

CSCI-E2 BITS - Problem Set 1 (60 points total)
Due February 15th 2011 at Noon

Question 1 (14 points)

- a. (4 points) The English language uses 26 letters, together with another 32 special characters plus the space character. How many bits are required to store a single character of English text (which could include upper and lower case letters, spaces and any of these special characters)? How many bytes should be allocated to store a single character of English text?
- b. (2 point) Based on your answer to part a., how much storage space (as a percentage) is wasted by requiring English text to be stored in complete bytes rather than just the minimum number of bits necessary?
- c. (4 points) The number of Chinese characters contained in the Kangxi dictionary (the standard Chinese dictionary used during the 18th and 19th centuries) is ~ 47,035. How many bits are required to store text consisting of these Chinese characters? How many bytes should be allocated per character? Does it make a difference if this alphabet includes the space, punctuation and special characters listed in part a.?
- d. (2 point) An alphabet of 109,000 characters covering 93 scripts would allow you to you to store documents written in most of the world's writing systems. How many bits would be required to store a character for this alphabet? How many bytes per character?
- e. (2 points) Based on your answer to d., how much space (as a percentage) is wasted by requiring characters in this universal alphabet to be stored in an integer number of bytes rather than just the minimum number of bits necessary?

Question 2 (18 points)

One of the Teaching Fellows in this course, David Abrams, bought his first hard drive with a capacity of 10 MB in March 1982 for \$1,000. He considered this an incredible bargain given that an advertisement just two years earlier in July 1980 had offered the same size drive for "only" \$4,495. In March 1986, a 20 MB drive could be purchased for \$960. In April 1989, Infoworld magazine tested a Seagate 42 MB drive that cost \$490. By Sept 1992, new computers were typically coming equipped with 210 MB drives that cost \$695. In Sept. 1995, 1 GB drives became available for \$625. A year later, in Sept 1996 you could get a 2.5 GB drive for \$440. In April 1998, 5.1 GB for \$280; May 1999, 10 GB for \$245; and in Aug 2000, 15 GB for \$144. In the past ten years, David Abrams has made the following hard drive purchases:

March 2001	30 GB	\$110
July 2002	60 GB	\$240
May 2007	500 GB	\$110
June 2007	500 GB	\$93.50
July 2007	500 GB	\$105
March 2008	750 GB	\$150
Nov 2009	1.5 TB	\$110
March 2010	1 TB	\$75
May 2010	1 TB	\$50
Sept 2010	2TB	\$108
Nov 2010	1.5 TB	\$60

(Note that hard disk manufacturers use Megabytes to refer to 10^6 bytes – not 2^{20} bytes. Similarly, Gigabytes refers to 10^9 bytes and Terabytes refers to 10^{12} bytes)

- (10 points) The cost of hard drive storage capacity between 1980 and 2010 is an example of exponential growth. Using Excel (or similar), plot the number of megabytes of hard disk storage a dollar buys over the period from 1980 through 2000 on the first graph and from 1980 through 2010 on the second. Do you notice any similarity between the two graphs? Based on the graphs, what can you say about exponential growth?
- (4 points) From your graphs, estimate how long in years it takes to get twice the storage space for the same dollar cost.
- (4 points) The Library of Congress has a collection of ~ 29 million books. Assume it takes about 2048 bytes to store a single page of text from a book and a typical book is 200 pages long. If the cost of hard drive capacity continues to drop at the rate you estimated in b., and you are willing to pay \$100 for a hard drive, in what year will you be able to store all the books in the Library of Congress on your personal hard drive?

Question 3 (23 points)

Imagine you are on an exploration voyage and you encounter a series of caves covered with a strange, previously unknown language.

Curiously, you find that the messages consist of the following six symbols:



You decide that this discovery is important enough to be sent back to base. However, due to the complexity of the symbols, and the poor performance of your data network out of your remote jungle location, you decide to transmit the messages using a binary representation of the symbols, rather than to send images of the symbols themselves.

a. (3 points) Generate a lookup table of symbols and binary representations using a fixed length binary encoding scheme.

b. (5 points) How would you encode the following string using the encoding scheme you have developed in part (a)



c. (6 points) Now generate a Huffman code for this string. Make sure you show your Huffman tree.

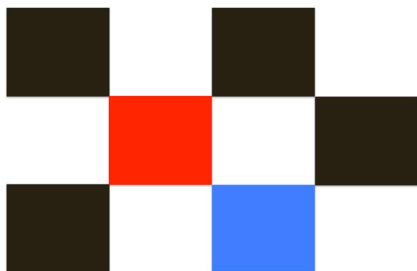
d. (4 points) What is the average number of bits you need to encode a symbol for the fixed length binary code? What about the Huffman code? How many bits do you 'save' by encoding the string using your Huffman code?

e. (5 points) What is the entropy of the string of symbols? What is the difference in efficiency between the two encoding schemes?

Question 4 (5 points)

Which of these pictures has higher entropy? Why?

(a)



(b)

