CSE 21 HW 7

Brian Masse

March 5, 2025

1. A slow-growing sequence of length n (with n $\geq$ 1) is a non-decreasing sequence of integers that start with 1 and each pair of entries differ by at most 1.

a) (for n $\geq$ 1), How many slow-growing sequence of length n are there?

$$A(n) = 2 \cdot A(n-1), A(1) = 1$$

$$\begin{array}{rl} 1) & A(n) = 2 \cdot A(n-1) \\ 2) & A(n) = 2^2 \cdot A(n-1) \\ & \vdots \\ k) & A(n) = 2^k \cdot A(n-1) \\ & \vdots \\ n-1) & A(n) = 2^{n-1} \end{array}$$

b) How many bits would the most efficient fixed-length encoding of sequenes use?

$$\begin{aligned} n &= \lceil log_2(2^{n-1}) \rceil \\ &= n-1 \end{aligned}$$

c) Develop your own encoding / decoding algorithm where the code uses this number of bits

Create an encoding such that each bit in the encoded binary string represents whether to add 1 or 0 to the previous number in the string.

d) Use your encoding to encode the follow slow-growing sequences

- *(1,2,3,3,3,4,4,5)* = 1100101
- *(1,1,1,2,2,3,4,4)* = 0010110
- *(1,2,2,3,3,4,5,6)* = 1010111

e) Use your decoding to decode the following strings

- 01010100 = (1,1,2,2,3,3,4,4,4)
- 11100011 = (1,2,3,4,4,4,4,5,6)
- 10111110 = (1,2,2,3,4,5,6,7,7)

2. Image files can be encoded using binary strings. In the most simple version, you can encode an nxm black and white image using nm bits with black corresponding to 0 and white corresopnding to 1.

We can use hexadecimal to encode each of the cunks of 4 pixels into a single hexadecimal character.

a) How many bits are required to encode this image by encoding each pixel with 1 bit?

$$nm = (96)(96) = 9216$$

b) How many hexadecimal characters are needed for this image?

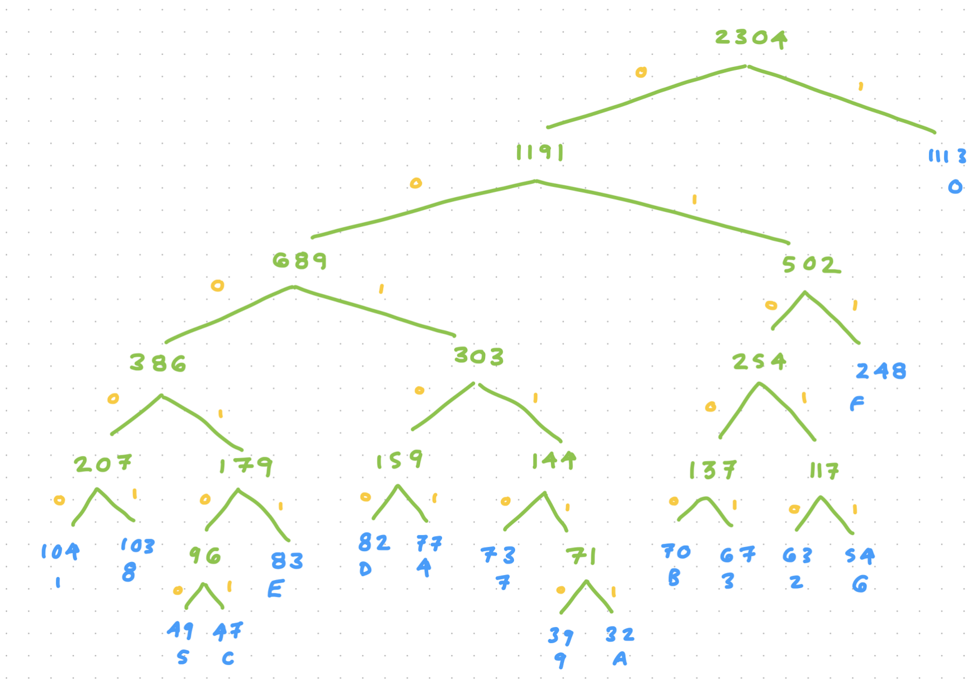
$$\left(\frac{96}{4}\right)(96) = 2304$$

- $\frac{96}{4}$: Groups per row
- 96 : Number of Rows

c) Huffman encoding is actually used in image compression. What we cand o is compute the frequency table of the hexadecimal chracters and build a Huffman code based on that. Then encode the hexadecimal string using the Huffman code. Here is the frequncy table for this particular image:

Character	*0*	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*A*	*B*	*C*	*D*	*E*	*F*
Weight	*1113*	*104*	*63*	*67*	*77*	*49*	*54*	*73*	*103*	*39*	*32*	*70*	*47*	*82*	*83*	*248*

- *Draw the huffman tree for the set of frequencies.*



- *Give the code for each character*

Number	*Code*
0	1
1	00000
2	01010
3	01001
4	00101
5	000100
6	01011
7	00110
8	00001
9	001110
A	001111
B	01000
C	000101
D	00100
E	00011
F	011

- *Calculate the total number of bits needed to encode this particular image of the moon using this coding.*

$$T(n) = 1113(1) + 104(5) + 63(5) + 67(5) + 77(5) + 49(6) + 54(5) + 73(5) + 103(5) \\ +39(6) + 32(6) + 70(5) + 47(6) + 82(5) + 83(5) + 248(3)$$

3. Consider the set of 26 capital letters of the Roman Alphabet

a) Using a fixed length character-by-character encoding, what is the minimum number of bits required to encode each character?

$$n = \lceil log_2(26) \rceil = 5$$

b) Develop a fixed length character-by-character encoding for this alphabet using the number of bits from the previous part and use it to encode / decode the following strings

- *Encode: "MATH"* $\implies (13)(1)(20)(8) \implies$ *01101 00001 10100 01000*
- *Encode: "BYTE"* $\implies (2)(25)(20)(5) \implies$ *00010 11001 10100 00101*
- *Decode: 10010 01110 10001 10011* $\implies (18)(14)(17)(19) \implies$ *RNQS*
- *Decode: 10001 01110 01110 10011* $\implies (17)(14)(14)(19) \implies$ *QNNS*

c) Using fixed length encoding. What is the minimum number of bits required to encode each 4 letter string over this alphabet.

$$n = \lceil log_2 26^4 \rceil = 19$$

d) Develop a fixed length encoding for 4 letter strings using the number of bits from the previous part and use it to encode / decode the following strings:

- *Encode: "MATH"* $\implies (13 \cdot 26^3 + 1 \cdot 26^2 + 20 \cdot 26 + 8) \implies 229692$
  $\implies$ *011 1000 0001 0011 1100*

- *Encode: "BYTE"* $\implies (2 \cdot 26^3 + 25 \cdot 26^2 + 20 \cdot 26 + 5) \implies 52577$
  $\implies$ *000 1100 1101 0110 0001*

- *Decode: 100 1111 1010 1001 0101* $\implies 326293 \implies$ *RNQS*
- *Decode: 100 1011 0101 1001 1111* $\implies 308639 \implies$ *QNNS*

4. Suppose you are traveling from the bottom left corner of a 9 by 6 grid of city blocks and you wish to get to the top right corner only using up and right movements

a) What is the minimum number of bits necessary to encode these paths with a fixed length encoding?

There are $\binom{6+9}{6} = \binom{15}{6} = 5005$ number of paths from the bottom left to the top right, where 1s represent moving to the right.

$$n = \lceil log_2 5005 \rceil = 13$$

b) Develop an encoding strategy and encode the path given in the example

Translate the path to a fixed-density binary string. Specifically, a length 13 binary strings with 6 1s, where 1 represents moving to the right and 0 represent moving up.

$$\text{path} \implies 111100001000100$$

c) Use your encoding strategy to decode the following string: 0 1010 1100 1100