Greedy Algorithms

Job sequencing with Deadline

for every Job — Profit (Maximum Profit)
— Deadline

Job	J_1	J_2	J_3	J_4	J_5	J_6	J_7	J_8	J_9
Profit	55	65	75	60	70	50	85	68	45
Deadline	5	2	7	3	2	1	4	5	3

Maximum Profit

1) Sort the array in decending order acc to profit

	J_7	J_3	J_5	J_8	J_2	J_4	J_1	J_6	J_9
	85	75	70	68	65	60	55	50	45
Deadline	4	7	2	5	2	3	5	1	3

1	2	3	4	5	6	7
J_2	J_5	J_4	J_7	J_8	—	J_3

f 0 f1 f2 f3 f4 f5 f6

65 + 70 + 60 + 85 + 68 + 75 = 423

MaxProfit

6

min(maxDeadline − 1 , arr(i)(2) − 1)

3

Huffman coding

↳ Data compression techniques

↳ Encoding Technique

char Name	frequency	
a	= 45	45 x 8 = 360
b	= 15	15 x 8 = 120
c	= 2	2 x 8 = 16
d	= 30	30 x 8 = 240
e	= 5	5 x 8 = 40
f	= 3	3 x 8 = 24

⟹ 800 bits

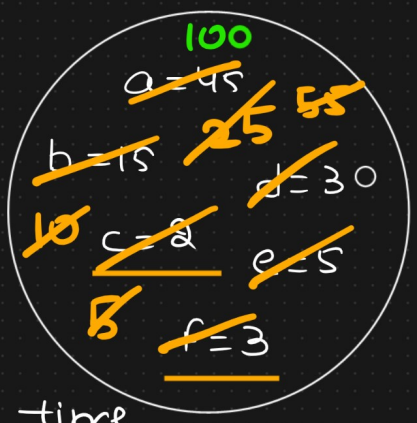
Huffman Tree

Psuedo code

Buildheap

1) create a minheap

↳ $\Theta(n)$



2) Pop two elements at a time (Two smallest element) — Delete — 2 log n

3) Insert one merged value — log n

Repeatedly

Build heap — n

$(n-1)$ 3 log n — Insert + Delete + Insert

$n + (n-1)\,3\log n = \Theta(n \log n)$

Time complexity

a - 45, b - 15, c - 2, d - 30, e - 5, f - 3

1 bit — 3 bits

a – 0, b – 101 — 2 bits

c – 10000, d – 11

45 bits

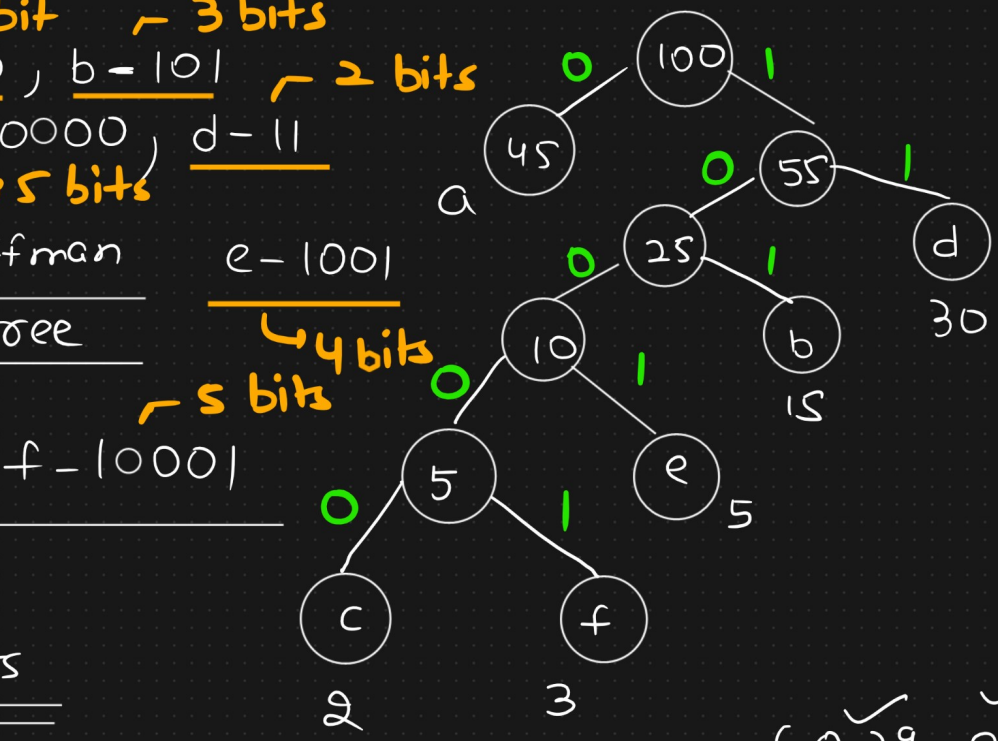
Huffman Tree

e – 1001

44 bits

5 bits

f – 10001



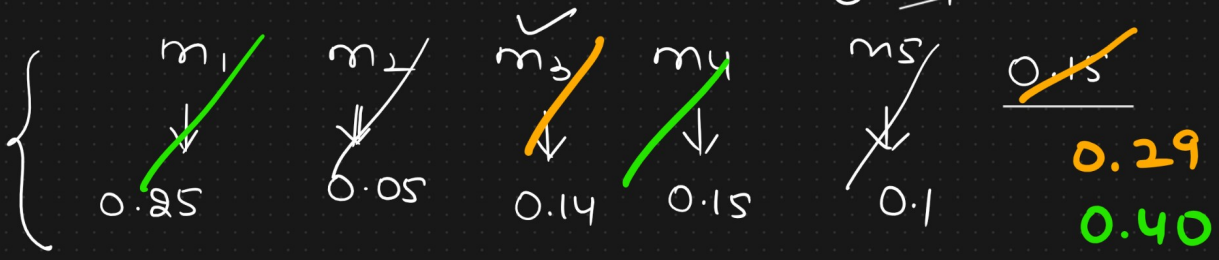
20 bits

↳ Data compression technique

0.29, 0.40

m1 – 0.25

m4 – 0.15



Huffman Tree

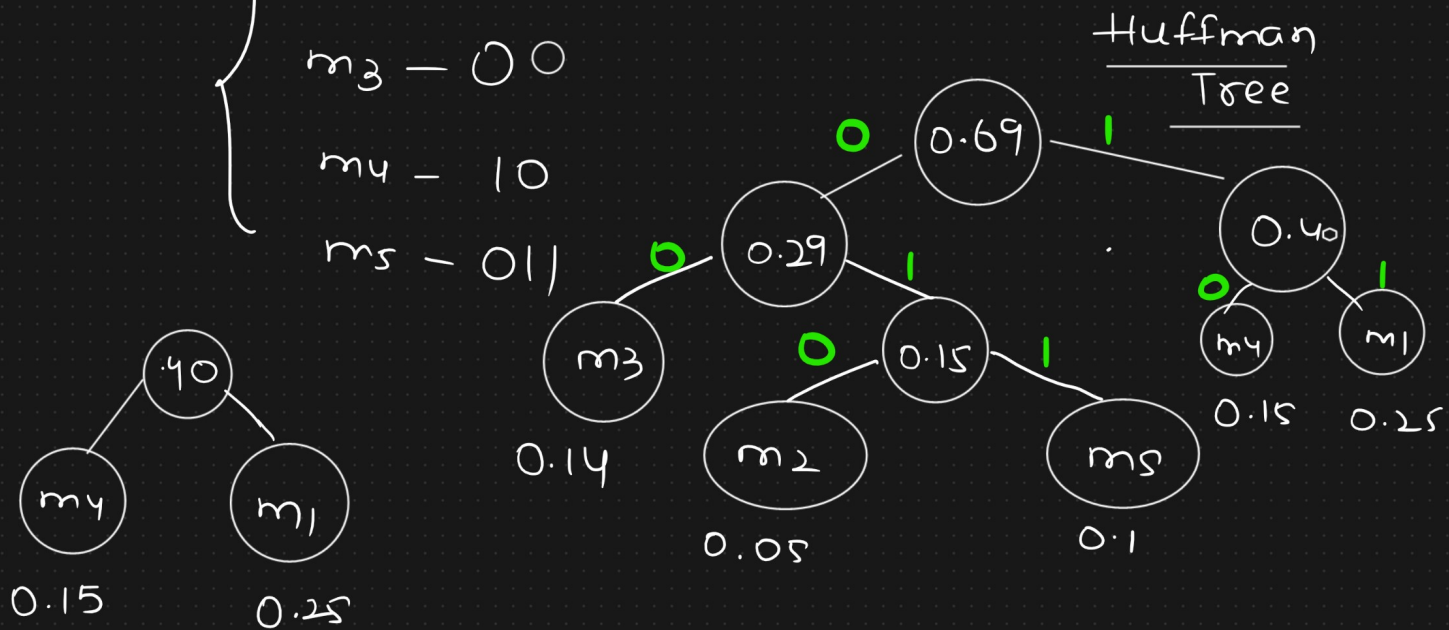
m1 – 11

m2 – 010

m3 – 00

m4 – 10

m5 – 011



Problem: 3

a — 10
e — 6
i — 15
o — 7
u — 16
s — 12
t — 20

Huffman Tree

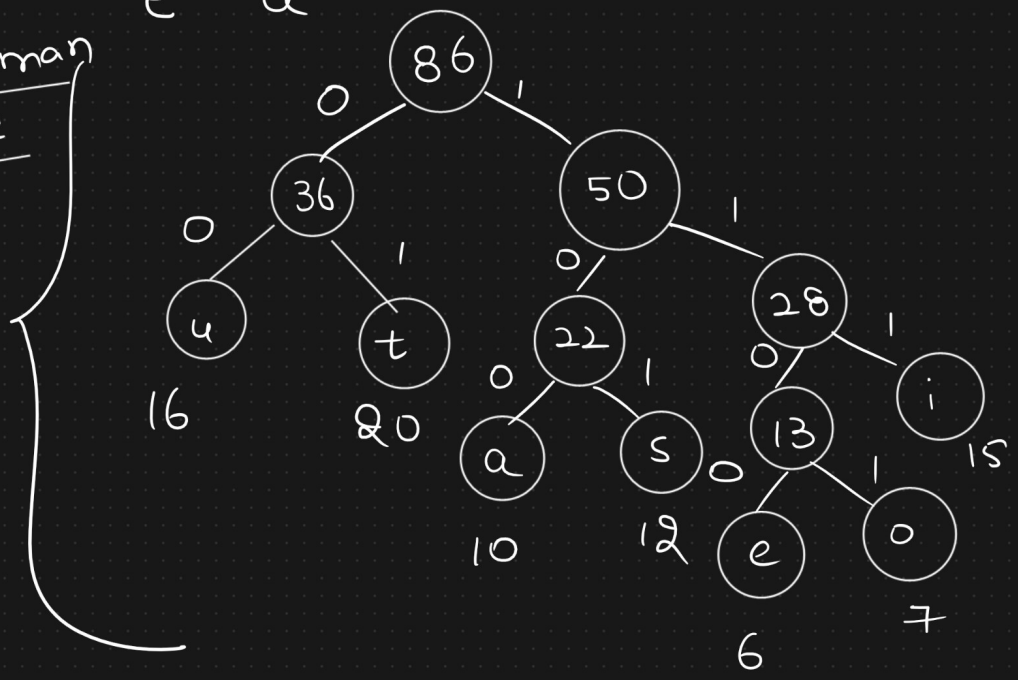
Encoded bits

a →
e →
i →
o →
u →
s →
t →

Encoded → 1100 100 00110 1101 1111100100

e a

Decode Huffman Tree



Optimal Merge Pattern

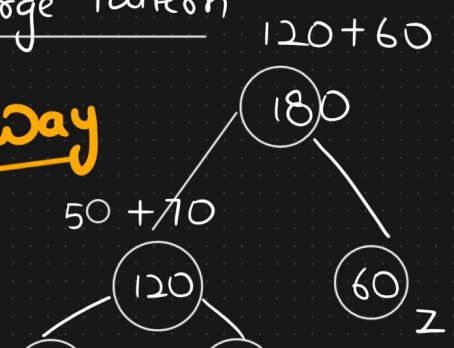
$n = 3$

$x = 50$

110, $y = 70$

$z = 60$

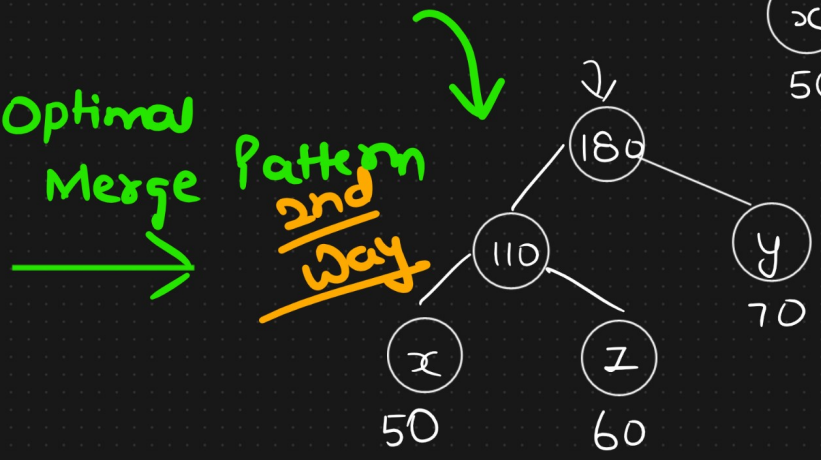
1st way



$120 + 180 = 300$

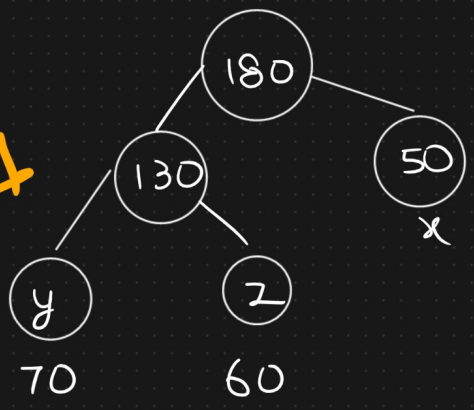
Optimal Merge Pattern

2nd way



$110 + 180 = 290$

3rd way



$130 + 180 = 310$

Time complexity $\rightarrow n + (n-1)\,3\log n$

$= \Theta(n \log n)$