

Printing Subsequences

1) Exclude then Include 2) Include then Exclude

s = abc

Output

	c	abc
c		ab
b		ac
bc		a
a		bc
ac		b
ab		c
abc		

Note → If it is data structure no same subsequence store kr rhe hai

If we first Include any element in that DS.

So During calling again for Exclude, so us DS se wo element pop krke bhejna exclude ke time pr.

- If hum phle exclude krta hai, means without including any element call function, so include ke time ~~for~~ element ko push krenge on particular DS.

Array

Case I Include then Exclude

PrintSeq (index, vector & ds, arr[7, n])

where index \rightarrow index on original array

vector & ds \rightarrow use ds to store subsequences.

arr[7] \rightarrow original array

n \rightarrow size of original array

Exclude

printSeq(nums, ans, index+1, ds)

Include

int element = nums[index]

ans.push-back(element)

printSeq(nums, ans, index+1, ds)

Output of Above 2 Cases

Case I

Case II

[]

3 2 1

[1] [2]

3 2

[1, 2]

3 1

[3]

3

[1, 3]

2 1

[2, 3]

2

[1, 2, 3]

1



String

Case I → Exclude then Include

```
PrintSeq(str, n, output, index, vector)
    {
        if (i == n) {
            vector.pushback(output);
            return;
        }
    }
```

Annotations:
- str : given
- $output$: store subsequence
- $vector$: collection of all subsequence

Exclude

```
PrintSeq(str, n, output, index + 1, vector);
```

Include

```
output = output + str[i];
PrintSeq(str, n, output, index + 1, vector);
}
```


Case - 2 → Include the Exclude

Include

PrintSeq (str, n, out + str[i],
index + 1, vector)

Exclude

PrintSeq (str, n, output,
index + 1, vector)

Print subsequences with Bit Masking

vector <vector <int>> Result;
store collection of subsequences

Ex: if $n = [1, 2, 3]$ array

Its total size $2^3 = 8$
{ [], [1], [2], [3], [1, 2], [2, 3],
[1, 3], [1, 2, 3] }

Here i use also

$1 \ll n \rightarrow n$ times multiply by 2

$1 \ll 3 \rightarrow 2 \times 2 \times 2$ (three times 2 multiply)

so data size $= 2^3 = 8$

for ($i = 0$; $i < \text{pow}(2, n)$; $i++$)

vector <int> ans

for ($i = 0$; $i < n$; $j++$)

{ if ($(1 \ll j) \& i$)

Andar
waala loop
value ki
bits per
chalega
up to n



} ans.push back (numselfj)

result.push-back (ans)

Ex

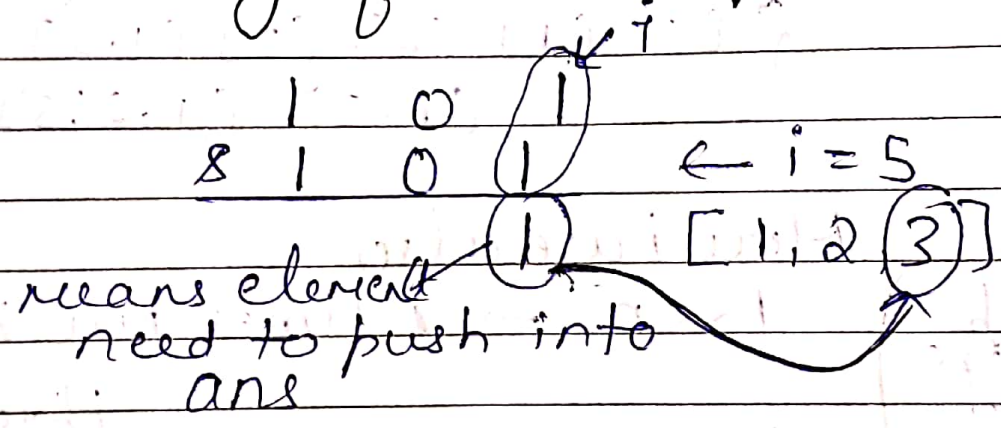
given Array

$i = 5$

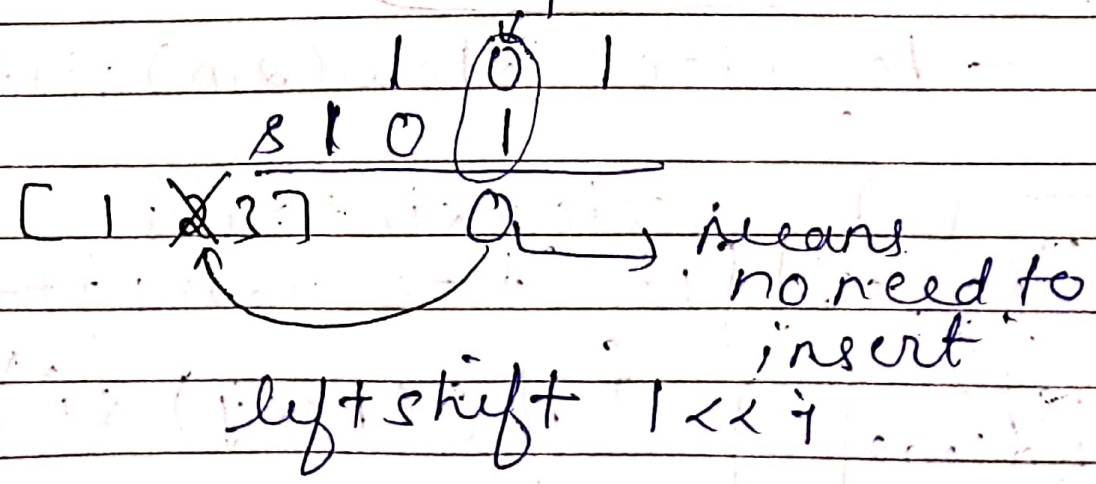
$i = 5 \rightarrow 101$

$[1, 2, 3]$

Working of Inner loop



then left shift $1 \leq j$





DATE _____

PAGE _____

