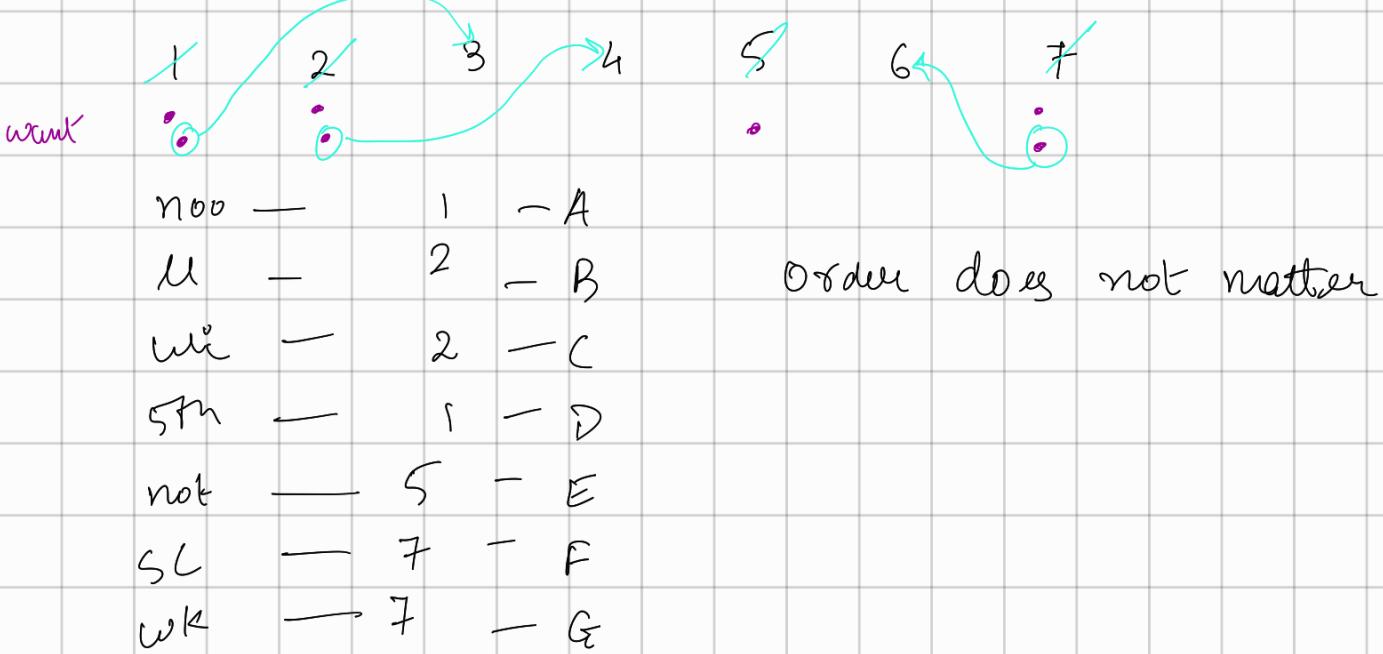


- ① Biased Standings
- ② Defense of a Kingdom
- ③ Wine trading in Georgia
- ④ Arranging amplifiers
- ⑤ Load balancing

① Biased Standings.



Approach:

put all elements in a bucket regardless of their order

$$\begin{array}{ccccccc}
 & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\
 & A & B & C & D & E & F & G \\
 \text{Buckets} & 0 + 1 + 1 + 2 + 0 + 1 + 0 = 5
 \end{array}$$

② Defense of a Kingdom

Hint: Row & Column are independent

Corner case: Area can also be along an edge

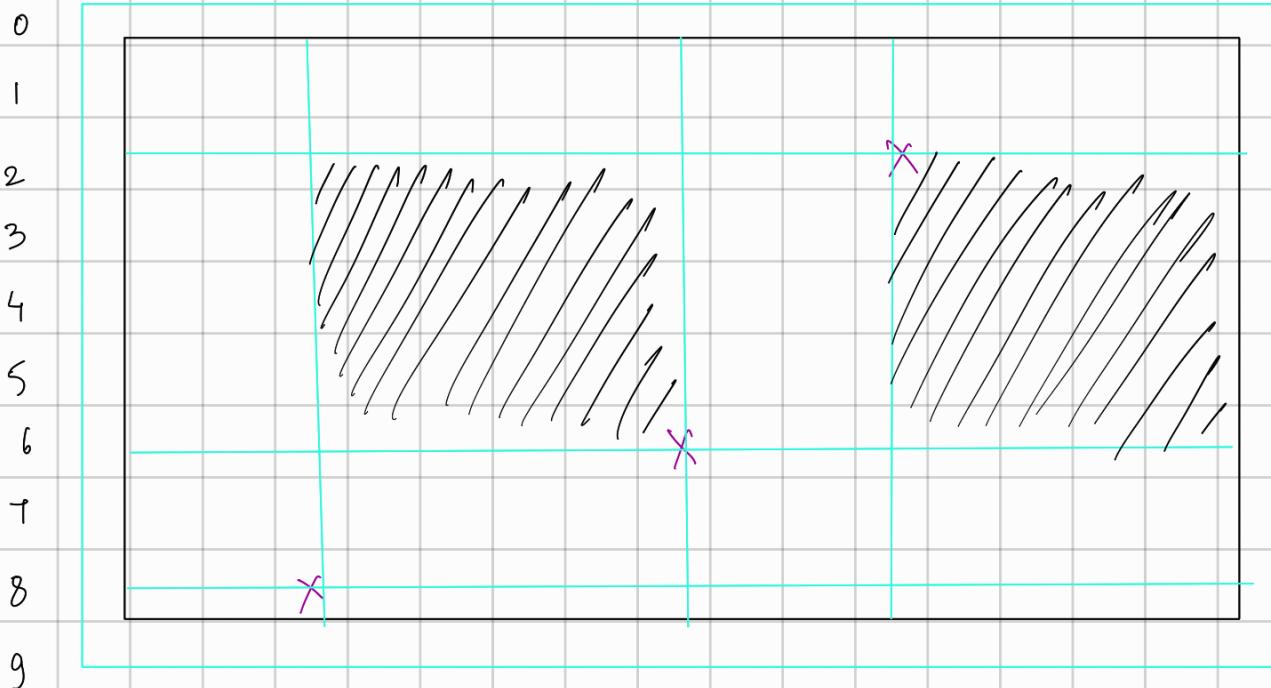
Approach?

make an array for rows & an array for columns
for 8×15

row $\{ 0, \dots, 9 \}$

col $\{ 0, \dots, 16 \}$

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16



15 8 3
3 8
11 2
8 6

$\text{ans} = \max \text{diff}$ $\max \text{diff}$
 b/w $X b/w$
 $c/dg \text{ rows}$ $c/dg \text{ cols.}$

③ Assign mice holes

Eg:

$$\begin{bmatrix} -4 & 2 & 3 \\ \downarrow 4 & \downarrow 4 & \downarrow 1 \\ 0 & -2 & 4 \end{bmatrix}$$

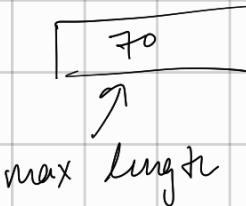
Approach: Sort both & put each mice in corresponding hole
 & max diff = min time taken,

4) Highest Pyramid

- ① i^{th} level no. of blocks $> (i-1)^{\text{th}}$ level no. of blocks
- ② i^{th} level breadth $> (i-1)^{\text{th}}$ level breadth

Eg $\{10, 20, 30, 50, 60, 70\}$

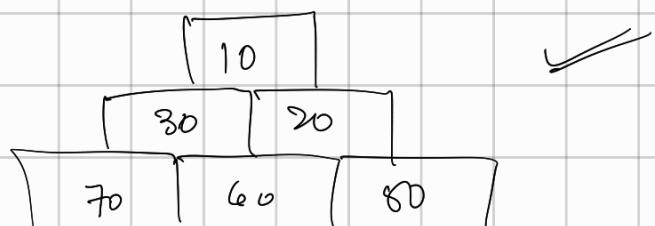
for level 1



for level 2



for level 3



So, start by placing largest & place progressively smaller next

Huffman Encoding & Decoding

Eg: B C C A B B D D A E C C B B A E D D C C

freq.

A	3
B	5
C	6
D	4
E	2

We have to send this string through an insecure medium

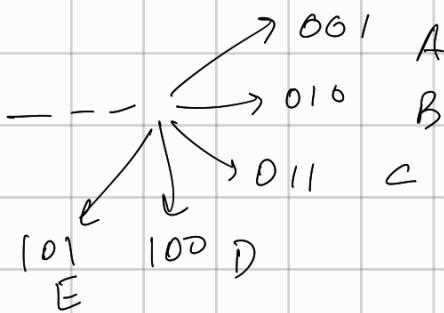
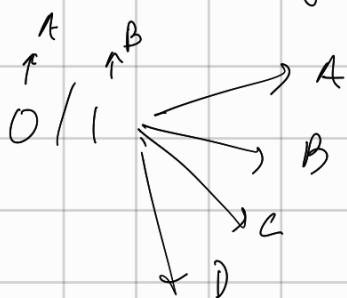
So we need to encode it & send after receiving decode it

Constraint: \rightarrow min size of encoded message

1 char \rightarrow 1 byte \rightarrow 8 bits ASCII

① Fixed size encodings

so length, 3 bits = 60 bits.



② Variable size encodings

$A \rightarrow 3$
 $B \rightarrow 5$
 $C \rightarrow 6$
 $D \rightarrow 4$
 $E \rightarrow 2$

}

Greedy: highest frequency \Rightarrow lowest bits assigned

Total bits minimum

Pre processing \rightarrow Encoding (using Binary Tree)
 ↴ Max Heap.



Eg: B C C A B B D D A E C C B B A E D D C C
 0

Binary Tree : C,6 B,5 on same level

Nodes

, child freq sum = $6+5=11$

C,6

B,5

New Nodes:

D,4

#, 15

#, 11

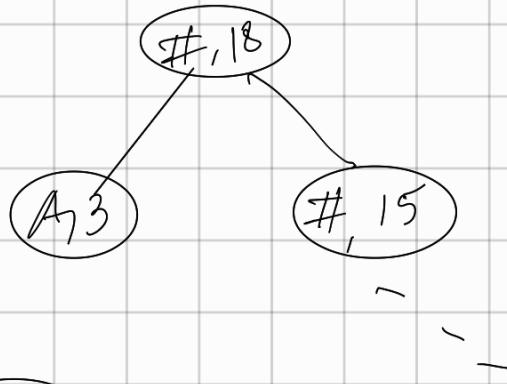
D,4

C,6

B,5

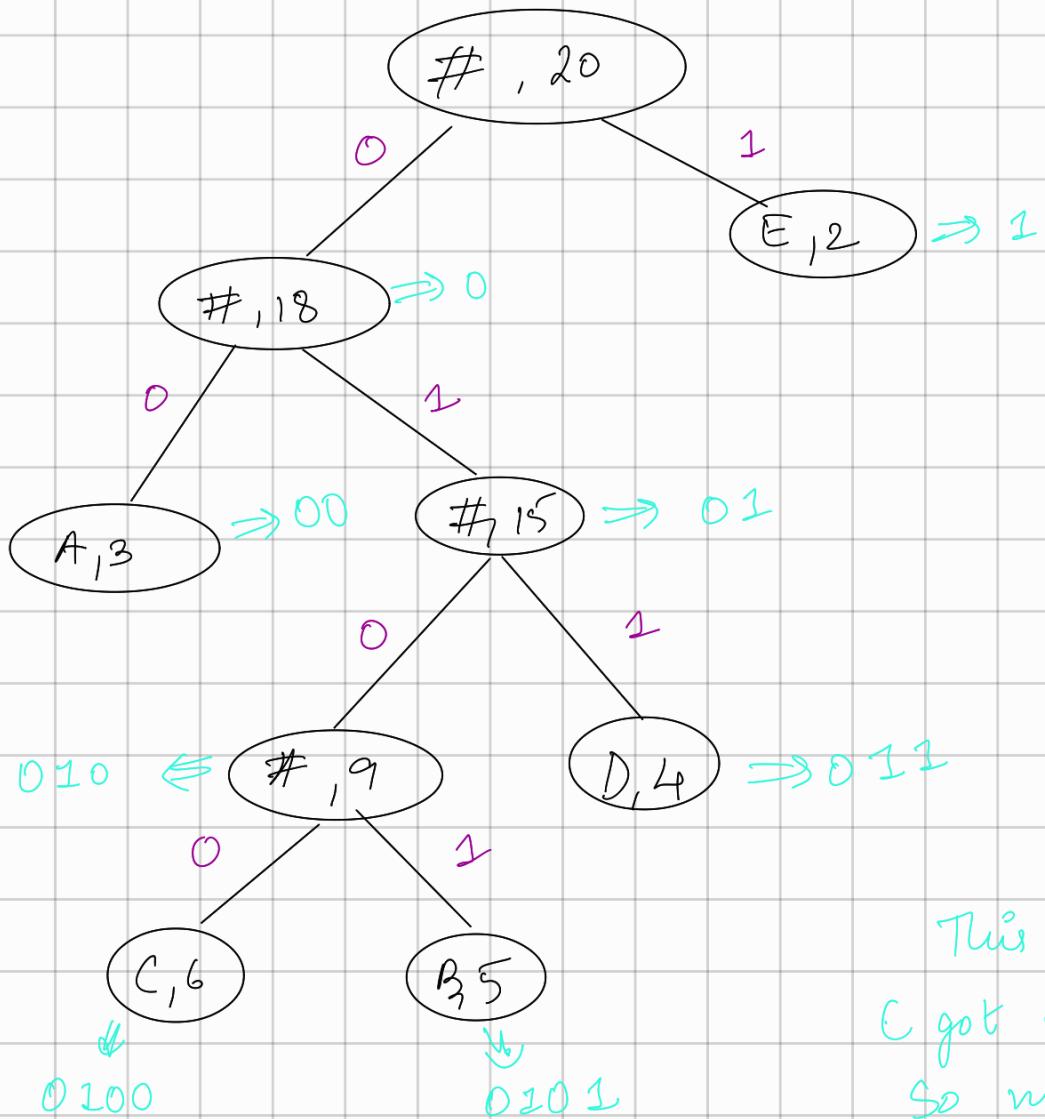
Next

A₃



Next

E₂



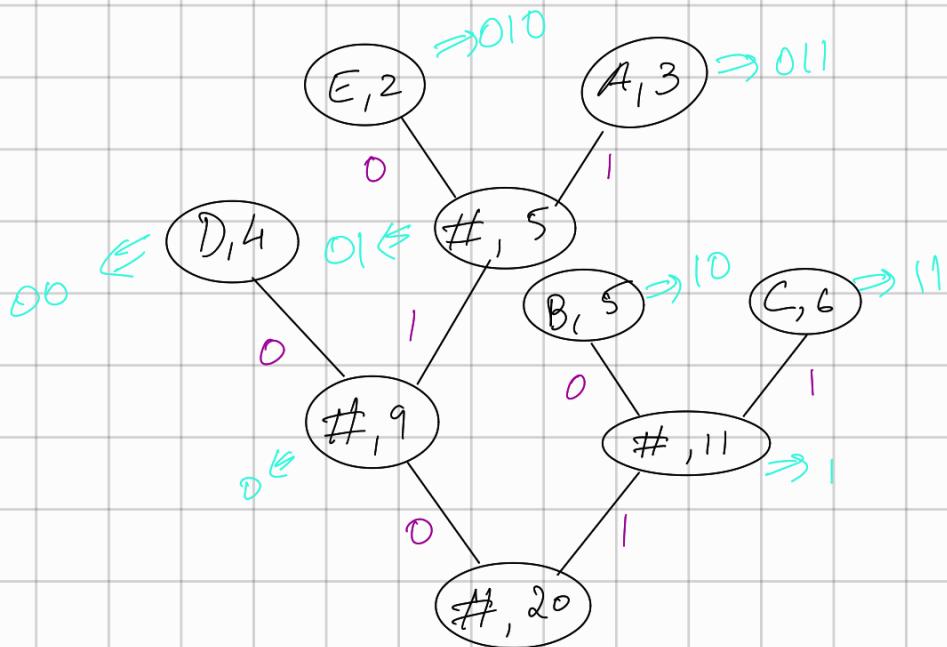
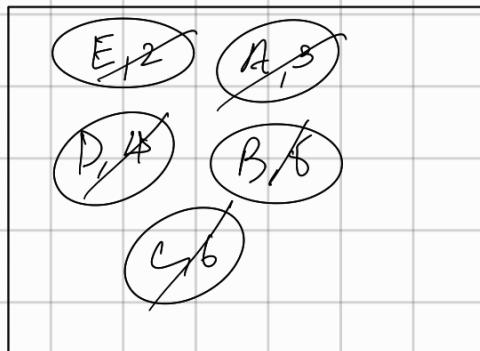
This is opposite
C got max bits
So make min heap instead

∴ Using min heap & creating bottom up → most bits

Bottom level → low freq. → highest bit

Top level → high freq. → lowest bit.

min heap



$$\begin{aligned}\text{Total Bits} &= 4 \times 2 + 3 \times 3 + 2 \times 3 + 5 \times 2 + 6 \times 2 \\ &= 8 + 9 + 6 + 10 + 12 \\ &= \underline{\underline{45}} \text{ Bits}\end{aligned}$$

To decode use 2 pointers & check for leaf node
if leaf found point & move ahead

