

CS180 - Huffman Codes

Announcements

- Program due ~~Wednesday~~ Thursday

Idea

We want to transmit information using as few bits as possible.

Standard ASCII: 8 bits per character,

Hello - 40 bits

Fixed length codes

32-bits in newer version

Hello - 32×5

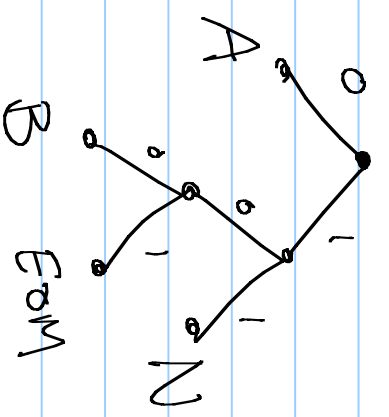
So- how can we do better?

What if we don't use every character?

Take characters we need.

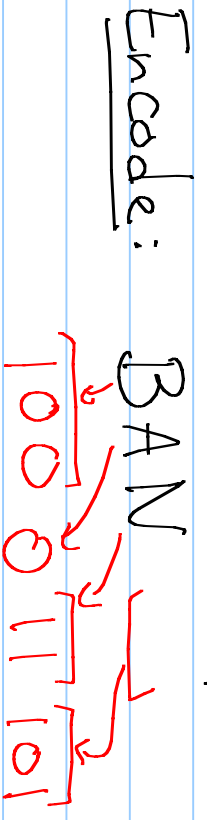
frequency-based \rightarrow use fewer bits
for common characters

Prefix-free codes



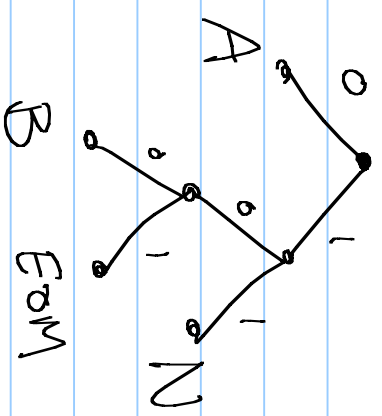
An unambiguous way to send information when we have characters that are not of a fixed length.

No letter's code is the prefix of another letter.



Decades:

1000 11011 6101
BANANA



So how do we do this? With exact frequency counts!

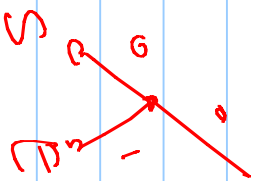
This sentence contains three a's, three c's, two d's, twenty-six e's, five f's, three g's, eight h's, thirteen i's, two l's, sixteen n's, nine o's, six r's, twenty-seven s's, twenty-two t's, two u's, five v's, eight w's, four x's, five y's, and only one z.

Letters : A, C, D, E, F, G, H, I, L, N, O, R, S, T
3 1 3 2
U, V, W, X, Y, Z

Using frequency counts, build one of those
trees.

A	C	D	E	F	G	H	I	L	N	O	R	S	T	U	V	W	X	Y	Z
3	3	2	26	5	3	8	13	2	16	9	6	27	22	2	5	8	4	5	1

Which ones should get few bits?



Which ones should use the
most bits?

Z, D or U or L

Huffman's algorithm

Take the two least frequent characters.

Merge them into 1 letter, which becomes a new "leaf".

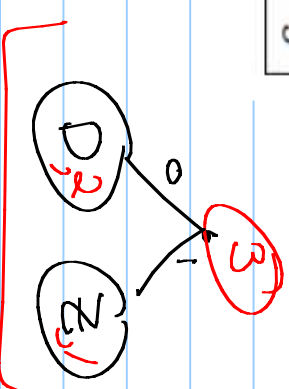
Example: ↓

A	C	D	E	F	G	H	I	L	N	O	R	S	T	U	V	W	X	Y	Z
3	3	2	26	5	3	8	13	2	16	9	6	27	22	2	5	8	4	5	1



Merge D & Z:

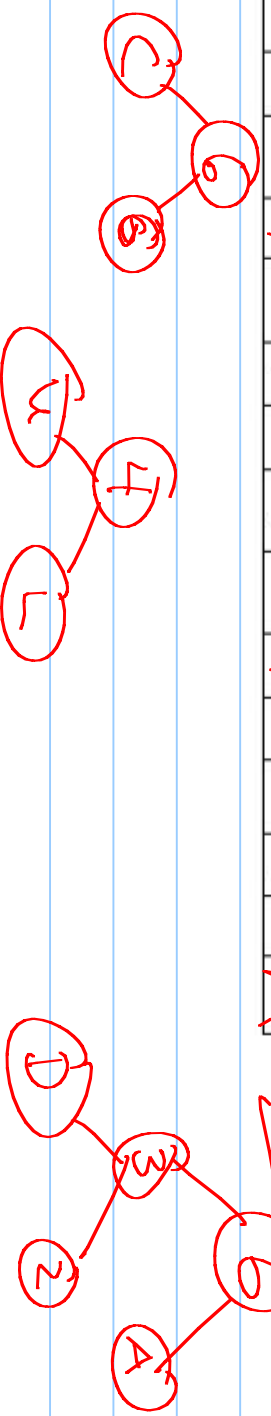
A	C	E	F	G	H	I	L	N	O	R	S	T	U	V	W	X	Y	<u>DZ</u>
3	3	26	5	3	8	13	2	16	9	6	27	22	2	5	8	4	5	3



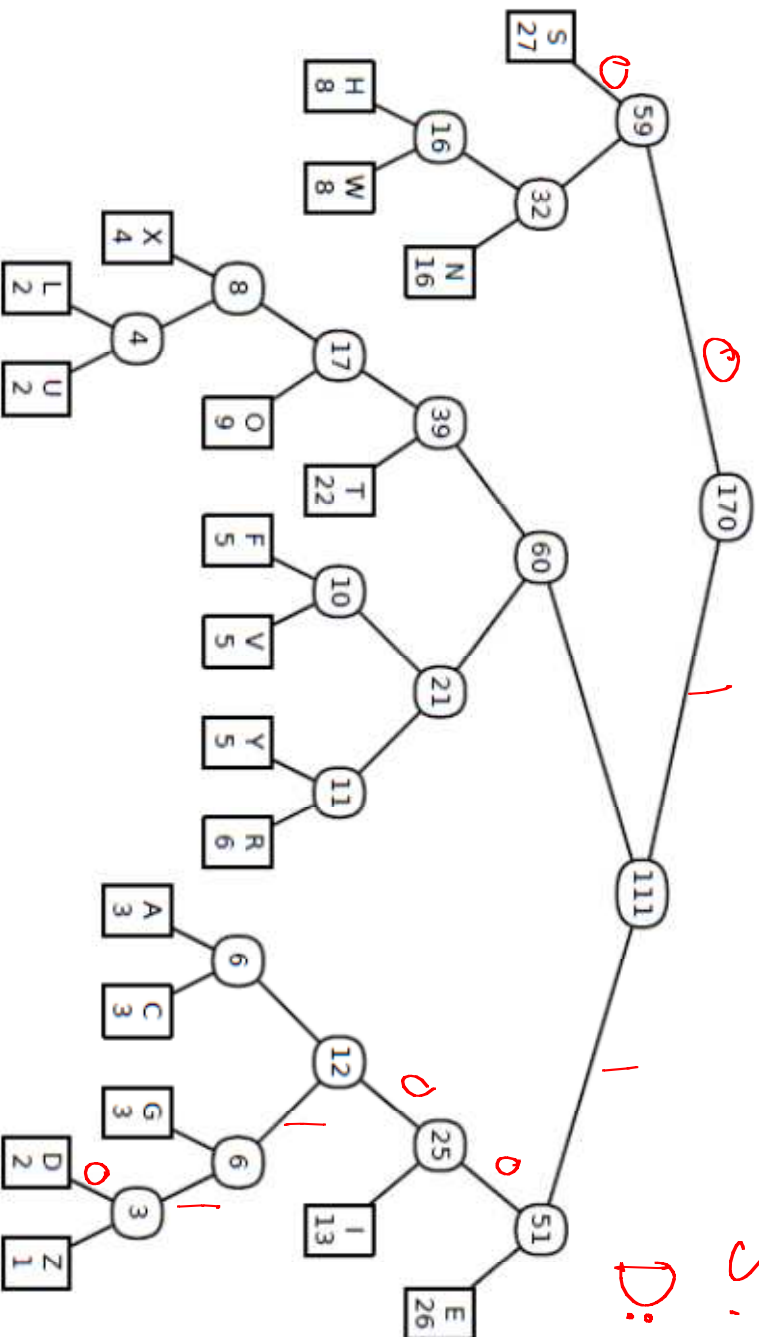
X	C	E	F	G	H	I	L	N	O	R	S	T	U	V	W	X	Y	Z
3	3	26	5	3	8	13	2	16	9	6	27	22	2	5	8	4	5	3



Next ?

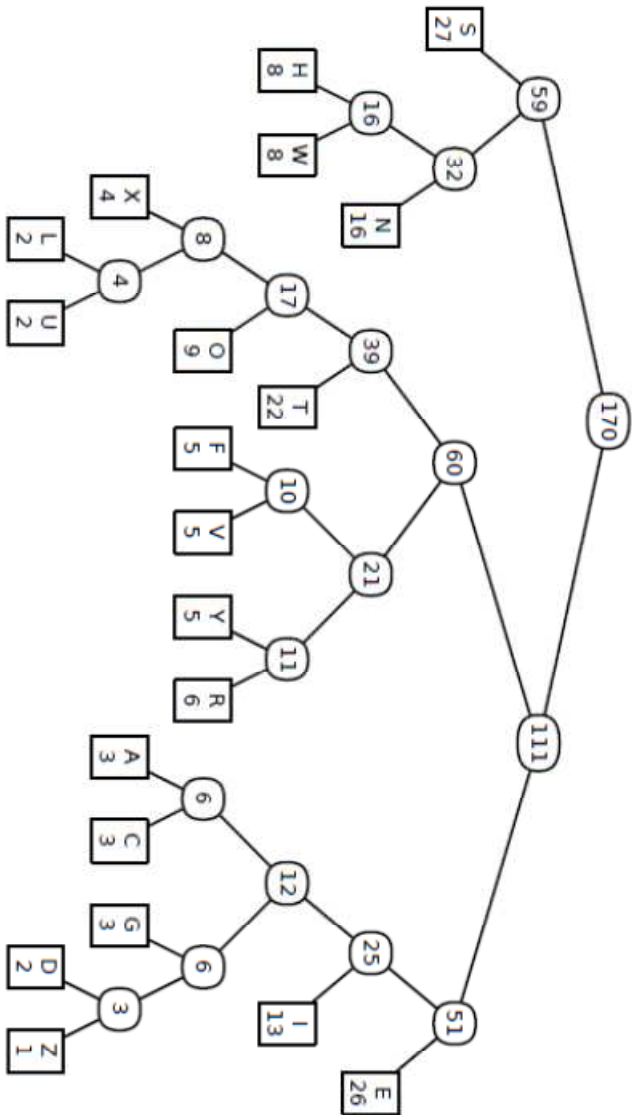


In end, build a tree!



S: 00
D: 1100110

Using the tree:



I 1001 0100 1101 0000 111 011 1001 111 011 110001 111 110001 10001 011 1001 110000 1101
 H I S S E N T E N C E C O N T A I



How many bits?

char.	A	C	D	E	F	G	H	I	L	N	O	R	S	T	U	V	W	X	Y	Z
freq.	3	3	2	26	5	3	8	13	2	16	9	6	27	22	2	5	8	4	5	1
depth	6	6	7	3	5	6	4	4	7	3	4	4	2	4	7	5	4	6	5	7
total	18	18	14	78	25	18	32	52	14	48	36	24	54	88	14	25	32	24	25	7

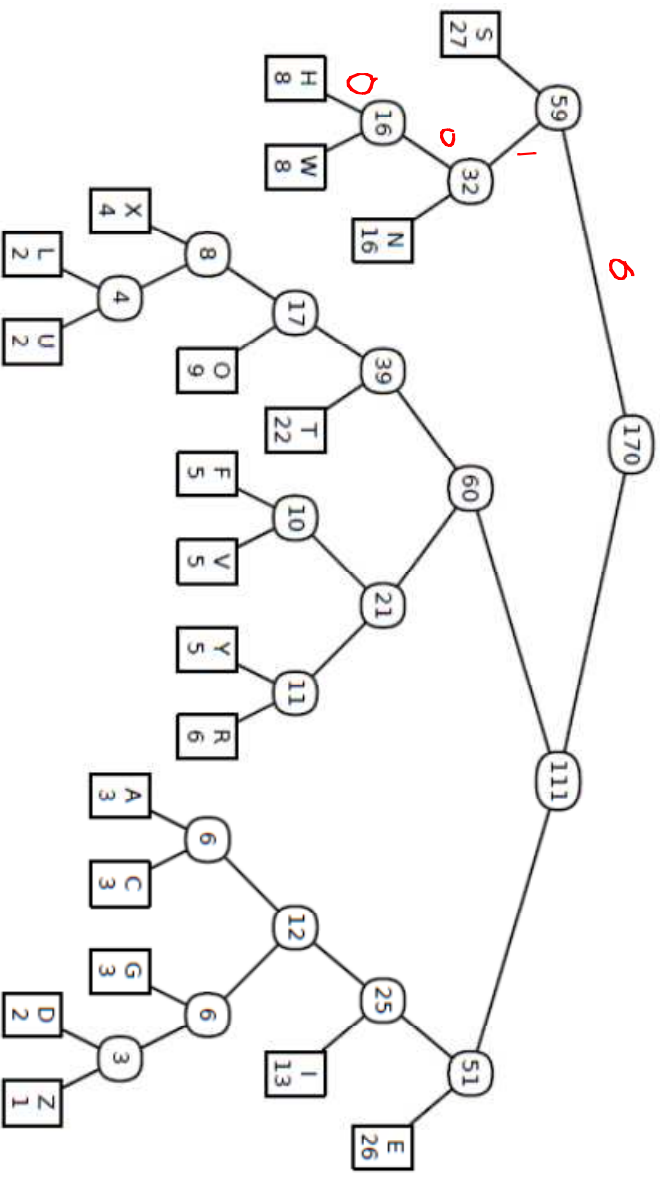
$$\text{total} = \underline{646} \text{ bits}$$

How many bits would ASCII use to send charset letters?

$$170 \times 8 = 1360$$

Exercise: $01001111000010100001010001$

H
E
L
L
O



Message? HELLO ASCII : 40-bits

How many bits? 26 bits

Good to know:

- Huffman trees are optimal
- Greedy algorithm

Next program: Decode

Read a tree
Build the tree
Decode a message