

## 18.310 Homework # 3 Due Wednesday, September 24

1a: Suppose we have 12 symbols, A, ..., L that occur with  $f(x)$  as follows in a message:  $f(A)= 10$ ,  $f(B)= 20$ , with the remaining  $f$ 's in order as  
11, 30, 9,4,7,8, 31, 7,1,2 .

Compute the Shannon entropy formula for the minimum of the average number of bits needed per symbol and the total minimum length of the message.  
(it is very easy to do this on a spreadsheet)

1b: Construct a Huffman code for these symbols and compute the average word length per symbol in the message and the total length message.

1c: Construct a Hu-Tucker code for same, with the order of code words for the symbols the same as alphabetical order of the symbols. Compute the same two numbers for this code.

2. Read Peter Shor's notes on the Lempel-Ziv algorithm, and describe it as concisely as you can in your own words. Give an example of encoding and decoding a sequence using it.

3: What is a length 30 sequence of 0's and 1's which would require the most bits to encode using the Lempel-Ziv compression algorithm discussed in class?  
What is a length 30 sequence which would require the fewest bits?

4: Compute the expected number of comparisons that the quicksort procedure makes as follows:

Consider the following sorting algorithm. Start with an array containing the keys in any order. Choose a random permutation  $P$  of the keys (uncorrelated with the actual order of the keys in the array). Now, go through the keys in the order of  $P$  one step (involving several comparisons) associated with each key. That step involves comparing the given key with all keys that lie in the same interval between previously keys in  $P$ . After that step you know the exact rank of the given key and whether it is above or below each of the keys after it in  $P$ . Continue this until all keys are in sorted order.

4a: Argue that this algorithm gives exactly the same number of comparisons as quick-sort.

4b: Consider an indicator variable  $I_{i,j}$  for every pair of keys  $(i; j)$  **with  $i < j$**  which is 1 if the  $i$ th key and the  $j$ th key are compared during the sort, and 0 otherwise (here  $i$  and  $j$  refer to the original sorted order of the keys). Show that the expectation value of this variable is  $2/(j-i+1)$ .

4c: Use 4b to show that the expected number of comparisons in quicksort is approximately  $2n \ln n$ , where  $\ln$  is the logarithm to base  $e$ . The fact that the sum of the reciprocals of the numbers from 1 to  $j$  lies between  $\ln(j)$  and  $1+\ln(j)$  is a useful fact here

5 Consider the algorithm for finding the maximum of  $n$  keys where you choose a random key, compare it to all the other keys, and throw away everything less than it, then repeat on the remaining keys.

5a: Show that for this algorithm, with the indicator variable  $I_{ij}$  defined as in 4b, its expectation is given by  $2/(n-i+1)$ .

5b: Use 5a to approximately compute the expected number of comparisons in this randomized maximum-finding algorithm. You should find the constant on the leading term.

6a: On a spreadsheet or otherwise compute  $w(n)$  which is  $n!/((n/e)^n)$  for  $n=1,2,4,\dots,128$ . Square these results and divide by  $2n$ .

6b: Extrapolate these results. You can do that by listing your answers in a column in increasing order of  $n$  and computing the next column by putting in the each row twice the next row entry less the entry in the same row of the column you start with.

(this eliminates errors of the form  $c/n$  from your results)

Then in the next column put  $4/3$  times the next row entry less  $1/3$  times the same row entry into each row. (This eliminates errors of the form  $c/n^2$ .)

Then do the same with  $8/7$  and  $1/7$  as coefficients here to eliminate errors of the form  $c/n^3$ .

By this time you should be able to recognize the answer. (you can go on with  $16/15$  and  $-1/15$  as coefficients etc. if you want)

Deduce the approximate formula for  $n!$  of the form  $c(n/e)^{n+1/2}$  implied by all of this. (determine  $c$  here)

7. Extra Credit: Use the same technique of question 6. to find an approximation formula for  $n!$  of the form  $(c(n/e)^{n+1/2})(1+d/n)$ ; that is, determine  $d$  here.