

the cat in the hat sat on the mat

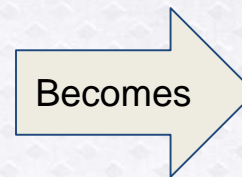
Method: Build the tree from the bottom up

Character	Count
T	7
H	4
E	3
C	1
A	4
I	1
N	2
S	1
O	1
M	1
SPACE	8

- Find 2 rows (sub-trees) with lowest frequency and join

the cat in the hat sat on the mat

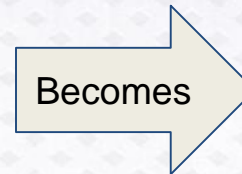
Character	Count
T	7
H	4
E	3
C	1
A	4
I	1
N	2
S	1
O	1
M	1
SPACE	8



Character	Count
T	7
H	4
E	3
(C + I)	2
A	4
N	2
S	1
O	1
M	1
SPACE	8

the cat in the hat sat on the mat (total:33)

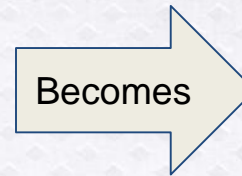
Character	Count
T	7
H	4
E	3
(C + I)	2
A	4
N	2
S	1
O	1
M	1
SPACE	8



Character	Count
T	7
H	4
E	3
(C + I)	2
A	4
N	2
(S + O)	2
M	1
SPACE	8

the cat in the hat sat on the mat (total:33)

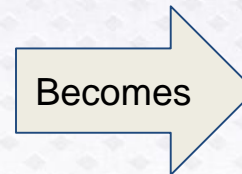
Character	Count
T	7
H	4
E	3
(C + I)	2
A	4
N	2
(S + O)	2
M	1
SPACE	8



Character	Count
T	7
H	4
E	3
((C + I) + M)	3
A	4
N	2
(S + O)	2
SPACE	8

the cat in the hat sat on the mat (total:33)

Character	Count
T	7
H	4
E	3
((C + I) + M)	3
A	4
N	2
(S + O)	2
SPACE	8



Character	Count
T	7
H	4
E	3
((C + I) + M)	3
A	4
(N + (S + O))	4
SPACE	8

the cat in the hat sat on the mat (total:33)

Character	Count
T	7
H	4
E	3
((C + I) + M)	3
A	4
(N + (S + O))	4
SPACE	8

Becomes

Character	Count
T	7
H	4
(E + ((C + I) + M))	6
A	4
(N + (S + O))	4
SPACE	8

the cat in the hat sat on the mat (total:33)

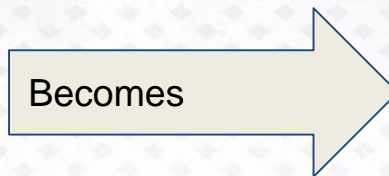
Character	Count
T	7
H	4
(E + ((C + I) + M))	6
A	4
(N + (S + O))	4
SPACE	8

Becomes

Character	Count
T	7
(H + A)	8
(E + ((C + I) + M))	6
(N + (S + O))	4
SPACE	8

the cat in the hat sat on the mat (total:33)

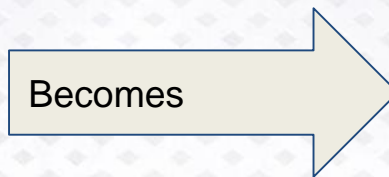
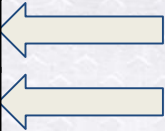
Character	Count
T	7
(H + A)	8
(E + ((C + I) + M))	6
(N + (S + O))	4
SPACE	8



Character	Count
T	7
(H + A)	8
((E + ((C + I) + M)) + (N + (S + O)))	10
SPACE	8

the cat in the hat sat on the mat (total:33)

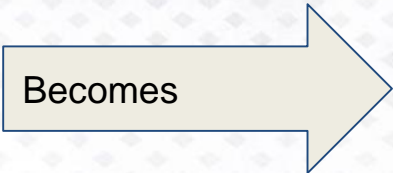
Character	Count
T	7
(H + A)	8
((E + ((C + I) + M)) + (N + (S + O)))	10
SPACE	8



Character	Count
(T + (H + A))	15
((E + ((C + I) + M)) + (N + (S + O)))	10
SPACE	8

the cat in the hat sat on the mat (total:33)

Character	Count
(T + (H + A))	15
((E + ((C + I) + M)) + (N + (S + O)))	10
SPACE	8

 Becomes	Character	Count
	(T + (H + A))	15
	((E + ((C + I) + M)) + (N + (S + O))) + SPACE)	18

the cat in the hat sat on the mat (total:33)

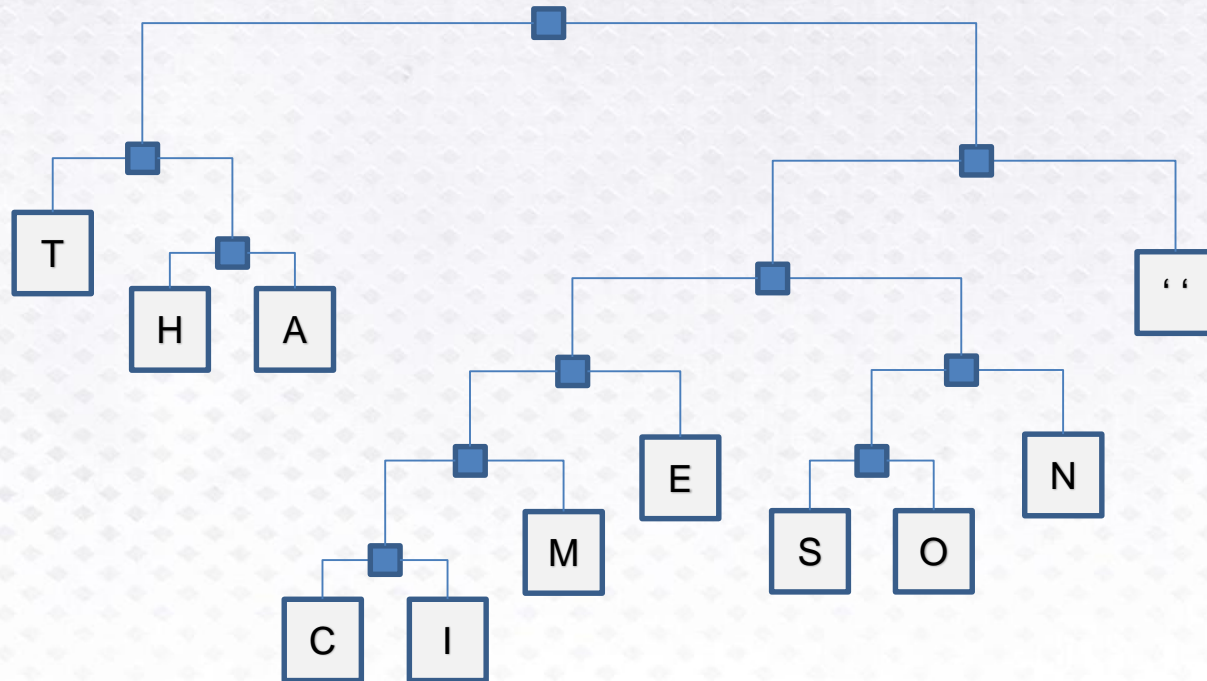
Character	Count
(T + (H + A))	15
(((E + ((C + I) + M)) + (N + (S + O))) + SPACE)	18

Becomes

Character	Count
((T + (H + A)) + (((E + ((C + I) + M)) + (N + (S + O))) + SPACE))	33

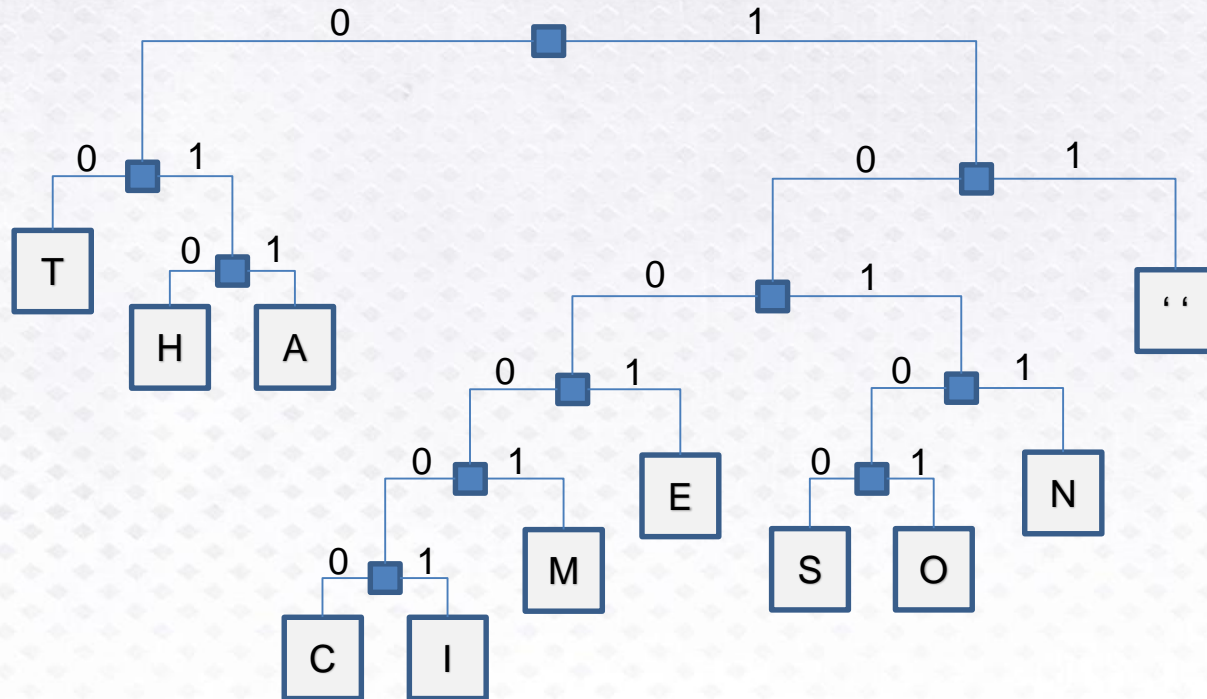
As a tree:

Character	Count
$((T + (H + A)) + (((E + ((C + I) + M)) + (N + (S + O)))) + \text{SPACE}))$	33



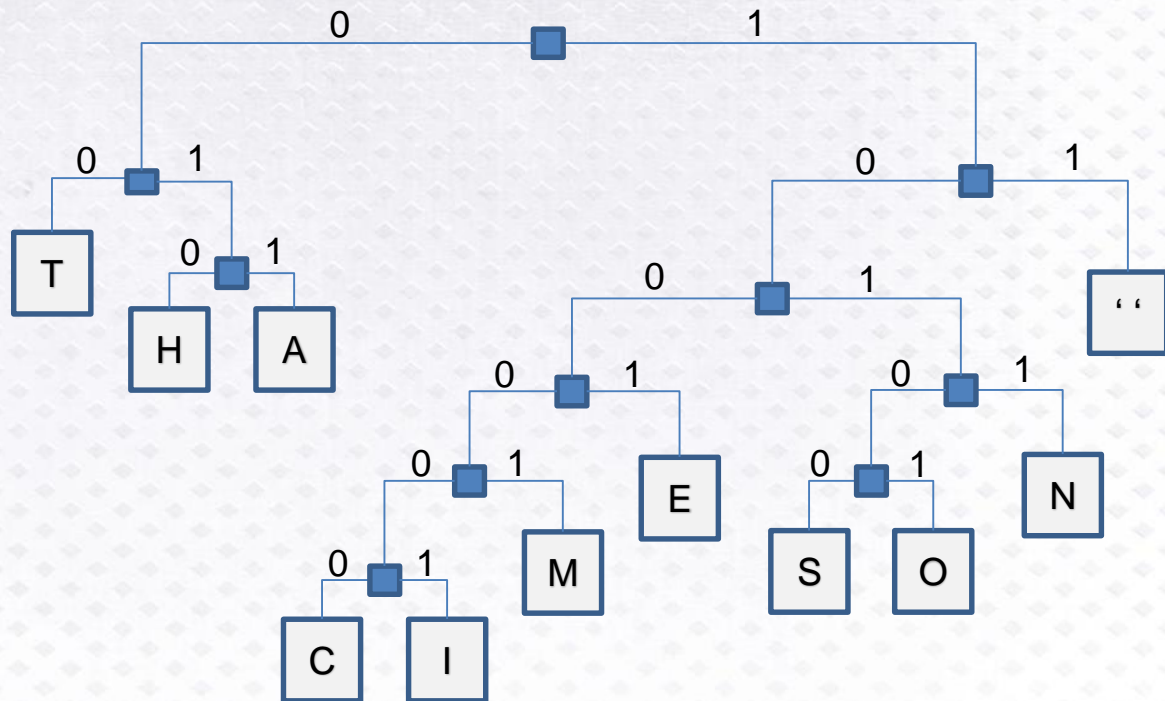
Label the edges:

Character	Count
$((T + (H + A)) + (((E + ((C + I) + M)) + (N + (S + O)))) + \text{SPACE}))$	33



Encode the characters:

Char	Count	Encoding
T	7	00
H	4	010
E	3	1001
C	1	100000
A	4	011
I	1	100001
N	2	1011
S	1	10100
O	1	10101
M	1	10001
SPACE	8	11



Encode the message:

- Original message:

the cat in the hat sat on the mat

- Becomes:

00010100111100000011001110000110111100010100111
01001100111010001100111010110111100010100111100
0101100

- And decode back to check:

The cat in the hat sat on the mat

- Presuming 8-bit ASCII:
Original Length: $33 \times 8 = 264$ bits
Encoded Length: 101bits
Ratio: $101/264 = 0.383$

Char	Count	Encoding
T	7	00
H	4	010
E	3	1001
C	1	100000
A	4	011
I	1	100001
N	2	1011
S	1	10100
O	1	10101
M	1	10001
SPACE	8	11