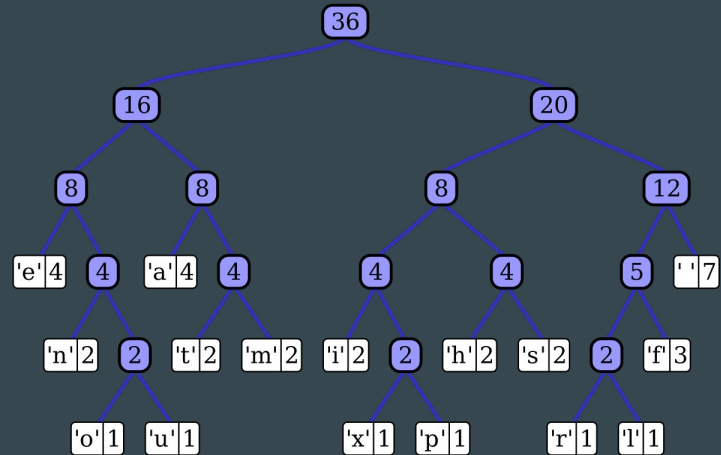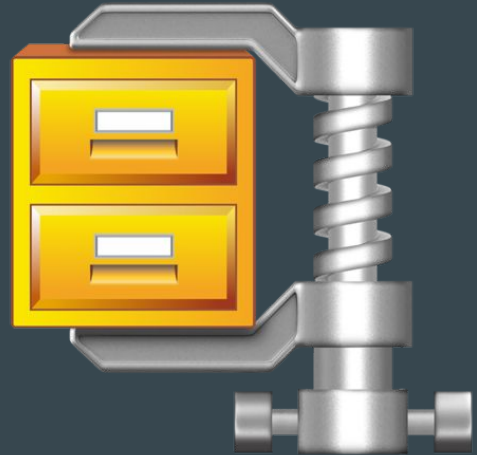# Huffman Coding

●●●

By: Martin Silva

What Huffman Coding Is

- Invented by David A. Huffman
- Algorithm which creates a unique encoding table for a specific piece of data
- Lowers the amount of bits necessary to store a file.
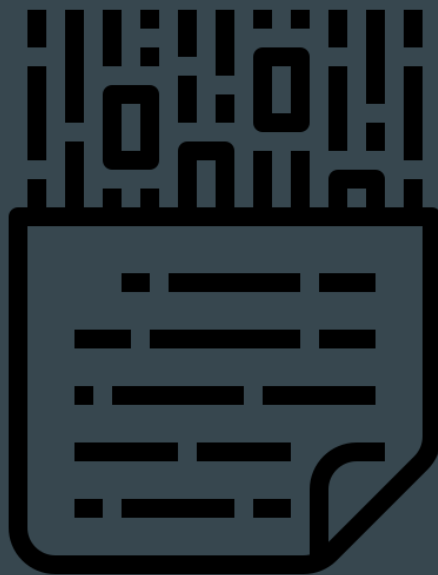
Why is Huffman Coding Important?

- Increase in file sizes throughout the years
- Faster download/upload speeds
- More efficient use of resources, especially important in the past.
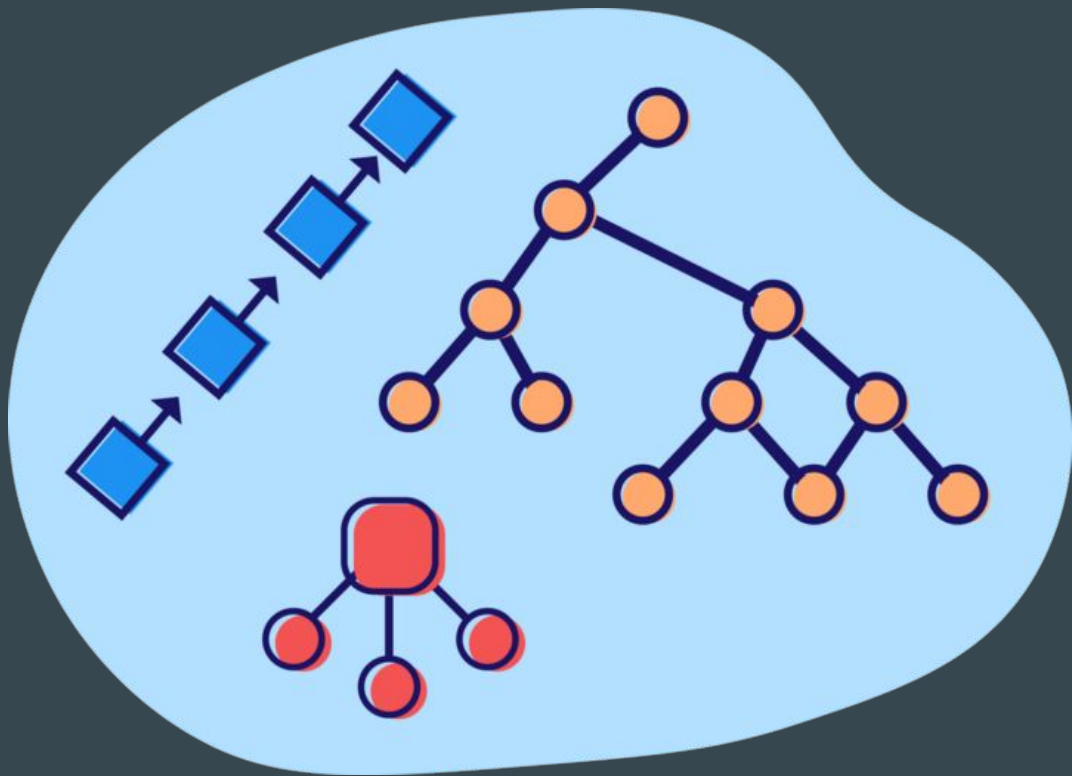
How Huffman Coding Works

Requires:

- Frequency Distribution
- Huffman Tree
- Encoding Table

Let's Write a Program…

Which data structures should we use?

Frequency Distribution

- Read each character in the file one-by-one
- Increment the value of each key

'g'

15

'a'

10

Priority Queue

- Convert key-value pairs into huffman nodes.
- Create a priority queue of huffman nodes.

Priority queue should be changed to

sort by lowest frequency to highest frequency.

element with the lowest priority

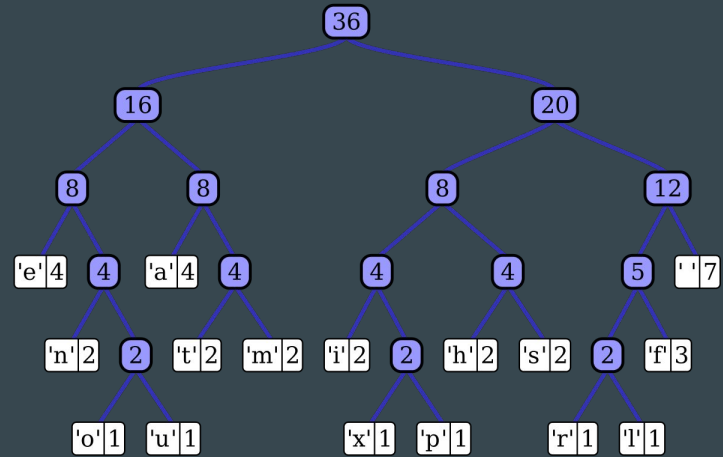| 2	6	7	10	11

element with the highest priority
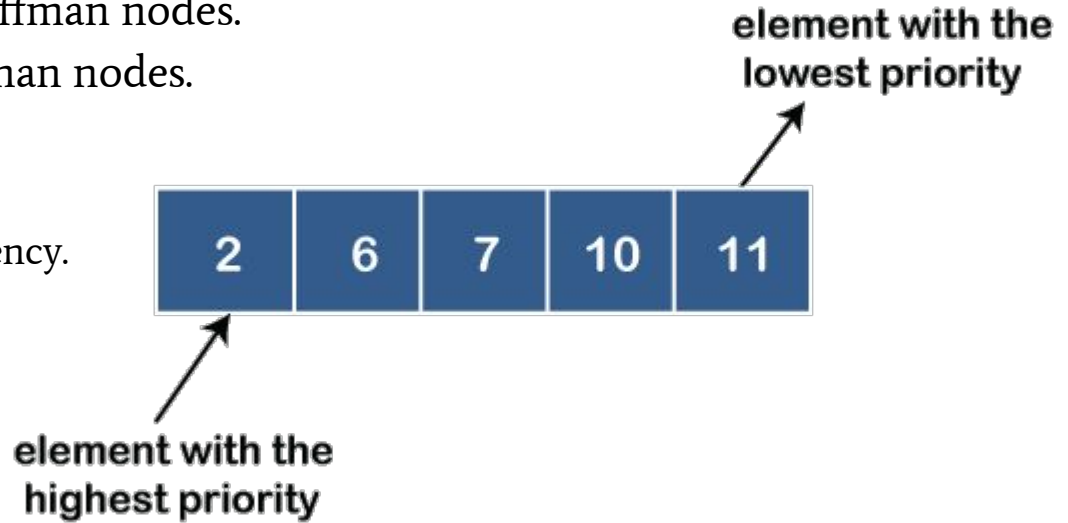
Huffman Tree

Head node contains the frequency of all characters in the file.

Connects all nodes together.

Each leaf node has a character and bitstring.

All other nodes have only a frequency.

Encoding

"this is an example of a huffman tree"

⬇

"011010101000101111110001011111010001 0111…"

Char ⬍	Freq ⬍	Code ⬍
space	7	111
a	4	010
e	4	000
f	3	1101
h	2	1010
i	2	1000
m	2	0111
n	2	0010
s	2	1011
t	2	0110
l	1	11001
o	1	00110
p	1	10011
r	1	11000
u	1	00111
x	1	10010

Decoding

- Move through bit-by-bit until forming a known bit string.
- Record our character and find the next bit string.

It works, but why?

Uses less bits per character than char (8 bits).

All larger sized bit strings are used the least.