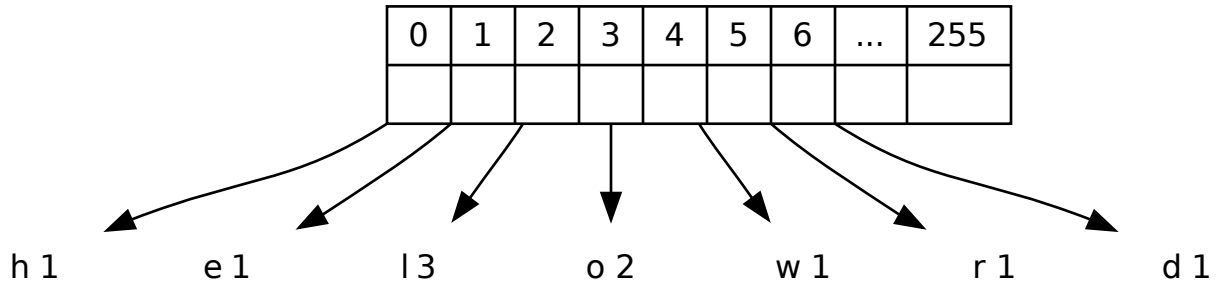
Huffman: Complete Trace

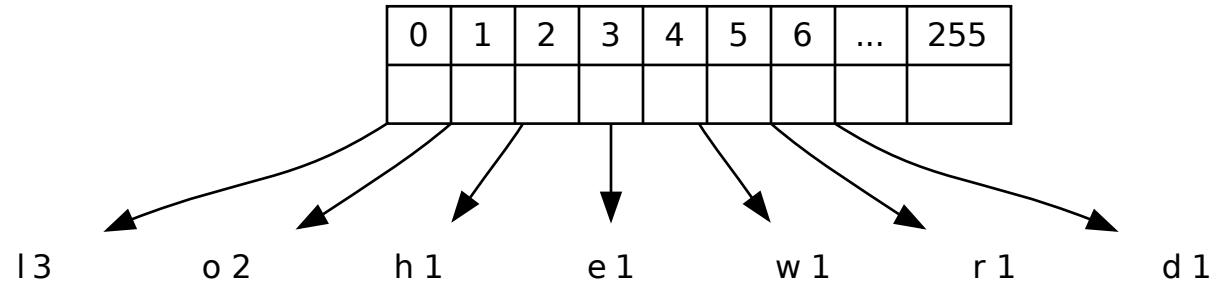
Fall Semester

Trace of the Encode Algorithm (1)

Original message:

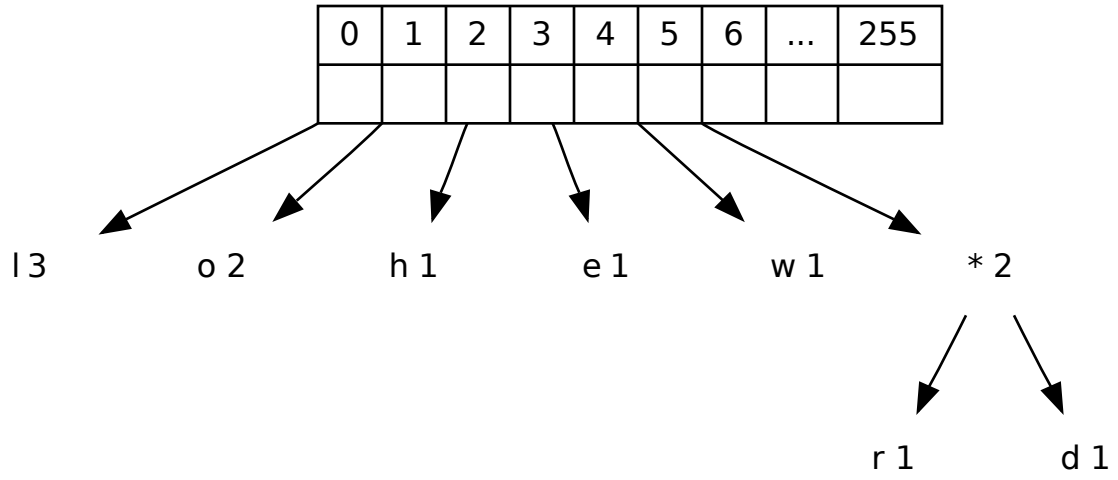


Sorted by frequency:



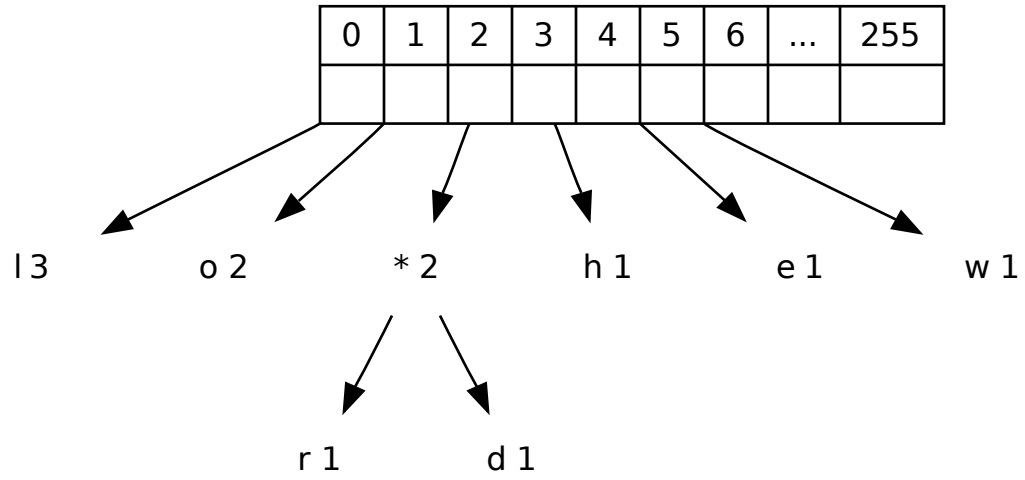
Trace of the Encode Algorithm (2)

- The new node's frequency must be the sum of the frequencies of the two nodes being glued together. The new node's symbol is irrelevant but for debugging purposes, if you want to print the array of trees, use something conspicuous like an asterisk.



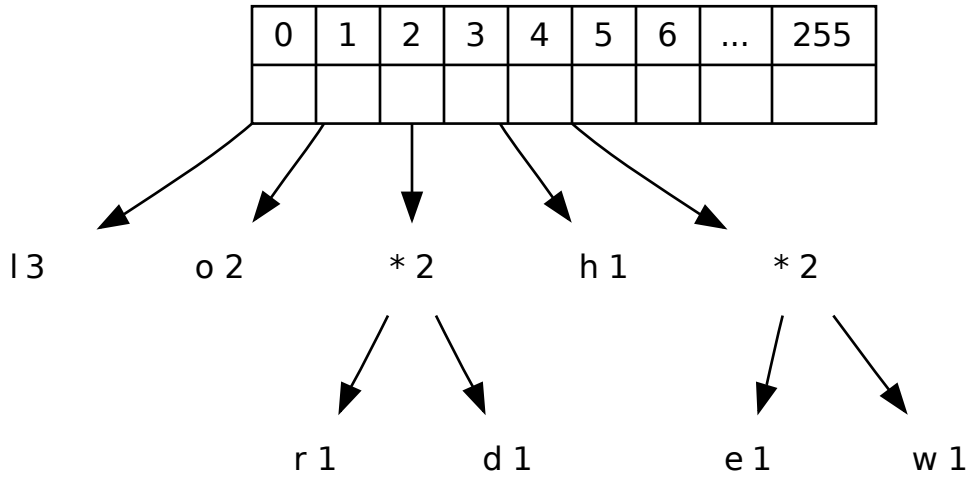
Trace of the Encode Algorithm (3)

Re-sorted by frequency:



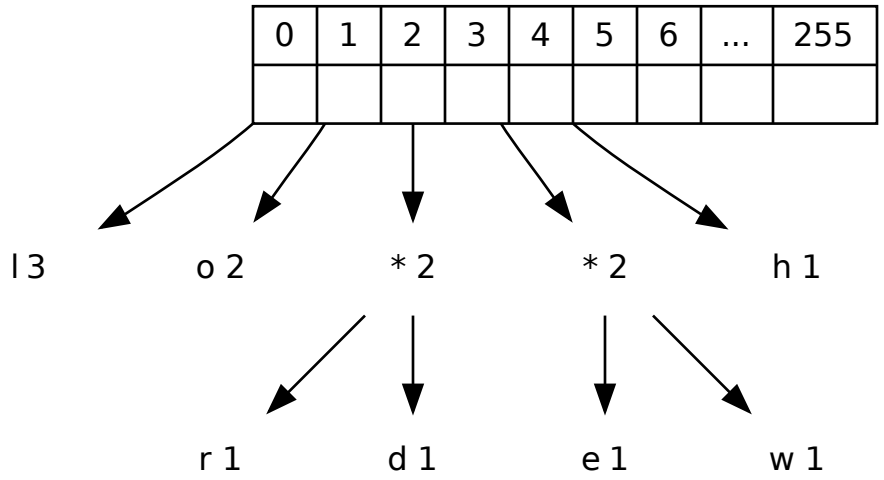
Trace of the Encode Algorithm (4)

Next low two glued together:

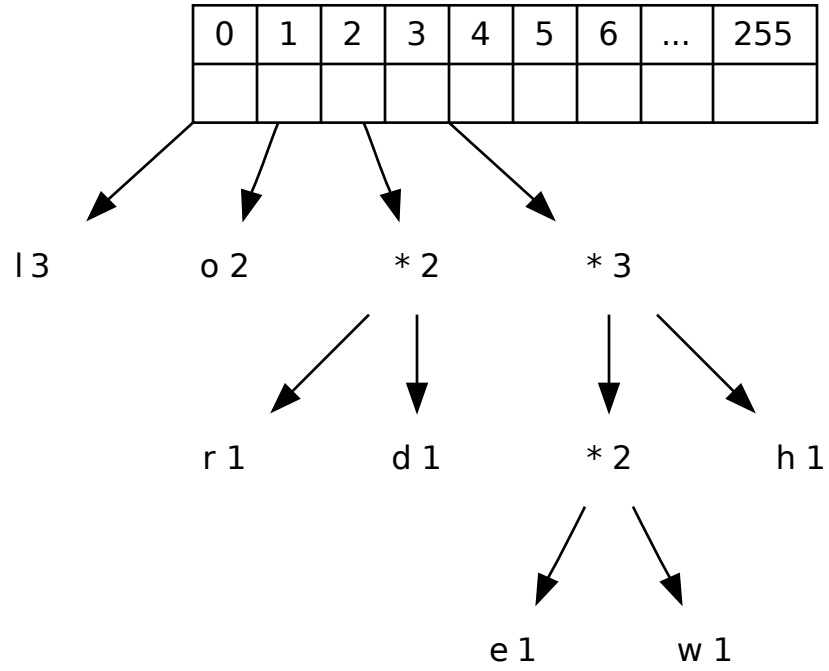


Trace of the Encode Algorithm (5)

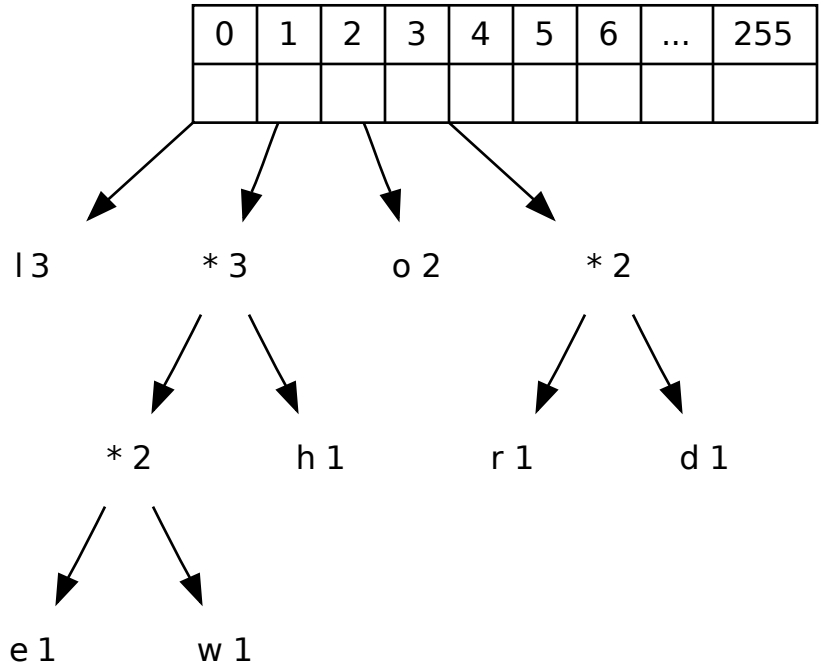
Re-sorted by frequency:



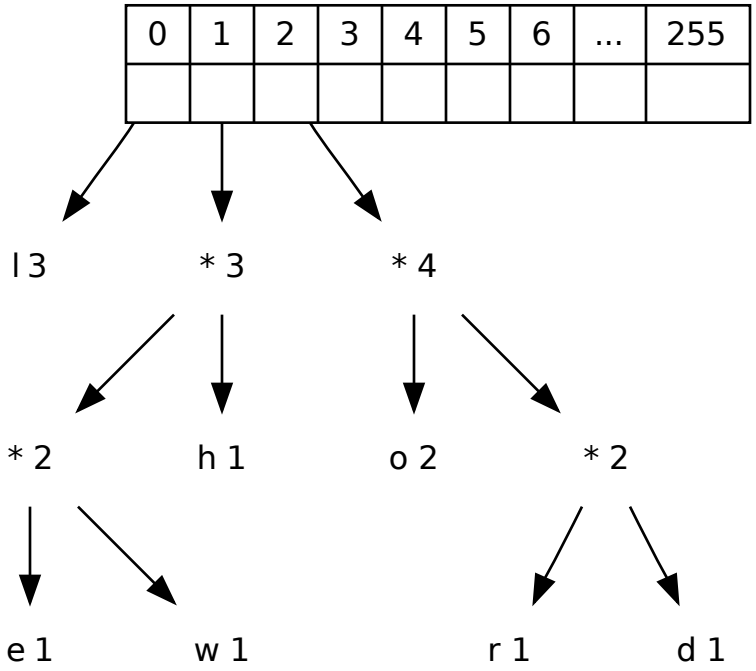
Trace of the Encode Algorithm (6)



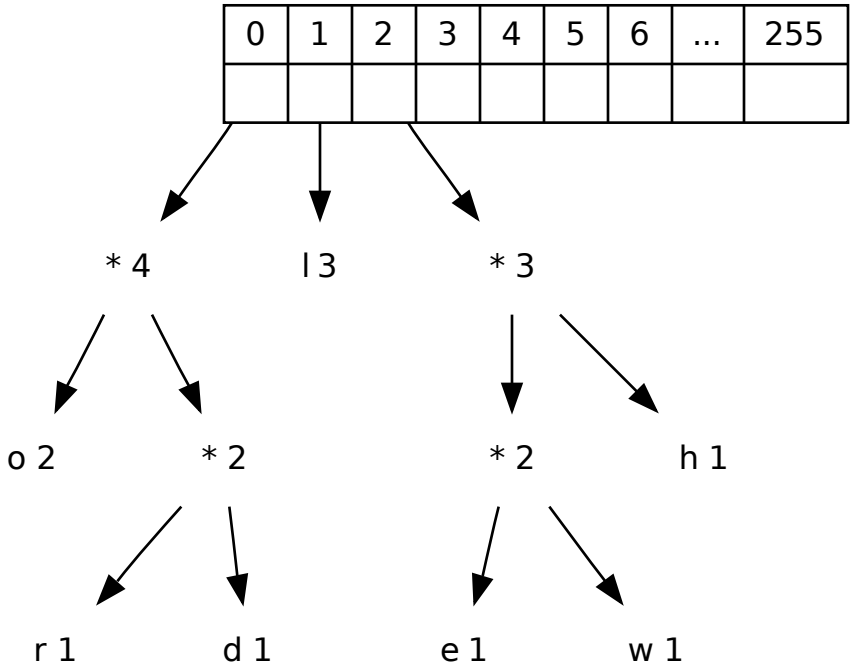
Trace of the Encode Algorithm (7)

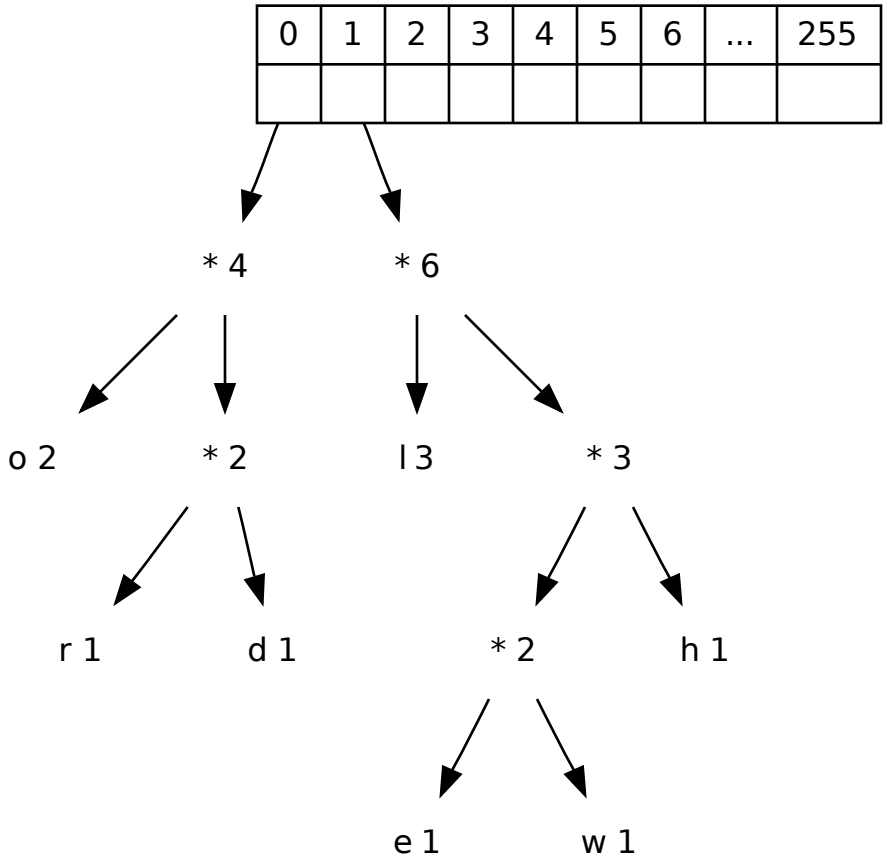


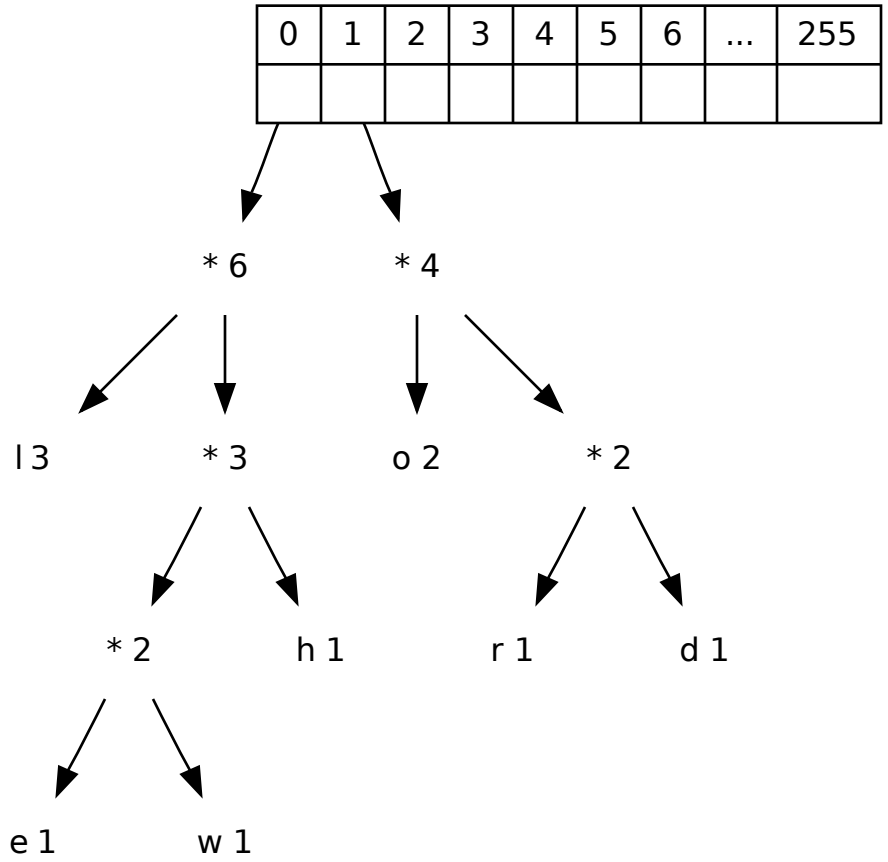
Trace of the Encode Algorithm (8)

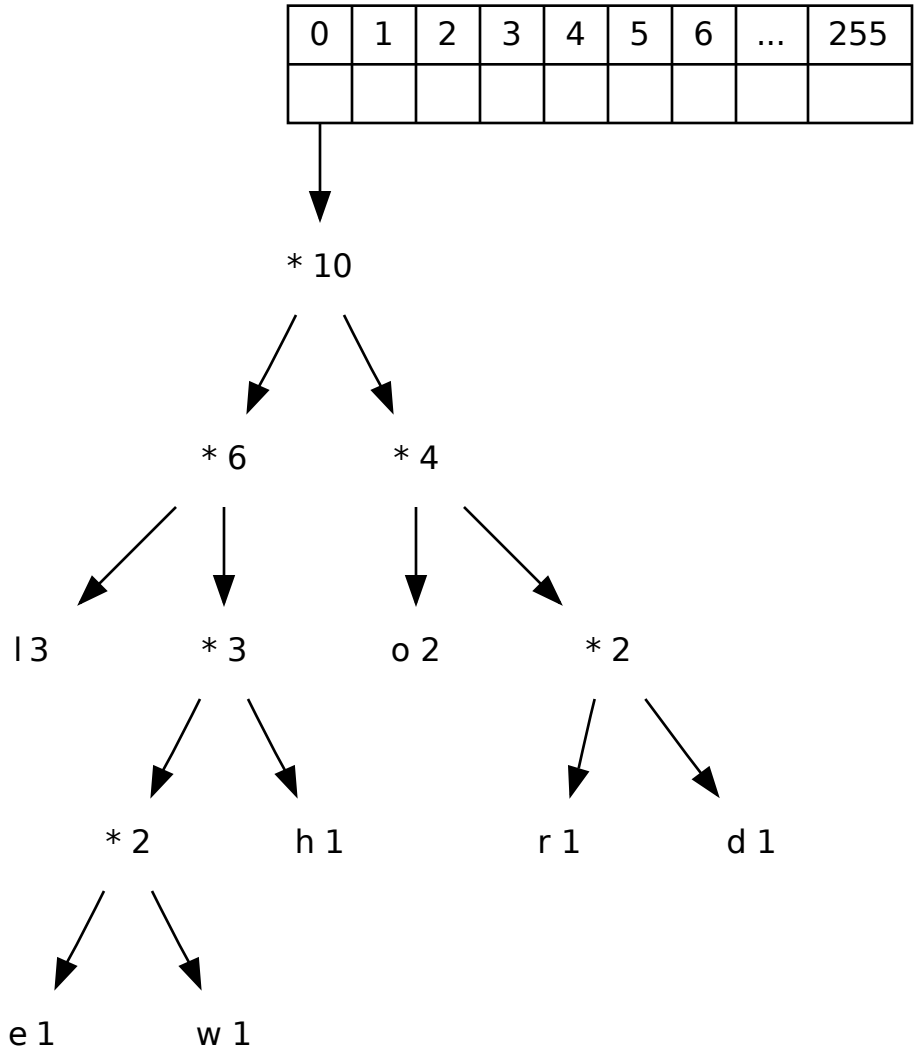


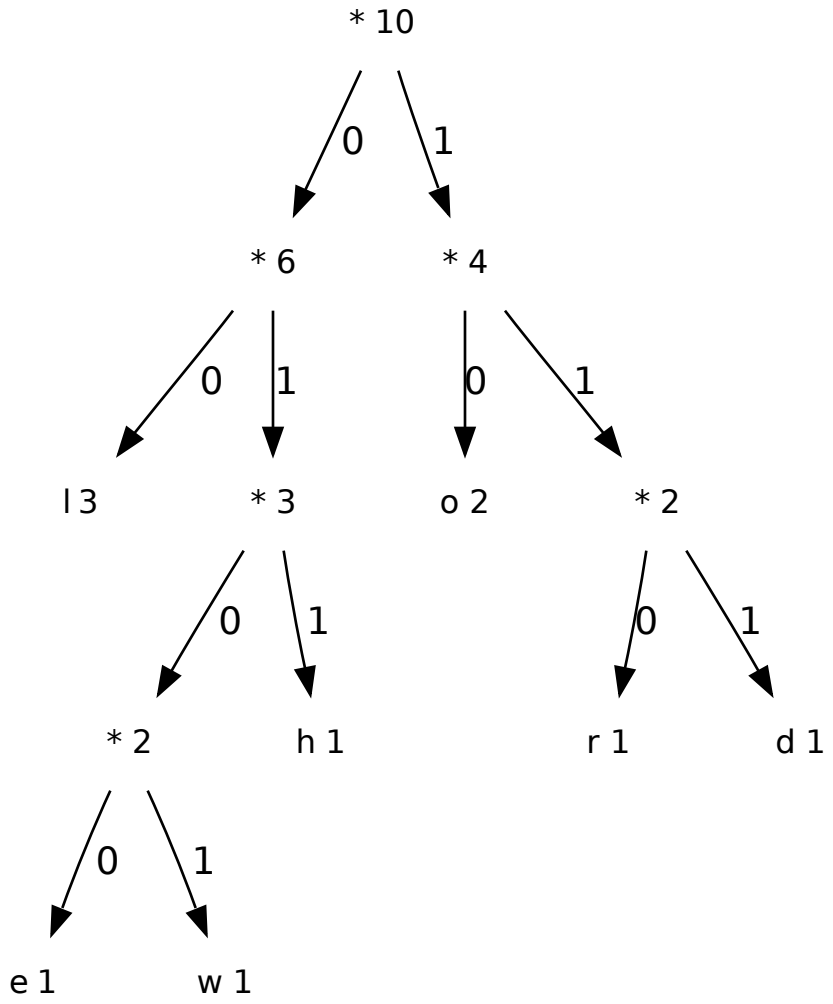
Trace of the Encode Algorithm (9)




0
1
2
3
4
5
6
...
255
* 4
* 6
o 2
* 2
l 3
* 3
r 1
d 1
* 2
h 1
e 1
w 1


0
1
2
3
4
5
6
...
255
* 6
* 4
l 3
* 3
o 2
* 2
* 2
h 1
r 1
d 1
e 1
w 1


0
1
2
3
4
5
6
...
255
* 10
* 6
* 4
l 3
* 3
o 2
* 2
* 2
h 1
r 1
d 1
e 1
w 1


* 10
0
1
* 6
* 4
0
1
0
1
l 3
* 3
o 2
* 2
0
1
0
1
* 2
h 1
r 1
d 1
0
1
e 1
w 1

Example Entropy Calculation (1)

x	F_x	P_x	H_x	bits
h	1	0.100	0.332	3.322
e	1	0.100	0.332	3.322
l	3	0.300	0.521	1.737
o	2	0.200	0.464	2.322
w	1	0.100	0.332	3.322
r	1	0.100	0.332	3.322
d	1	0.100	0.332	3.322

- Start simple and only once everything is working then try a more complicated input like the `declaration.eng` file.

Example Entropy Calculation (2)

- The bits column, not required by the lab, indicates the theoretical minimum number of bits for each symbol in the message.
- The entropy H_x multiplies the bits times P_x, essentially weighting the value so that the overall H gives the average number of bits needed across all symbols. Thus, $\lceil H \cdot n \rceil$ is the total bits.
- Encoded, 011010000001001011011000111 contains 27 bits for a compression factor of $1 - 27/80 = 66.25\%$, plus overhead:

Char	Code	Char	Code	Char	Code	Char	Code
h	011	l	00	w	0101	d	111
e	0100	o	10	r	110		