 2060	–10
2053 – 2050	3
2065 – 2053	12
2030 – 2065	–35
2025 – 2030	–5
2002 – 2025	–23

the compressed numbers are 2015 42 22 –19 –10 3 12 – 35 – 5 – 23; We go from 49 characters to 34 characters, a reduction of $\frac{49-34}{49} = \frac{15}{49} \approx 30.6\%$.

30. Given the following:

1207	1207
1207 + 373	1580
1580 – 57	1523
1523 – 97	1426
1426 – 234	1192
1192 – 105	1087
1087 + 178	1265
1265 – 73	1192
1192 + 275	1467
1467 + 79	1546
1546 – 183	1363
1363 – 146	1217
1217 – 94	1123
1123 + 129	1252

the original numbers are:

1207 1580 1523 1426 1192 1087 1265
1192 1467 1546 1363 1217 1123 1252.

31. 11100100001001110110011010 can be written as 1110, 01, 00, 00, 10, 01, 110, 110, 01, 10, 10. Thus, the message is decoded to be *BCEEFCDDCFF*.

32. Arranging these is order we have the following.

C	0.015
A	0.025
B	0.150
D	0.170
E	0.200
G	0.215
F	0.225

Since *C* can *A* are the least likely to occur, we begin the tree by merging them.

CA	0.040
B	0.150
D	0.170
E	0.200
G	0.215
F	0.225

Now *CA* and *B* are least likely, so we merge them.

D	0.170
CAB	0.190
E	0.200
G	0.215
F	0.225

Now *D* and *CAB* are least likely, so we merge them.

E	0.200
G	0.215
F	0.225
DCAB	0.360

Now *E* and *G* are least likely, so we merge them.

F	0.225
DCAB	0.360
EG	0.415

Now *F* and *DCAB* are least likely, so we merge them.

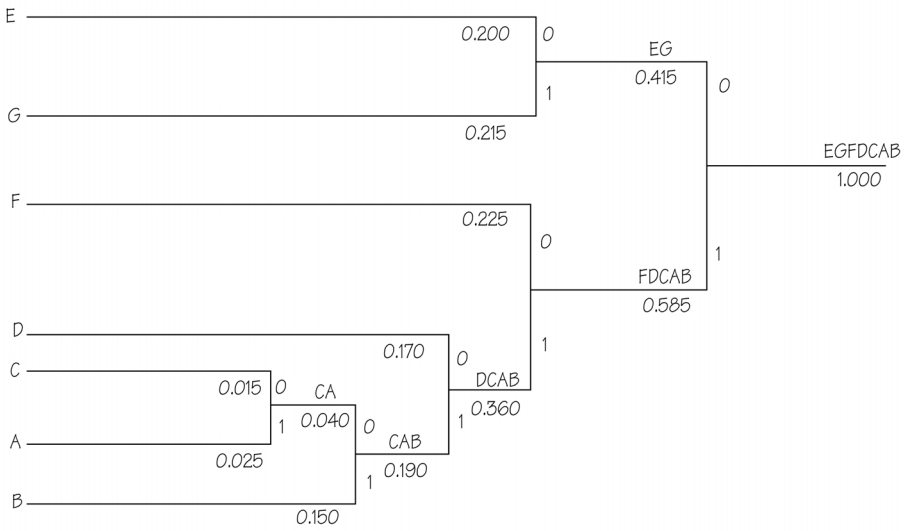
EG	0.415
FDCAB	0.585

And finally,

EGFDCAB	1.000

Continued on next page

32. continued
Working backwards we have the following diagram.



Thus, we have $A = 11101$, $B = 1111$, $C = 11100$, $D = 110$, $E = 00$, $F = 10$, $G = 01$.

33. *B* is the least likely letter, *J* is the second least likely, and *G* is the third least likely letter.

34. (a) $16 \bmod 7 = 2$, so add 2 days onto Wednesday.
(b) $37 \bmod 12 = 1$, so add 1 hour to 4 o'clock.
(c) $37 \bmod 24 = 13$, so add 1300 to 0400.
(d) July and August have 31 days and $65 \bmod 31 = 3$, so add 3 days to 20.
(e) An odometer uses mod 100000, so $(97000 + 12000) \bmod 100000 = 9000$.

35. RETREAT would be encrypted as UHWUHDW. DGYDQFH would be decoded as ADVANCE.

36. It would take 26 iterations.

37. One can take a position, such as 0 and determine that it takes 13 iterations to arrive at 0 again.

$$(0+8) \bmod 26 = 8 \bmod 26 = 8$$
$$(8+8) \bmod 26 = 16 \bmod 26 = 16$$
$$(16+8) \bmod 26 = 24 \bmod 26 = 24$$
$$(24+8) \bmod 26 = 32 \bmod 26 = 6$$
$$(6+8) \bmod 26 = 14 \bmod 26 = 14$$
$$(14+8) \bmod 26 = 22 \bmod 26 = 22$$
$$(22+8) \bmod 26 = 30 \bmod 26 = 4$$
$$(4+8) \bmod 26 = 12 \bmod 26 = 12$$
$$(12+8) \bmod 26 = 20 \bmod 26 = 20$$
$$(20+8) \bmod 26 = 28 \bmod 26 = 2$$
$$(2+8) \bmod 26 = 10 \bmod 26 = 10$$
$$(10+8) \bmod 26 = 18 \bmod 26 = 18$$
$$(18+8) \bmod 26 = 26 \bmod 26 = 0$$

38. H is in the position 7; E is in position 4; L is in position 11; P is in position 15.

Original	Location			Encrypted
P	15	7	$(15+7) \bmod 26 = 22 \bmod 26 = 22$	W
H	7	4	$(7+4) \bmod 26 = 11 \bmod 26 = 11$	L
O	14	11	$(14+11) \bmod 26 = 25 \bmod 26 = 25$	Z
N	13	15	$(13+15) \bmod 26 = 28 \bmod 26 = 2$	C
E	4	7	$(4+7) \bmod 26 = 11 \bmod 26 = 11$	L
H	7	4	$(7+4) \bmod 26 = 11 \bmod 26 = 11$	L
O	14	11	$(14+11) \bmod 26 = 25 \bmod 26 = 25$	Z
M	12	15	$(12+15) \bmod 26 = 27 \bmod 26 = 1$	B
E	4	7	$(4+7) \bmod 26 = 11 \bmod 26 = 11$	L

The encrypted message would be WLZCL LZBL.

39. Given the key word BEATLES, we note that B is in the position 1; E is in position 4; A is in position 0; T is in position 19; L is in position 11; S is in position 18.

Encrypted	Location				Decrypted
S	18	1	$18 = 18 \bmod 26 = (17+1) \bmod 26$	17	R
S	18	4	$18 = 18 \bmod 26 = (14+4) \bmod 26$	14	O
L	11	0	$11 = 11 \bmod 26 = (11+0) \bmod 26$	11	L
E	4	19	$4 = 30 \bmod 26 = (11+19) \bmod 26$	11	L
T	19	11	$19 = 19 \bmod 26 = (8+11) \bmod 26$	8	I
R	17	4	$17 = 17 \bmod 26 = (13+4) \bmod 26$	13	N
Y	24	18	$24 = 24 \bmod 26 = (6+18) \bmod 26$	6	G
T	19	1	$19 = 19 \bmod 26 = (18+1) \bmod 26$	18	S
X	23	4	$23 = 23 \bmod 26 = (19+4) \bmod 26$	19	T
O	14	0	$14 = 14 \bmod 26 = (14+0) \bmod 26$	14	O
G	6	19	$6 = 32 \bmod 26 = (13+19) \bmod 26$	13	N
P	15	11	$15 = 15 \bmod 26 = (4+11) \bmod 26$	4	E
W	22	4	$22 = 22 \bmod 26 = (18+4) \bmod 26$	18	S

The original message was ROLLING STONES.

40. C is in the position 2; L is in position 11; U is in position 20; E is in position 4.

Original	Location			Encrypted
T	19	2	$(19+2) \bmod 26 = 21 \bmod 26 = 21$	V
H	7	11	$(7+11) \bmod 26 = 18 \bmod 26 = 18$	S
E	4	20	$(4+20) \bmod 26 = 24 \bmod 26 = 24$	Y
W	22	4	$(22+4) \bmod 26 = 26 \bmod 26 = 0$	A
A	0	2	$(0+2) \bmod 26 = 2 \bmod 26 = 2$	C
L	11	11	$(11+11) \bmod 26 = 22 \bmod 26 = 22$	W
R	17	20	$(17+20) \bmod 26 = 37 \bmod 26 = 11$	L
U	20	4	$(20+4) \bmod 26 = 24 \bmod 26 = 24$	Y
S	18	2	$(18+2) \bmod 26 = 20 \bmod 26 = 20$	U
W	22	11	$(22+11) \bmod 26 = 33 \bmod 26 = 7$	H
A	0	20	$(0+20) \bmod 26 = 20 \bmod 26 = 20$	U
S	18	4	$(18+4) \bmod 26 = 22 \bmod 26 = 22$	W
P	15	2	$(15+2) \bmod 26 = 17 \bmod 26 = 17$	R
A	0	11	$(0+11) \bmod 26 = 11 \bmod 26 = 11$	L
U	20	20	$(20+20) \bmod 26 = 40 \bmod 26 = 14$	O
L	11	4	$(11+4) \bmod 26 = 15 \bmod 26 = 15$	P

The encrypted message would be VSY ACWLYU HUW RLOP.

41. I'll.

42. that

43. To start, X would be an encrypted A or I. Similarly, G would be an encrypted A or I. The original statement is "I am a man."

44. The letter *e* never is used.

45. could've or would've.

46. your and you.

47. and

48. But, in a larger sense, we can not dedicate – we cannot consecrate – we can not hallow – this ground. The brave men, living and dead, who struggled here, have consecrated it, far above our poor power to add or detract. The world will little note, nor long remember what we say here, but it can never forget what they did here. It is for us the living, rather, to be dedicated here to the unfinished work which they who fought here have thus far so nobly advanced. It is rather for us to be here dedicated to the great task remaining before us – that from these honored dead we take increased devotion to that cause for which they gave the last full measure of devotion – that we here highly resolve that these dead shall not have died in vain – that this nation, under God, shall have a new birth of freedom – and that government of the people, by the people, for the people, shall not perish from the earth.

49. Oh, I love your magazine. Especially the "Enrich Your Word Power" section. I think it's really, really, really good.

50. No person shall be held to answer for a capital, or otherwise infamous crime, unless on a presentment or indictment of a grand jury, except in cases arising in the land or naval forces, or in the militia, when in actual service in time of war or public danger; nor shall any person be subject for the same offense to be twice put in jeopardy of life or limb; nor shall be compelled in any criminal case to be a witness against himself, nor be deprived of life, liberty, or property, without due process of law; nor shall private property be taken for public use, without just compensation. (Amendment V of the U. S. Constitution.)

51. (a)
$$\begin{array}{r} 10111011 \\ +\ 01111011 \\ \hline 11000000 \end{array}$$

(b)
$$\begin{array}{r} 11101000 \\ +\ 01110001 \\ \hline 10011001 \end{array}$$

52. (a) In order to show that a set cannot be a binary code one needs to find any instance where two words sum to be a non-word. In this case, 0011 + 0111 = 0100 demonstrates that this is not a binary code. (There are other examples.)

(b) In order to show that a set cannot be a binary code one needs to find any instance where two words sum to be a non-word. In this case, 0010 + 0111 = 0101 demonstrates that this is not a binary code. (There are other examples).

(c) In order to show that a set is a binary code, one needs to show that all sums of words are in turn words.

+	0000	0110	1011	1101
0000	0000	0110	1011	1101
0110	0110	0000	1101	1011
1011	1011	1101	0000	0110
1101	1101	1011	0110	0000

Thus, (a) and (b) cannot be binary codes.

53. 1. VIP converts to 220916.

2. We will send 22, 09, and 16 individually.

3. Since $p = 5$ and $q = 17$, $n = pq = 5 \cdot 17 = 85$.
Since $\text{GCF}(22,85) = 1, \text{GCF}(9,85) = 1,$ and $\text{GCF}(16,85) = 1,$ we can proceed. (GCF stands for *greatest common factor*.) Thus, $M_1 = 22,$ $M_2 = 09,$ and $M_3 = 16.$

4. Since $R_i = M_i^{\,r} \bmod n$ and $r = 3,$ we have the following.

$$R_1 = 22^3 \bmod 85 = (2 \cdot 11)^3 \bmod 85 = \left[\left(2^3 \cdot 11\right) \cdot 11^2\right] \bmod 85$$
$$= \left[\left[\left(2^3 \cdot 11\right) \bmod 85\right]\left[11^2 \bmod 85\right]\right] \bmod 85$$
$$= \left[(88 \bmod 85)(121 \bmod 85)\right] \bmod 85 = (3 \cdot 36) \bmod 85 = 108 \bmod 85 = 23$$

$$R_2 = 9^3 \bmod 85 = \left(3^2\right)^3 \bmod 85 = 3^6 \bmod 85 = \left[\left(3^5 \bmod 85\right)(3 \bmod 85)\right] \bmod 85$$
$$= \left[(243 \bmod 85)(3 \bmod 85)\right] \bmod 85 = (73 \cdot 3) \bmod 85 = 219 \bmod 85 = 49$$

$$R_3 = 16^3 \bmod 85 = \left(2^4\right)^3 \bmod 85 = 2^{12} \bmod 85 = \left[\left(2^7 \bmod 85\right)\left(2^5 \bmod 85\right)\right] \bmod 85$$
$$= \left[(128 \bmod 85)(32 \bmod 85)\right] \bmod 85 = (43 \cdot 32) \bmod 85 = (43 \cdot 2 \cdot 16) \bmod 85$$
$$= \left[(43 \cdot 2) \cdot 16\right] \bmod 85 = \left[(86 \bmod 85)(16 \bmod 85)\right] \bmod 85$$
$$= \left[1 \cdot (16 \bmod 85)\right] \bmod 85 = (16 \bmod 85) = 16$$

Thus, the numbers sent are 23, 49, 16.

54. 1. Since $p = 5$ and $q = 17$, $n = pq = 5 \cdot 17 = 85$.

2. Since $p - 1 = 5 - 1 = 4$ and $q - 1 = 17 - 1 = 16$, m will be the least common multiple of 4 and 16, namely 16.

3. We need to choose r such that it has no common divisors with 16. Thus, r can be 3. This confirms that $r = 3$ is a valid choice.

4. We need to find s such that $rs = 1 \bmod m$.

$$r^2 \bmod m = 3^2 \bmod 16 = 9 \bmod 16 = 9$$
$$r^3 \bmod m = 3^3 \bmod 16 = 27 \bmod 16 = (1 \cdot 16 + 11) \bmod 16 = 11$$
$$r^4 \bmod m = 3^4 \bmod 16 = 81 \bmod 16 = (5 \cdot 16 + 1) \bmod 16 = 1$$

Thus $t = 4$.

Since $s = r^{t-1} \bmod m$, where $r = 3$, $m = 16$, and $t = 4$, we have the following.

$$s = 3^{4-1} \bmod 16 = 3^3 \bmod 16 = 27 \bmod 16 = 11$$

5. Since $52^{11} \bmod 85 = \left[\left(52^4\right)^2 \cdot 52^3 \right] \bmod 85 = \left[\left(\left(52^4\right) \bmod 85 \right)^2 \cdot 52^3 \bmod 85 \right] \bmod 85$

$$= \left(1^2 \cdot 18\right) \bmod 85 = 18 \bmod 85 = 18,\ R_1 = 18.$$

Since $72^{11} \bmod 85 = \left[\left(72^4\right)^2 \cdot 72^3 \right] \bmod 85 = \left[\left(\left(72^4\right) \bmod 85 \right)^2 \cdot 72^3 \bmod 85 \right] \bmod 85$

$$= \left(1^2 \cdot 13\right) \bmod 85 = 13 \bmod 85 = 13,\ R_2 = 13.$$

Thus, the letters received are R and M.

55. 1. Since $p = 5$ and $q = 17$, $n = pq = 5 \cdot 17 = 85$.

2. Since $p - 1 = 5 - 1 = 4$ and $q - 1 = 17 - 1 = 16$, m will be the least common multiple of 4 and 16, namely 16.

3. We need to choose r such that it has no common divisors with 16. Thus, r can be 5. This confirms that $r = 5$ is a valid choice.

4. We need to find s such that $rs = 1 \bmod m$.

$$r^2 \bmod m = 5^2 \bmod 16 = 25 \bmod 16 = 9$$
$$r^3 \bmod m = 5^3 \bmod 16 = 125 \bmod 16 = 13$$
$$r^4 \bmod m = 5^4 \bmod 16 = 625 \bmod 16 = 1$$

Thus $t = 4$.

Since $s = r^{t-1} \bmod m$, where $r = 5$ and $t = 4$, we have the following.

$$s = 5^{4-1} \bmod 16 = 5^3 \bmod 16 = 125 \bmod 16 = 13$$

56. Because 3 has a divisor other than 1 in common with the least common multiple of 6 and 10 (which is 30).

57. N converts to 14 and O converts to 15, but 14 and 77 have a greatest common divisor of 7. On the other hand, using blocks of length 4, NO converts to 1415 and the greatest common divisor of 77 and 1415 is 1.

58. $13^9 \bmod 77 = 6$, $13^6 \bmod 77 = 64$. Google yields $13^{15} \bmod 77 = 0$, so we can calculate $13^{15} \bmod 77$ by doing $(13^9) \bmod 77 \times \left(13^6\right) \bmod 77 = \left(6 \times 64\right) \bmod 77 = 76$.

59. As the following shows, since the entries in the column for the variable P are exactly the same as the entries in the column for $P \vee (P \wedge Q)$, the two expressions are logically equivalent.

P	Q	$P \wedge Q$	$P \vee (P \wedge Q)$
T	T	T	T
T	F	F	T
F	T	F	F
F	F	F	F

60. First we construct the truth table for $\neg(P \vee Q)$.

P	Q	$P \vee Q$	$\neg(P \vee Q)$
T	T	T	F
T	F	T	F
F	T	T	F
F	F	F	T

Next we construct the truth table for $\neg P \wedge \neg Q$.

P	Q	$\neg P$	$\neg Q$	$\neg P \wedge \neg Q$
T	T	F	F	F
T	F	F	T	F
F	T	T	F	F
F	F	T	T	T

Since the last columns of the two truth tables are identical, the expression $\neg(P \vee Q)$ is logically equivalent to $\neg P \wedge \neg Q$.

61. First we construct the truth table for $\neg(P \wedge Q)$.

P	Q	$P \wedge Q$	$\neg(P \wedge Q)$
T	T	T	F
T	F	F	T
F	T	F	T
F	F	F	T

Next we construct the truth table for $\neg P \vee \neg Q$

P	Q	$\neg P$	$\neg Q$	$\neg P \vee \neg Q$
T	T	F	F	F
T	F	F	T	T
F	T	T	F	T
F	F	T	T	T

Since the last columns of the two truth tables are identical, we have shown that $\neg(P \wedge Q)$ is logically equivalent to $\neg P \vee \neg Q$.

62. First we construct the truth table for $P \vee (Q \wedge R)$.

P	Q	R	$Q \wedge R$	$P \vee (Q \wedge R)$
T	T	T	T	T
T	T	F	F	T
T	F	T	F	T
T	F	F	F	T
F	T	T	T	T
F	T	F	F	F
F	F	T	F	F
F	F	F	F	F

Next we construct the truth table for $(P \vee Q) \wedge (P \vee R)$.

P	Q	R	$P \vee Q$	$P \vee R$	$(P \vee Q) \wedge (P \vee R)$
T	T	T	T	T	T
T	T	F	T	T	T
T	F	T	T	T	T
T	F	F	T	T	T
F	T	T	T	T	T
F	T	F	T	F	F
F	F	T	F	T	F
F	F	F	F	F	F

Since the last columns of the two truth tables are identical, the expression $P \vee (Q \wedge R)$ is logically equivalent to $(P \vee Q) \wedge (P \vee R)$.

63. First we construct the truth table for $P \wedge (Q \vee R)$.

P	Q	R	$Q \vee R$	$P \wedge (Q \vee R)$
T	T	T	T	T
T	T	F	T	T
T	F	T	T	T
T	F	F	F	F
F	T	T	T	F
F	T	F	T	F
F	F	T	T	F
F	F	F	F	F

Continued on next page

63. continued

Next we construct the truth table for $(P \wedge Q) \vee (P \wedge R)$.

P	Q	R	$P \wedge Q$	$P \wedge R$	$(P \wedge Q) \vee (P \wedge R)$
T	T	T	T	T	T
T	T	F	T	F	T
T	F	T	F	T	T
T	F	F	F	F	F
F	T	T	F	F	F
F	T	F	F	F	F
F	F	T	F	F	F
F	F	F	F	F	F

Since the last columns of the two truth tables are identical, the expression $P \wedge (Q \vee R)$ is logically equivalent to $(P \wedge Q) \vee (P \wedge R)$.

64. Let P denote "lots of anchovies," let Q denote "spicy," and let R denote "large portion." Then the patron's order can be represented as the expression $(P \vee \neg Q) \wedge R$. The waiter's statement to the chef can be expressed as $(P \wedge R) \vee (Q \wedge R)$. A truth table for $(P \vee \neg Q) \wedge R$ is as follows.

P	Q	R	$\neg Q$	$P \vee \neg Q$	$(P \vee \neg Q) \wedge R$
T	T	T	F	T	T
T	T	F	F	T	F
T	F	T	T	T	T
T	F	F	T	T	F
F	T	T	F	F	F
F	T	F	F	F	F
F	F	T	T	T	T
F	F	F	T	T	F

A truth table for $(P \wedge R) \vee (Q \wedge R)$ is as follows.

P	Q	R	$P \wedge R$	$Q \wedge R$	$(P \wedge R) \vee (Q \wedge R)$
T	T	T	T	T	T
T	T	F	F	F	F
T	F	T	T	F	T
T	F	F	F	F	F
F	T	T	F	T	T
F	T	F	F	F	F
F	F	T	F	F	F
F	F	F	F	F	F

Because the last columns of these truth tables are not identical, we conclude that the two expressions are not logically equivalent. Thus, the waiter did not communicate the patron's wishes to the chef.

65. The truth table for $\neg P \vee Q$ is as follows.

P	Q	$\neg P$	$\neg P \vee Q$
T	T	F	T
T	F	F	F
F	T	T	T
F	F	T	T

Because the last column of this truth table is identical for the one for $P \to Q$, we conclude that the two expressions are logically equivalent.

66. Let P denote "the Vikings win," let Q denote "the Vikings make the playoffs". Then the coach's statement to the team is $P \to Q$ and the given conditions are P is F and Q is T. From the truth table, we have that $P \to Q$ is true.

67. Let P denote "it snows" and let Q denote "there is school." Then the statement "If it snows, there will be no school" can be expressed as $P \to \neg Q$. Similarly, the statement "it is not the case that it snows and there is school" can be expressed as $\neg(P \wedge Q)$. We now construct the truth tables for each of these expressions. A truth table for $P \to \neg Q$ is as follows.

P	Q	$\neg Q$	$P \to \neg Q$
T	T	F	F
T	F	T	T
F	T	F	T
F	F	T	T

A truth table for $\neg(P \wedge Q)$ is as follows.

P	Q	$P \wedge Q$	$\neg(P \wedge Q)$
T	T	T	F
T	F	F	T
F	T	F	T
F	F	F	T

Because the two tables have identical last columns, the two expressions are logically equivalent.

Word Search Solution