MC Compression Assignment

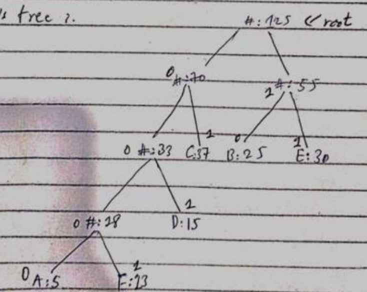
Q:1:- Construct a Huffman tree and provide the encoded bit sequence of each characters in the table.

Ans:

Table:

Character	Frequency		Character	Frequency
A	05		A	05
B	25	after sorting	F	13
C	37	⟹	D	15
D	15		B	25
E	30		E	30
F	13		C	37

Huffman's tree :



Bit sequence of each character

A = 0000
B = 10
C = 01
D = 001
E = 11
F = 0001

Q:2:- Use Huffman encoding to compress the following set of Characters

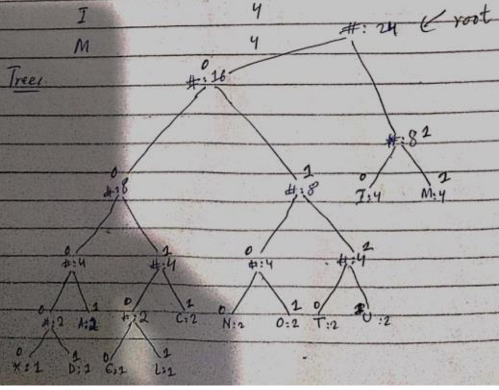
MULTIMEDIA COMMUNICATION

Ans.

sorted table of the given Characters (acc to frequency)

Character	Frequency
*(space)	1
D	1
E	1
L	1
A	2
C	2
N	2
O	2
T	2
U	2
I	4
M	4

Tree:



Bit sequence of each character.

*(space) = 00000 = 5 bits total
D = 00001 = 5 bits
E = 00100 = 5 "
L = 00101 = 5 "
A = 0001 = 4 "
C = 0011 = 4 "
N = 0100 = 4 "
O = 0101 = 4 "
T = 0110 = 4 "
U = 0111 = 4 "
I = 10 = 2 bits
M = 11 = 2 bits

Encoded output :

MULTIMEDIA COMMUNICATION =

110111 00101 0110 10 11 00100 00001 10 0001 00000 0011 0101
1111 0111 0100 10 0011 0001 0110 10 0101 0100

memory in bits required to store before compression :

MULTIMEDIA COMMUNICATION = total count of characters = 24

and we know 1 character takes one byte = 8 bits

So total bits = 24×8 = 192

Memory required before compression = 192 bits

memory required to store after compression:

MULTIMEDIA COMMUNICATION ⟹ ~~A×2×4~~ ~~(A×2×4)~~

total bits of encoded output ⟹ 84 bits

before compression bits	after compression bits
192	84

Q:3 Use Arithmetic coding to compress and decompress word 'ENGINEERED'.

Ans:

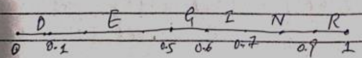
a) Compression:

let h = high, L = low, r = range before solving calculations

we must draw character range line with the help of probability distribution.

Character	Probability distribution
D	0.1
E	0.4
G	0.1
I	0.1
N	0.2
R	0.1

so the Range-Character line is



1) h = 1, L = 0, Character = 'E'

→ r = h − L = 1 − 0 = 1

r = 1

→ h = L + (r × high_range(E))

h = 0 + (1 × 0.5)

h = 0.5

→ L = L + (r × low_range(E))

L = 0 + (1 × 0.1)

L = 0.1

→ Contd

2) Char = 'N', h = 0.5, l = 0.2

r = 0.5 - 0.2 = 0.4

h = 0.2 + (0.4 × 0.9) = 0.46

l = 0.2 + (0.4 × 0.7) = 0.38

3) Char = 'E', h = 0.46, l = 0.38

→ r = 0.46 - 0.38 = 0.08

→ h = 0.38 + (0.08 × 0.6) = 0.38 + 0.048

h = 0.428

→ l = 0.38 + (0.08 × 0.5) = 0.38 + 0.04

l = 0.42

4) Char = 'I', h = 0.428, l = 0.42

r = 0.428 - 0.42 = 0.008

h = 0.42 + (0.008 × 0.7) = 0.4256

l = 0.42 + (0.008 × 0.1) = 0.4248

5) Char = 'N', h = 0.4256, l = 0.4248

r = 0.4256 - 0.4248 = 0.0008

h = 0.4248 + (0.0008 × 0.9) = 0.42552

l = 0.4248 + (0.0008 × 0.7) = 0.42536

6) Char = 'E', h = 0.42552, l = 0.42536

r = 0.42552 - 0.42536 = 0.00016

h = 0.42536 + (0.00016 × 0.3) = 0.42544

l = 0.42536 + (0.00016 × 0.1) = 0.425376

contd

Q.3 (cont'd)

Decompression of ~~the~~ code = 0.42540686 using Arithmetic Algorithm.

Ans.

From the encryption part we have the character range like as follows :-

D	E	G	I	N	R
0	0.1	0.5	0.6	0.7	0.9	1

g = symbol

again let r = range, n = input code ~~f~~, so let's begin

1) n = 0.42540686

s = 'E'

r = highrange('E') - lowrange('E')

r = 0.5 - 0.1 = 0.4

n = n - lowrange('E')/r = (0.42540686 - 0.1)/0.4

n = 0.81351424

2) n = 0.81351424

s = 'N'

r = 0.9 - 0.7 = 0.2

~~n = 0.81351424 - 0.1~~

n = (0.81351424 - 0.7)/0.2

n = 0.5675712

3) n = 0.5675712

s = 'G'

r = 0.6 - 0.5 = 0.1

n = (0.5675712 - 0.5)/0.1

n = 0.675712

Q.3 (cont'd)

7) Char = 'E', h = 0.42544, l = 0.425376

r = 0.42544 - 0.425376 = 0.000064

h = 0.425376 + (0.000064 × 0.5) = 0.425408

l = 0.425376 + (0.000064 × 0.2) = 0.42537824

8) Char = 'R', h = 0.425408, l = 0.42537824

r = 0.425408 - 0.42537824 = 0.00002256

h = 0.42537824 + (0.00002256 × 1) = 0.425408

l = 0.42537824 + (0.00002256 × 0.9) = 0.42540544

9) Char = 'E', h = 0.425408, l = 0.42540544

→ r = 0.425408 - 0.42540544 = 0.00000256

→ h = 0.42540544 + (0.00000256 × 0.5) =

h = 0.42540672

→ l = 0.42540544 + (0.00000256 × 0.2) =

l = 0.42540569 6

10) Char = 'D', h = 0.42540672, l = 0.42540569 6

→ r = 0.42540672 - 0.42540569 6

= r = 0.00001024

→ h = 0.42540569 6 + (0.00001024 × 0.2) =

h = 0.42540578 04

→ l = 0.42540569 6 + (0.00001024 × 0) =

l = 0.42540569 6

Decoded code for word 'ENGINEERED'.

4) n = 0.675712

s = 'I'

r = 0.7 - 0.6 = 0.1

n = (0.675712 - 0.6)/0.1

n = 0.75712

5) n = 0.75712

s = 'N'

r = 0.9 - 0.7 = 0.2

~~h = 0.757121~~

n = (0.75712 - 0.7)/0.2

n = 0.2856

6) n = 0.2856

s = 'E'

r = 0.5 - 0.1 = 0.4

n = (0.2856 - 0.1)/0.4

n = 0.464

7) n = 0.464

s = 'E'

r = 0.5 - 0.1 = 0.4

n = (0.464 - 0.1)/0.4

n = 0.91

8) n = 0.91

s = 'R'

r = 1 - 0.9 = 0.1

n = (0.91 - 0.9)/0.1 ⇒ n = 0.1

9) $n = 0.1$
$S = 'E'$
$r = 0.5 - 0.1 = 0.4$
$n = 0.1 - 0.1 = 0$
$n = n/r = 0$

10) $n = 0$
$S = 'D'$
$r = 0.2 - 0 = 0.2$
$n = 0 - 0 = 0$
$n = n/r = 0$

→ Here the loop will be terminated (to Arithmetic decoding), how many characters should it generate from code is also provided along with code i.e the number of iteration of loop, so our encoded text ~~output~~ had 9 characters so loop should now terminate).
now

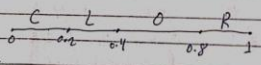
So the decompressed text we got is
ENGINEERED Ans.

Q.4: Use Arithmetic coding to compress and decompress the word 'COLOR'.

Sol:
The probability distribution of given word is

Character	Probability distribution
C	0.2
L	0.2
O	0.4
R	0.2

So the character-range line becomes



1) Compression:
Let h = high, L = low, r = range, so let's begin encoding

1) Character = 'C', h = 1, L = 0
→ $r = 1 - 0 = 1$
→ $h = 0 + (1 \times \text{highrange}('C')) = 1 \times 0.2$
$h = 0.2$
$L = 0 + (1 \times \text{lowrange}('C')) = 1 \times 0$
$L = 0$

2) Char = 'O', h = 0.2, L = 0
$r = 0.2 - 0 = 0.2$
$h = 0 + (0.2 \times 0.8) = 0.16$
$L = 0 + (0.2 \times 0.4) = 0.08$

→ Contd

3) Char = 'L', h = 0.16, L = 0.08
$r = 0.16 - 0.08 = 0.08$
$h = 0.08 + (0.08 \times 0.4) = 0.112$
$L = 0.08 + (0.08 \times 0.2) = 0.096$

4) Char = 'O', h = 0.112, L = 0.096
$r = 0.112 - 0.096 = 0.016$
$h = 0.096 + (0.016 \times 0.8) = 0.1088$
$L = 0.096 + (0.016 \times 0.4) = 0.1024$

5) Char = 'R', h = 0.1088, L = 0.1024
$r = 0.1088 - 0.1024 = 0.0064$
$h = 0.1024 + (0.0064 \times 1) = 0.1088$
$L = 0.1024 + (0.0064 \times 0.8) = 0.10752$
$L = 0.10752$
↳ encoded code for word 'COLOR'

2) Decompression:
Let h = high, L = low, r = range, S = Symbol, n = input code

1) $n = 0.10752$
$S = 'C'$
→ $r = \text{highrange}('C') - \text{lowrange}('C') = 0.2 - 0$
$r = 0.2$
→ $n = n - \text{lowrange}('C')/r = (0.10752 - 0)/0.2$
$n = 0.5376$

2) $n = 0.5376$
$S = 'O'$
$r = 0.8 - 0.4 = 0.4$
$n = (0.5376 - 0.4)/0.4$
$n = 0.344$

→ Contd

3) $n = 0.344$
$S = 'L'$
$r = 0.4 - 0.2 = 0.2$
→ $n = (0.344 - 0.2)/0.2$
$n = 0.72$

4) $n = 0.72$
$S = 'O'$
$r = 0.8 - 0.4 = 0.4$
→ $n = (0.72 - 0.4)/0.4$
$n = 0.8$

5) $n = 0.8$
$S = 'R'$
$r = 1 - 0.8 = 0.2$
$n = (0.8 - 0.8)/0.2$
$n = 0$

∴ the decompressed text for code 0.10752 is
COLOR Ans.

Q:5 Use LZW encoding to compress and decompress the string 'abcacbabaa'.

Ans:

Iteration #	String	Character	Output
1)	a	b	1
2)	b	c	2
3)	c	a	3
4)	a	c	1
5)	c	b	3
6)	b	a	2
7)	a	b	
8)	ab	a	4
9)	a	a	1
10)	for output line outside of loop		→ 1

Mapping table

Code/Index	Character(s)
1	a
2	b
3	c
4	ab
5	bc
6	ca
7	ac
8	cb
9	ba
10	aba
11	aa

So the encoded output is

123132411

⇒ contd

Q:5 contd

Decompression of code = 123132411 using LZW

Sol:

n = 123132411

Iteration #	Code	Output
1)	1	a
2)	2	b
3)	3	c
4)	1	a
5)	3	c
6)	2	b
7)	4	ab
8)	1	a
9)	1	a

Mapping Table

Code/Index	Character
1	a
2	b
3	c
4	ab
5	bc
6	ca
7	ac
8	cb
9	ba
10	aba
11	aa

So the decode text for code = 123132411 is

abcacbabaa Ans.