Computation Structures

Basics of Information Worksheet

Concept Inventory:

- Measuring information content; entropy
- Two's complement; modular arithmetic
- Variable-length encodings; Huffman's algorithm
- Hamming distance, error detection, error correction

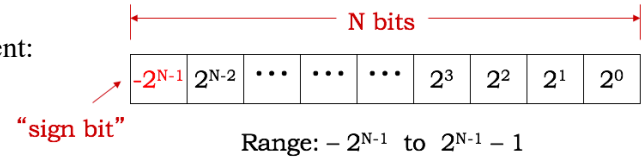
Notes:

Measuring information: $I(x_i) = \log_2(1/p_i)$ bits

N equally-probable choices down to M choices: $\log_2(N/M)$ bits

Entropy: $H(X) = E(I(X)) = \sum_i p_i \log_2(1/p_i)$

N-bit 2's complement:



Variable-length encoding:

Symbols with smallest p_i have longest encodings, symbols with largest p_i have shortest encodings.

Huffman's algorithm:

- Build binary decoding tree bottom-up starting with symbols that have smallest p_i.
- Each step: combine the two symbols or subtrees with smallest p_i into new subtree.

Hamming distance:

- HD = # of bit positions that differ between to codewords
- need to know min Hamming distance (HD_{min}) considering all pairs of codewords
- # of errors detected = $HD_{min} - 1$
- # of errors corrected = $\left\lfloor \frac{HD_{min}-1}{2} \right\rfloor$

1. Information Content and Entropy

A. You are given an unknown 3-bit binary number. You are then told that the binary representation contains exactly two 1's. How much information have you been given?

B. You are then given the **additional** information that the number is also odd. How much additional information have you been given?

C. A random variable X represents the outcome of flipping an unfair coin, where p(HEADS) = 0.6. Please give the value for the entropy H(X). You may express your answer as a numeric expression (i.e., you don't have to actually do the arithmetic).

D. A single decimal digit is chosen at random and you're told that its value mod 3 is 0. How much information have you learned about the digit?

E. X is an unknown 8-bit binary number. You are given another 8-bit binary number, 10101100, and told that the Hamming distance between X and 10101100 is one. How many bits of information about X have you been given? You can give a formula if you wish.

F. We wish to transmit messages comprised of the four symbols shown below with their associated probabilities and 5-bit fixed-length encoding.

Symbol	p(symbol)	encoding
α	0.5	00000
β	0.125	11100
γ	0.25	11011
δ	0.125	10111

An unknown symbol is received and you are told it's not δ. How much information have you received?

G. When transmitting a message comprised of these four symbols with the probabilities as given above, what is the expected amount information received when you are told the next symbol in the message?

H. You are given an unknown 5-bit binary number. You are then told that the first and last bits are the same. How much information have you been given?

I. I've randomly selected a letter from the alphabet and tell you that my selection is neither "X", "Y", nor "Z". How much information have I given you about my letter?

J. I make up a random 4-bit **two's complement** number by flipping a fair coin to determine each bit. You're trying to guess the number. If I tell you that the number is positive (> 0), how many bits of information have I given you? Be precise; you may answer by a formula or a number.

2. Two's Complement

A. What is the 6-bit two's complement representation of the decimal number -21?

B. What is the hexadecimal representation for decimal -51 encoded as an 8-bit two's complement number?

C. The hexadecimal representation for an 8-bit two's complement number is 0xD6. What is its decimal representation?

D. Since the start of official pitching statistics in 1988, the highest number of pitches in a single game has been 172. Assuming that remains the upper bound on pitch count, how many bits would we need to record the pitch count for each game as a *two's complement* binary number?

E. Can the value of the sum of two 2's complement numbers 0xB3 + 0x47 be represented using an 8-bit 2's complement representation? If so, what is the sum in hex? If not, write NO.

F. Can the value of the sum of two 2's complement numbers 0xB3 + 0xB1 be represented using an 8-bit 2's complement representation? If so, what is the sum in hex? If not, write NO.

G. Please compute the value of the expression 0xBB – 8 using 8-bit two's complement arithmetic and give the result in decimal (base 10).

H. What is the smallest (most negative) integer that can be represented as an 8-bit two's-complement integer? Give your answer as a decimal integer.

I. The following operations are performed on an 8-bit adder. Give the 8-bit sum produced for each, in **hexadecimal**.

0xF0 + 0x34 = 0x_____

0xF0 + 0x80 = 0x_____

J. Using a 5-bit two's complement representation, what is the range of integers that can be represented with a single 5-bit quantity?

K. Consider the following subtraction problem where the operands are 5-bit two's complement numbers. Compute the result and give the answer as a decimal (base 10) number.

$$\begin{array}{rr} & 10101 \\ - & \underline{00011} \end{array}$$

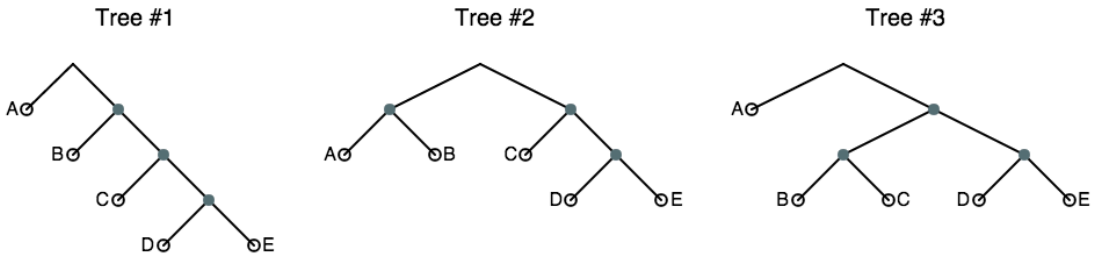
3. Variable-length Encodings

A. Given a variable X that can take on one of four values A, B, C, or D with the following probabilities.

Symbol	Probability
A	0.5
B	0.3
C	0.1
D	0.1

If you encoded this variable using a Huffman encoding, how many bits would be in the encoding of each of the symbols?

For each of the probability distributions for symbols A-E, select the Huffman encoding tree or trees that could result from running Huffman's algorithm on those probability distributions.



B. p(A) = 0.3, p(B) = 0.3, p(C) = 0.2, p(D) = 0.1, p(E) = 0.1. Tree(s): _____

C. p(A) = 0.6, p(B) = 0.1, p(C) = 0.1, p(D) = 0.1, p(E) = 0.1. Tree(s): _____

D. p(A) = 0.5, p(B) = 0.15, p(C) = 0.15, p(D) = 0.1, p(E) = 0.1. Tree(s): _____

E. p(A) = 0.5, p(B) = 0.2, p(C) = 0.15, p(D) = 0.05, p(E) = 0.1. Tree(s): _____

Baseball loves statistics! There are many different types of pitches that a pitcher can throw – the table below shows the probability for each type of pitch during 2014.

Type of pitch	*Probability*
Fastball	0.34
Change-up	0.14
Curveball	0.08
Slider	0.28
Other	0.16

F. How much information have you received when learning that particular pitch was NOT a fastball? You can express your answer as a formula if you wish.

G. To save on storage costs, Major League Baseball would like to use an optimal variable-length code to record, one at a time, the type of each pitch (i.e., to record one of the 5 types shown in the table above). Use Huffman's algorithm to derive such a code and list the resulting binary encodings below.

H. The table below shows the 2012-13 enrollments in the various EECS majors. To save a bit of space in their database the department would like to use a variable-length Huffman code to encode a student's choice of major. For each of the four majors, please give the encoding the department should use.

Major	**Count**	***p***	***p log*$_2$(1/*p*)**
6-1	74	0.09	0.30
6-2	387	0.44	0.52
6-3	360	0.41	0.53
6-7	54	0.06	0.25
Total	*875*	*1.00*	*1.60*

I. We wish to transmit messages comprised of the four symbols shown below with their associated probabilities and 5-bit fixed-length encoding

Symbol	p(symbol)	encoding
α	0.5	00000
β	0.125	11100
γ	0.25	11011
δ	0.125	10111

Huffman's algorithm is used to construct a variable-length code for the four symbols for transmitting a single symbol at a time. The resulting encoding could be

(1) α: 00, β: 01, γ: 10, δ: 10
(2) α: 00, β: 01, γ: 100, δ: 101
(3) α: 1, β: 01, γ: 000, δ: 001
(4) α: 0, β: 110, γ: 01, δ: 111
(5) none of the above

NerdLink is a new web-based startup that aims to keep MIT EECS students in touch with their parents. NerdLink streamlines parental communication by providing each student with an online choice of one of the five messages, then automatically fills in boilerplate and emails the parent a long and charming version of the message. The five messages, and their relative probabilities, are listed below:

Message #	*Message to parents*	*p(Message)*
M1	Send money!	60%
M2	I love this course called 6.004	8%
M3	I'm changing my major to Poetry	2%
M4	I'm getting a 5.0 this term!	1%
M5	Nothing much is new… (none of the above)	29%

NerdLink's initial implementation conveyed each message using a fixed-length code.

J. What is the average number of bits needed to convey a message, using a fixed-length code?

K. Given the probability distribution of the messages, what is the *actual* amount of information conveyed by message M5? Your answer may be a formula.

L. To enable error correction, the fixed-length code for a given message is sent *five* times. Using the five copies of the received message, in the worst case how many bit errors can be corrected at the receiver?

NerdLink, wanting to economize on communication costs, has hired you as a consultant to design a Huffman code for sending the messages.

M. Give the **number of bits** sent by your Huffman code for each message (M1 though M5), and the average number of bits transmitted per message using your code (a formula will be fine).

The Registrar's office would like to encode the letter grades (A, B, C, D, F) from a large GIR with 1000 students. They plan to encode each grade separately using a variable-length code. An analysis of previous terms has produced the following table of grade probabilities. In case it's useful, a thoughtful former 6.004 student has augmented the table by computing $p \log_2(1/p)$ for each grade.

Grade	p	$p \log_2(1/p)$
A	0.24	0.49
B	0.35	0.53
C	0.21	0.47
D	0.13	0.38
F	0.07	0.27
Totals	1.00	2.14

N. Use Huffman's algorithm to construct an optimal variable-length encoding.

O. Two 6.004 students have proposed competing variable-length codes. Alice says that encoding 1000 grades using her code will, on the average, produce messages of 2200 bits. Bob says that encoding 1000 grades using his code will, on the average, produce messages of 1950 bits. Which of the following is your best response when the Registrar asks your opinion?

(A) Choose Bob's: it has the shorter average length
(B) Choose Alice's: more bits means more information is transmitted
(C) Choose Bob's: Bob's average message length is less than the information entropy
(D) Choose Alice's: Bob's average message length is less than the information entropy
(E) Choose neither: a fixed-length code will have lower average message size

Best Choice (A through E): _____

4. Error Detection and Correction

A. A message about the suit of a card is sent using the encoding shown at the right. Using this encoding, how many bit errors can be detected? How many bit errors can be corrected?

```
Club:      000
Diamond:   011
Heart:     101
Spade:     110
```

B. A message about the suit of a card is sent using the encoding shown at the right. Give an example of a 5-bit received message with an uncorrectable single-bit error or write NONE if all single-bit errors can be corrected.

```
Heart:     00000
Diamond:   11001
Spade:     10111
Club:      01011
```

C. The MIT baseball coach likes to record the umpire's call for each pitch (one of "strike", "ball" or "other"). Worried that bit errors might corrupt the records archive, he proposes using the following 5-bit binary encoding for each of the three possible entries:

Strike	11111
Ball	01101
Other	00000

Using this encoding what is the largest number of bit errors that be *detected* when examining a particular record? The largest number of bit errors that can be *corrected*?

D. When transmitting the information about EECS majors over a noisy communication link, the department has chosen to use the 7-bit encoding shown on the right in the hopes that they'll be able to correct multiple-bit errors during transmission. Using this code, how many bit errors in a message about a single major will the receiver be able to correct?

```
6-1:  0101010
6-2:  1001100
6-3:  0110001
6-7:  1010010
```

E. We wish to transmit messages comprised of the four symbols shown below with their associated probabilities and 5-bit fixed-length encoding

Symbol	p(symbol)	encoding
α	0.5	00000
β	0.125	11100
γ	0.25	11011
δ	0.125	10111

If we transmit messages using the 5-bit fixed-length encoding shown above, will it be possible to perform single-bit error detection and correction at the receiver?

F. What is the Hamming distance between the encodings for A and B? Using an encoding scheme with this Hamming distance, how many bits of error can be detected? How many bits of error can be corrected?

```
A: 010010
B: 110101
```

An internet Sudoku gaming site transmits messages containing nine data bits and seven parity bits, arranged in a rectangle as follows:

D_{00}	D_{01}	D_{02}	P_{0x}
D_{10}	D_{11}	D_{12}	P_{1x}
D_{20}	D_{21}	D_{22}	P_{2x}
P_{x0}	P_{x1}	P_{x2}	P_{xx}

Each D_{ij} in the above diagram indicates a data bit, equally likely to be a 0 or 1. Each P_{ix} and P_{xj} is an odd parity bit chosen to make the total number of 1s in the i^{th} row or j^{th} column, respectively, odd. P_{xx} is an odd parity bit chosen to make the total number of 1s in the entire transmission odd. Thus in an error-free transmission, the total number of 1s in 4-bit columns 0 thru 2 and 4-bit rows 0 thru 2, as well as in the entire 16-bit transmission, is odd.

Note that each 9-bit data word determines a unique 16-bit *valid codeword* to be transmitted.

G. What is the minimum Hamming distance between valid codewords? [Hint: flipping one bit of the data word changes how many bits of the codeword?]

Each of the following represents a transmission received, with at most a single-bit error. For each message, circle the bit, if any, that was changed due to a transmission error.

H.

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

I.

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

J.

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

K.

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