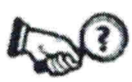
Lecture 8

The Puzzle-Mad Kidnapper (7.1 of Ecco): Baskerhound has kidnapped the son of a wealthy heiress and has sent her the following message.

"I am thinking of a number between 1 and 2000. If you can determine what that number is in 15 or fewer questions, I will release your son. Otherwise I will kill him. I will answer each question with a yes or a no. But beware, I may lie once. Also, I will answer your questions only after you have asked all of them."

Consider the following questions.



1. If Baskerhound does not lie, how can we solve the problem with 11 questions?
2. How can we extend our solution with 11 questions to a solution with 33 questions when Baskerhound may lie once?
3. How can we extend our solution with 11 questions to a solution with 23 questions when Baskerhound may lie once?
4. How can we solve the problem?

Counting in binary:

Binary Number System	Decimal Number System
1	1
10	2
11	3
100	4
101	5
110	6
111	7
$\vdots$	$\vdots$
$b_n b_{n-1} \cdots b_1 b_0$	$\sum_{i=0}^{n} b_i 2^i$

Changing from Binary to Decimal:

Given the binary number $b_n b_{n-1} \cdots b_0$ to convert to decimal we use:

$$\sum_{i=0}^{n} b_i \cdot 2^i$$

Example 1: Change $\overset{543210}{101010}$ from binary to decimal.

$$1 \cdot 2^5 + 0 \cdot 2^4 + 1 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0$$

$$= 32 + 8 + 2 = \boxed{42}$$

Changing from Decimal to Binary:

Given a decimal number n write it out as sum of powers of 2. To do this start from the largest power of 2 that does not exceed n and then work your way down through the powers of 2.

Example 2: Change 111 from decimal to binary.

$$111 = 64 + 47$$
$$= 64 + 32 + 15$$
$$= 64 + 32 + 8 + 7$$
$$= 64 + 32 + 8 + 4 + 2 + 1$$
$$\therefore \boxed{1 \quad 1 \quad 0 \quad 1 \quad 1 \quad 1 \quad 1}$$

$2^0 = 1$
$2^1 = 2$
$2^2 = 4$
$2^3 = 8$
$2^4 = 16$
$2^5 = 32$
$2^6 = 64$
$2^7 = 128$

Example 3: A solution to part 1 of the puzzle-mad kidnapper problem:

In binary is there a 1 in the first (rightmost) spot?

____________________ 2^{nd} __________?

⋮

____________________ 11^{th} __________?

Single error correcting codes

Definition 1: The *Triple Repetition Code* (published in 1950 by Alt) is a code which sends the same message 3 times. It can correct one corrupted symbol and detect up to two corrupted symbols.

Example 4: You have received the following codeword from the Triple Repetition Code, if necessary correct the codeword. Up to one symbol could be corrupted.

10101
10111
10101

The codeword that was sent : 10101 10101 10101 , the message is: 10101.

Definition 2: The efficiency of a code is given by the following calculation:

$$\text{efficiency} = \frac{\text{"codeword length"} - \text{"number of check digits"}}{\text{"codeword length"}} = \frac{\text{"message length"}}{\text{"codeword length"}}$$

Example 5: What is the efficiency of the triple repetition code?

$$\text{eff.} = \frac{15 - 10}{15} = \frac{5}{15}$$

Example 6: Ask the questions that solve part 2 of the puzzle-mad kidnapper problem.

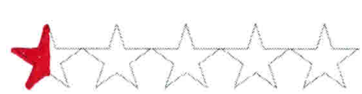
Repeat Ex 3, three times

Ranking the Triple Repetition Code

Efficiency: 5/15

Likeability: ☐ ☑ ☐ ☐ ☐

Overall Math Quality:



Definition 2: The *Rabenstein Code* (discovered by Mark Rabenstein in 1984 while in grade 8 at McKernan Jr. High in Edmonton) is a binary code which sends the same message 2 times and adds one more check digit at the end. The check digit is chosen so that there is an even number of 1's in both the message and the check digit. This code can correct one corrupted symbol and detect up to two corrupted symbols.

Example 7: You have received the following codewords from the Rabenstein Code if necessary correct the codewords. Up to one symbol could be corrupted.

4+0 4+0
1010101 | 1010101 | 0
✓ ✓ ✓
No corrupted symbols

4+1 5+1
1$\boxed{0}$11001 | 1111001 | 1
✗ ✓ ✓
∴ the message is: 1111001

2+0 3+0
0010001 | 0010$\boxed{1}$01 | 0
✓ ✗ ✓
∴ the message is: 0010001

3+0 3+0
1010001 | 1010001 | $\boxed{0}$
✓ ✓ ✗
∴ the message is: 1010001
(the check digit was corrupted)

Example 8: Ask the questions that solve part 3 of the puzzle-mad kidnapper problem.

Ex 3	Ex 3	23. In binary are there an odd # of 1's?

Ranking the Rabenstein Code

Efficiency: 7/15

Likeability: ☐ ☐ ☑ ☐ ☐

Overall Math Quality: ★★½☆☆

Definition 3: *Liu's Grid Code* (discovered in 1996 by a grade 6 student in Taipei, Taiwan) places the message bit in a 3 by 3 grid with an additional check digit at the end of each row and each column. The check digits are chosen so that each row and each column have an even number of 1's. This code can correct one corrupted symbol and detect up to two corrupted symbols.

Example 9: You have received the following codeword from Liu's Grid Code, if necessary correct the codeword. Up to one symbol could be corrupted.

$$\begin{array}{ccc|cc} 1 & 1 & 1 & 1\ \checkmark & 4 \\ 1 & 0 & 1 & 0\ \checkmark & 2 \\ 0 & 1 & 0 & 1\ \checkmark & 2 \\ \hline 0 & 0 & 0 & & \\ \checkmark & \checkmark & \checkmark & & \\ 2 & 2 & 2 & & \end{array}$$

No corrupted symbols

$$\begin{array}{ccc|cc} 1 & 0 & 0 & 1\ \checkmark & 2 \\ 0 & 1 & \boxed{0} & 0\ \times & 1 \\ 1 & 0 & 1 & 0\ \checkmark & 2 \\ \hline 0 & 1 & 0 & & \\ \checkmark & \checkmark & \times & & \\ 2 & 2 & 1 & & \end{array}$$

$\therefore$ the message is: 100 011 101

$$\begin{array}{ccc|cc} 1 & 1 & 1 & 1\ \checkmark & 4 \\ 0 & 1 & 1 & 0\ \checkmark & 2 \\ 0 & 0 & 1 & 1\ \checkmark & 2 \\ \hline 1 & 0 & \boxed{0} & & \\ \checkmark & \checkmark & \times & & \\ 2 & 2 & 3 & & \end{array}$$

$\therefore$ the message is: 111 011 001
(the check digit in the 3rd column was corrupted)

Ranking for *Liu's Grid Code*

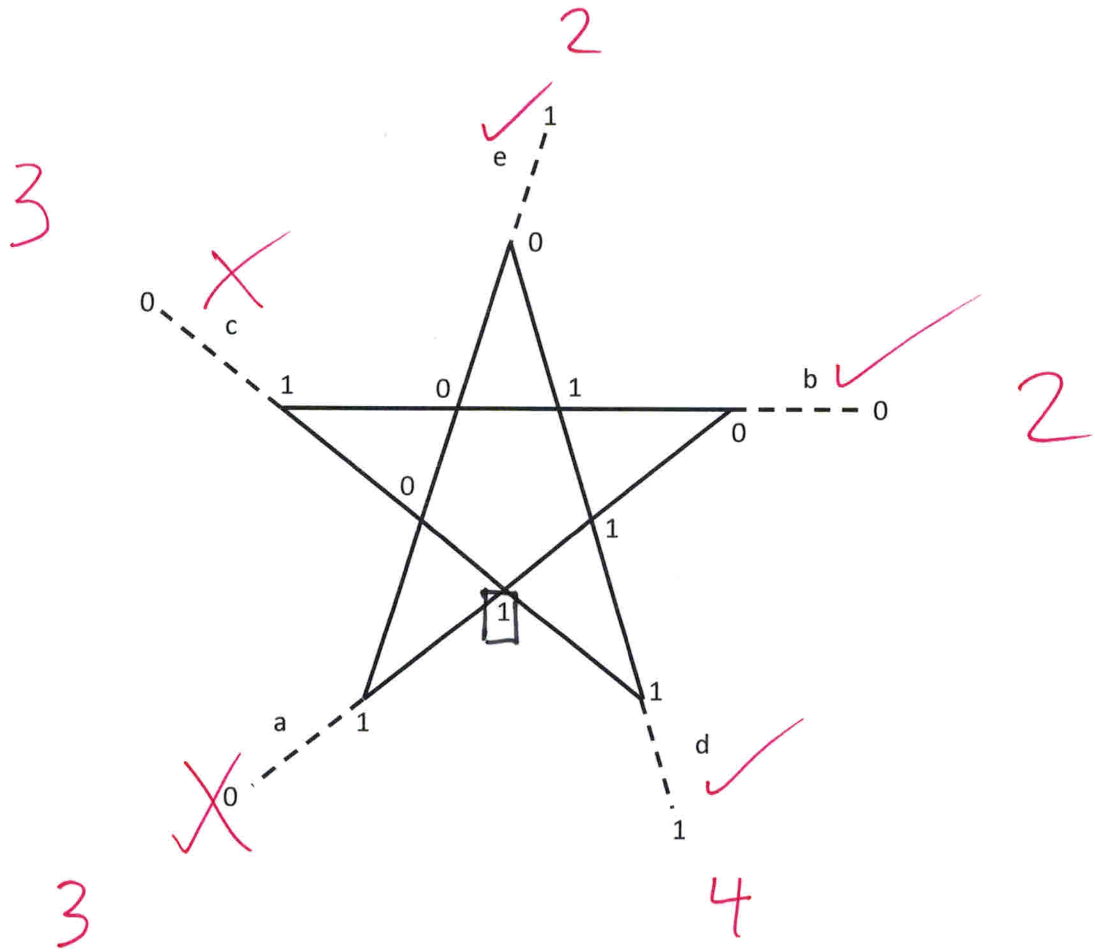
Efficiency: 9/15

Likeability: ☐ ☐ ☐ ☑ ☐

Overall Math Quality: ★★★☆☆

Definition 4: *Liu's Star Code* (discovered in 1997 by the same student in Taipei, Taiwan) places 10 message bits on the intersection points of a star made with 5 lines. Each of the five lines are extended beyond the star to make room for 5 check digits. The check digits are chosen so that each line in the star has an even number of 1's. This code can correct one corrupted symbol and detect up to two corrupted symbols.

Example 10: You have received the following codeword from Liu's Star Code, if necessary correct the codeword. Up to one symbol could be corrupted.



<u>Ranking for *Liu's Star Code*</u>

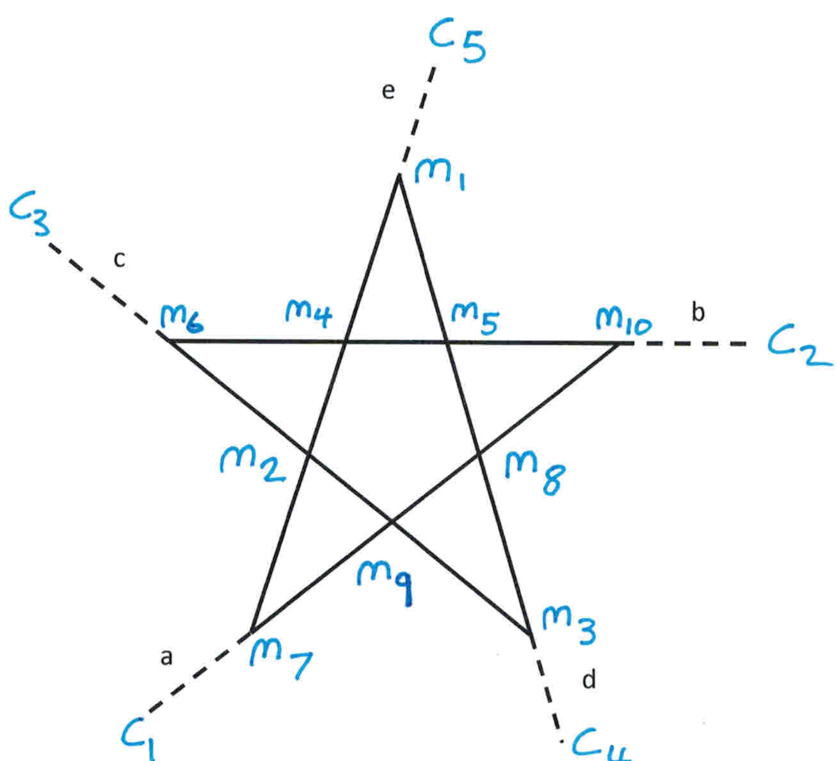
Efficiency: 10/15

Likeability: ☐ ☐ ☐ ☐ ☑

Overall Math Quality:

Example 11: Liu's Star Code can be represented in different way without using a star. In this representation the subsets of the set $\{a, b, c, d, e\}$ are used. Each subset of size two represents a message bit m_i while each subset of size one represents a check digit c_j. Find a correspondence between the two representations:

a	a	a	a							a				
b				b	b	b					b			
	c			c			c	c				c		
		d			d		d		d				d	
			e			e		e	e					e
m_{10}	m_9	m_8	m_7	m_6	m_5	m_4	m_3	m_2	m_1	c_1	c_2	c_3	c_4	c_5



The Hamming code:

- Is a binary code that has codewords of length $2^k - 1$ $(1,3,7,15\ldots)$, where k is the number of check digits.

- Assign each check digit a letter and write a different combination of these letters above each information digit. The check digits are chosen so there is an even number of 1's under every letter. For example:

a	a	a		a		
b	b		b		b	
c		c	c			c
0	1	0	0	**1**	**1**	**0**

a	a	a		a		
b	b		b		b	
c		c	c			c
1	1	0	0	0	0	1

a	a	a		a		
b	b		b		b	
c		c	c			c
0	0	0	1	0	1	1

a	a	a	a		a	a	a				a			
b	b	b		b	b			b	b			b		
c	c		c	c		c		c		c			c	
d		d	d	d			d		d	d				d
0	1	1	1	1	1	1	0	0	0	1	1	0	1	0

- The Hamming code can correct one corrupted digit. To do so find all letters that have an odd total of one's underneath them; afterwards switch the digit in the column containing these letters.

a	*a*	***a***		*a* ✗		
b	*b*		*b*		*b* ✓	
c		***c***	*c*			*c* ✗
0	1	$\boxed{1}$	0	1	1	0

a's – 3
b's – 2
c's – 1

Gets corrected to:

0 1 **0** 0 | 1 1 0

a	*a*	*a*		*a* ✓		
b	*b*		*b*		*b* ✓	
c		*c*	*c*			*c* ✗
1	1	0	0	0	0	$\boxed{0}$

a's – 2
b's – 2
c's – 1

Gets corrected to:

1 1 0 0 0 0 1

a	*a*	*a*		*a* ✗		
b	*b*		*b*		*b* ✗	
c		*c*	*c*			*c* ✗
$\boxed{1}$	0	0	1	0	1	1

a's – 1
b's – 3
c's – 3

Gets corrected to:

0 0 0 1 0 1 1

a	*a*	*a*	*a*		*a*	*a*	*a*				*a* ✓			
b	*b*	*b*		*b*	*b*			*b*	*b*			*b* ✓		
c	*c*		*c*	*c*		*c*		*c*		*c*			*c* ✗	
d		*d*	*d*	*d*			*d*		*d*	*d*				*d* ✗
0	1	1	1	1	1	1	0	0	0	$\boxed{0}$	1	0	1	0

a's – 6
b's – 4
c's – 5
d's – 3

Gets corrected to:

0 1 1 1 1 1 1 0 0 0 1 1 0 1 0

Ranking the Hamming code

Efficiency: 11/15

Likeability: ☐ ☐ ☐ ☑ ☐

Overall Math Quality: 

Example 12: Ask the questions that correspond to the hamming code:

position:	11^{th}	10^{th}	9^{th}	8^{th}	7^{th}	6^{th}	5^{th}	4^{th}	3^{rd}	2^{nd}	1^{st}	Check digits			
	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓	↓				
	a	a	a	a		a	a	a				a			
	b	b	b		b	b			b	b			b		
	c	c		c	c		c		c		c			c	
	d		d	d	d			d		d	d				d
	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
	11.	10.	9.	8.	7.	6.	5.	4.	3.	2.	1.	12.	13.	14.	15.

to solve part 4 of the puzzle-mad kidnapper problem.

- Ex 3
- 12. Is the an odd # of 1's in positions: $\boxed{4, 5, 6, 8, 9, 10, 11}$.

13. ________________________________ : $\boxed{2, 3, 6, 7, 9, 10, 11}$.

14. ________________________________ : $\boxed{1, 3, 5, 7, 8, 10, 11}$.

15. ________________________________ : $\boxed{1, 2, 4, 7, 8, 9, 11}$.