LECTURE # 14

CS311 - ADA

5-Dec-2023

WEDNESDAY

Revision :-

6-Dec-2023

thursday

→ Optimal Merge Solution

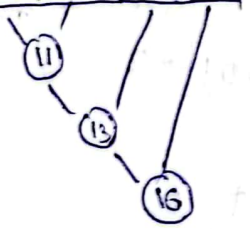
A	B	C
3	5	
8	9	
18	11	
20	16	
n	m	

① (n+m)

→ more than two lists

①

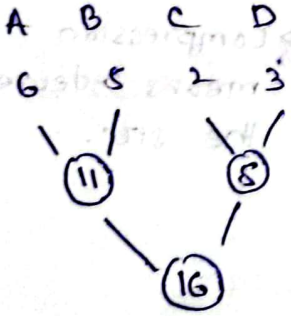
List	A	B	C	D
Sizes	6	5	2	3



Cost = 11 + 13 + 16

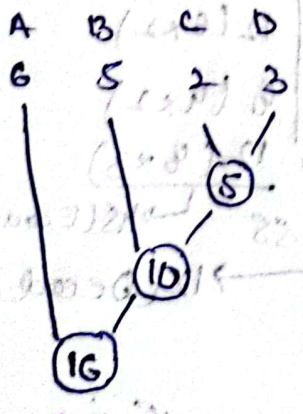
= 40

②



Cost = 11 + 5 + 16 = 32

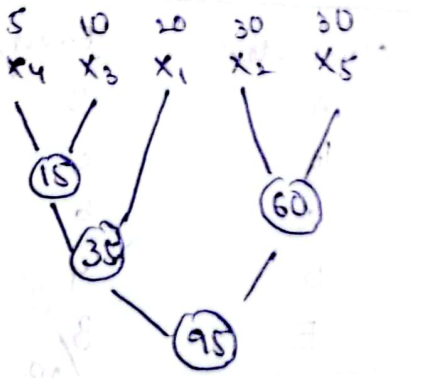
③



Cost = 16 + 10 + 5 = 31 (Optimal)

→

list	x_1	x_2	x_3	x_4	x_5
Size	20	30	10	5	30



Cost = 15 + 35 + 95 + 60

= 205

→ sort in ascending order

→ minimum value must get added

$\sum di \cdot x_i = (3\times5)+(10\times3)+(2\times20)+(2\times30)+(2\times30)$

$= 205$

di ↓ distance from root

x_i → that list member (x)

Distance From root (last sum)

---

② Huffman Coding

→ a compress technique

→ message ↔ BCCABBDDAECCBBAEDDCC

Length = 20 (x)

A → 65

B → 66

01000001 → B

Total Bits = 20 × 8 = 160 Bits

▷ greedy doesnot guarantee optimal solution.

▷ Compression means decreasing the size.

• Distance from Root

1- Encode then

2- Decode

FIXED SIZE CODING

character	Freq	Code	①	
A	3	000	001	9 (3×3)
B	5	001	10	10 (5×2)
C	6	010	11	12 (6×2)
D	4	011	01	8 (4×2)
E	8/20	100	000	12 (8×3)
5×8=40	+	15 Bits	= 55	→ 45 (Encode) → 12 (Decode)

Encode

Total Bits = 20 × 3 = 60 Bits

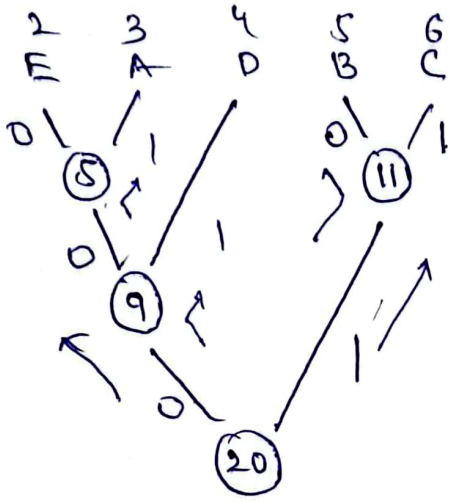
Decode

Total Bits = 55

• Total = 55 + 60 = 115 Bits

llowing Optimal merge pattern
-ive small size code to frequency
appearing character.

① Sorting in Ascending Order



Right side = 1
Left side = 0

Decode = 82 Bits (40 + 12)

Encode = 45 Bits

Total = 97 Bits