

CLASS-27

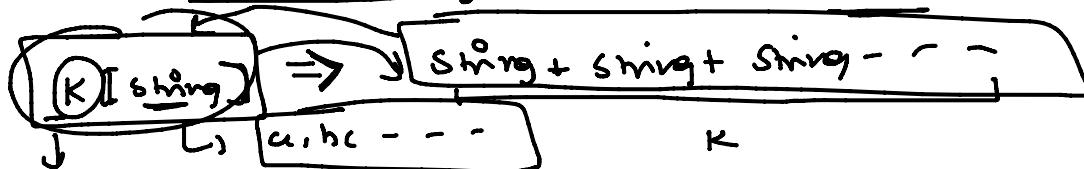
37

lecture planavit

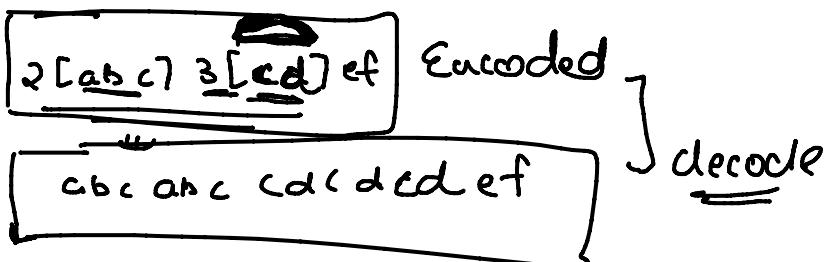
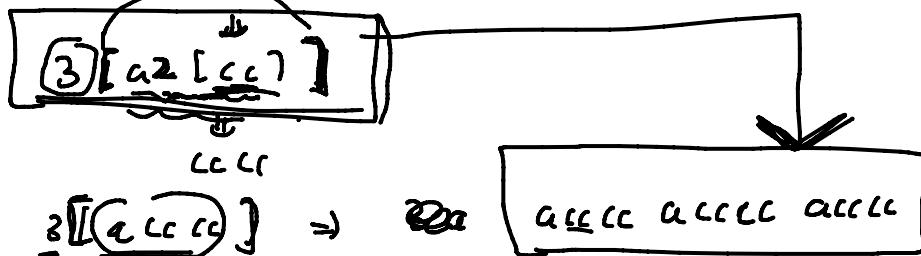
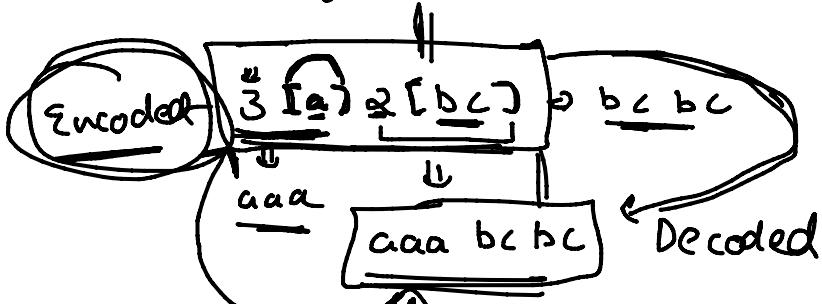


Doubt class →

June May Mid

Stack QuestionsEncoded String \Rightarrow Decoded String

integer



I see

Online judge

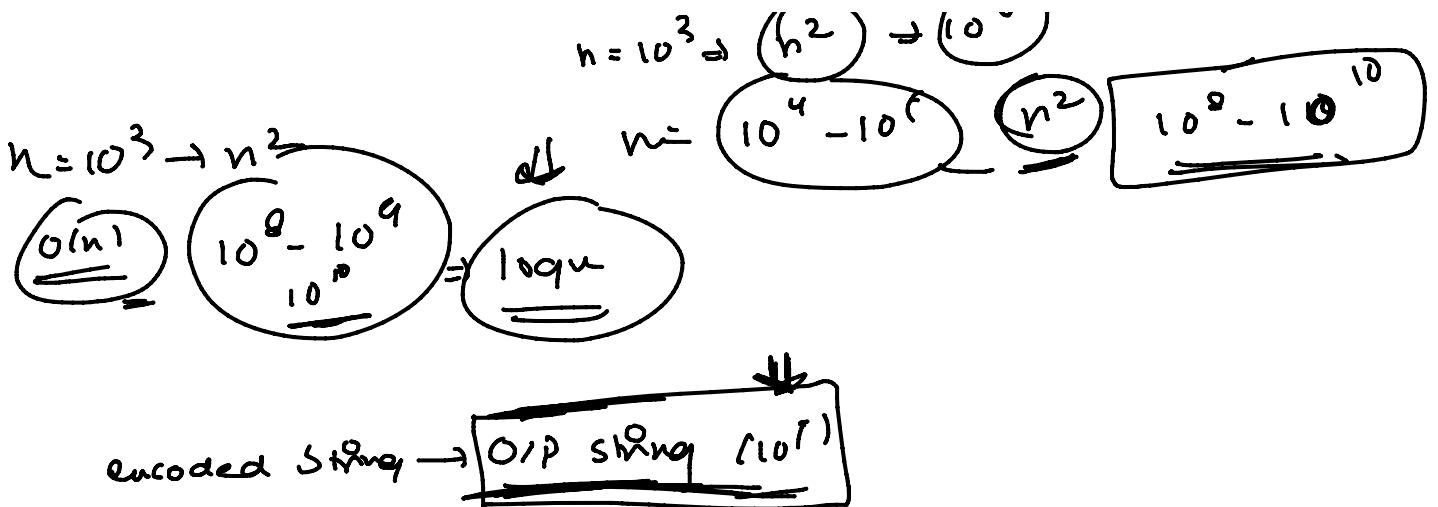
1s

 $10^7 - 10^8$

$$n = 10^3 \Rightarrow$$

$$n^2 \Rightarrow 10^6$$

 10^{10}



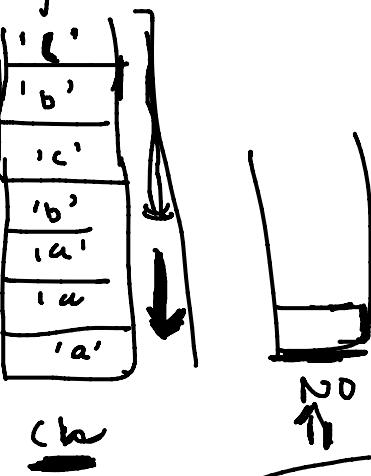
~~(3) [a] 2 [bc]~~
~~- 1 stack~~

~~H.w!~~
~~1 stack~~

~~String~~
~~String wise~~

Stack < char > ch; (String Repeat)
 Stack < int > number;

es = ~~(3) [a] 2 [bc]~~
 s[i] digit
 closed bracket encountered?



(Kisi Baar
 How many times?)
 []

String temp = "a"

~~no.top()~~

(3)

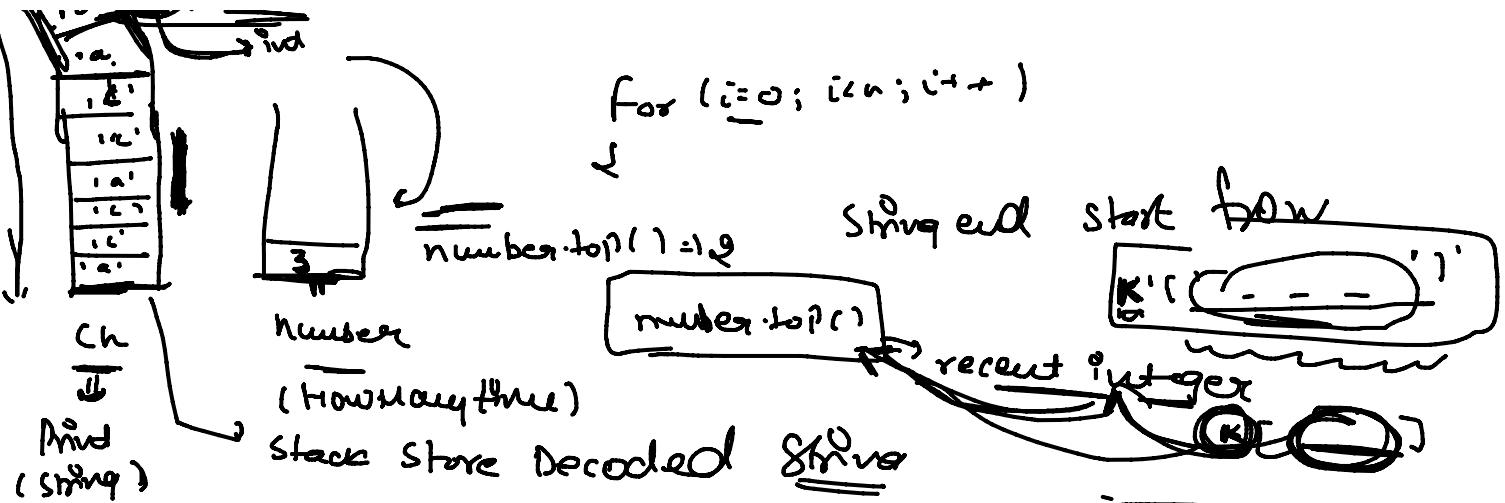
String temp = "cb"
 = "b c" \Rightarrow reverse(temp)

c b c h a a a

\Rightarrow a a a b c b c

[xyz]

~~(3) [a] 2 [bc]~~
~~- 1 stack~~
~~i = 0~~
~~acc acc acc E~~
~~L... (i=0; i<n; i++)~~



String temp = "cc" = ("cc")

temp = "ccca" = "acc"

"ccca" = "acc acc acc"

Decode String

Stack <char> ch;
Stack <int> number;

} n = length of input
string 's'

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

if (s[i] ≥ '0' and s[i] ≤ '9')

int no = 0

while (isdigit (s[i]))

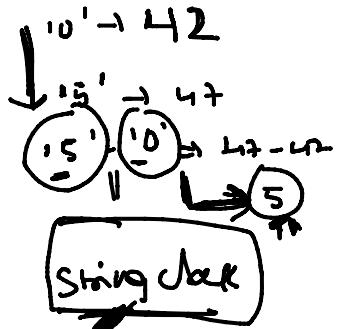
no = (no * 10) + s[i] - '0';

number.push(no);

$$no = 0 \times 10 + 2 = 2$$

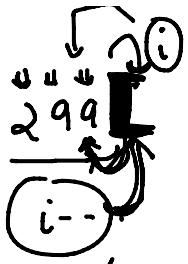
$$no = 2 \times 10 + 9 = 29$$

$$= 29 \times 10 + 0 = 290$$



else if (s[i] == ']')

(characters)



for loop went
iteration

Character

```

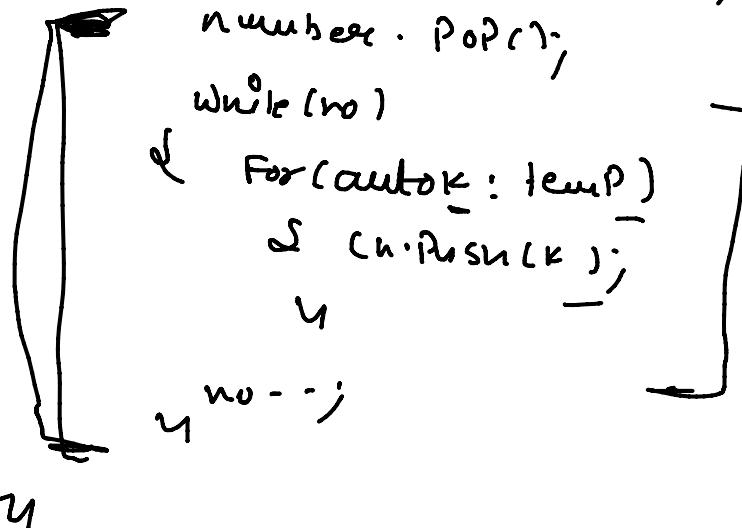
else if (s[i] == '2')
    c.push(s[i]);
}

else if (s[i] == '9')
    "get the string"
    string temp = " "j *;
    while ((c.top() == '[')
        {temp.pop_back(c.top());
        c.pop();
    }

    ch.pop();
    reverse(temp.begin(), temp.end());
    int no = number.top();
    number.pop();
    while (no)
        {for (auto k : temp)
            c.push(k);
        no--;
    }
}

```

⑦



"Final Decoded String in stack;
String temp & = " ";

```

while (cu.empty())
{
    temp.push_back(cu.top());
    cu.pop();
}

```

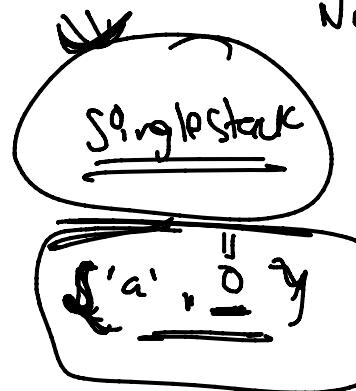
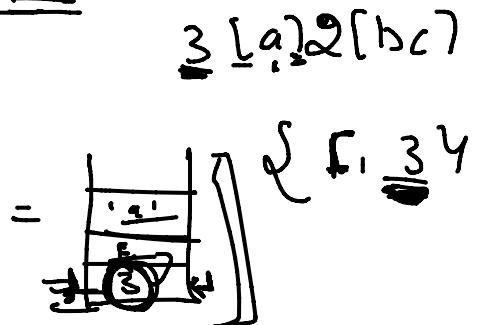


reverse (temp.begin(), temp.end());
return temp;



String temp

Single Stack



No Stack

Design Queue

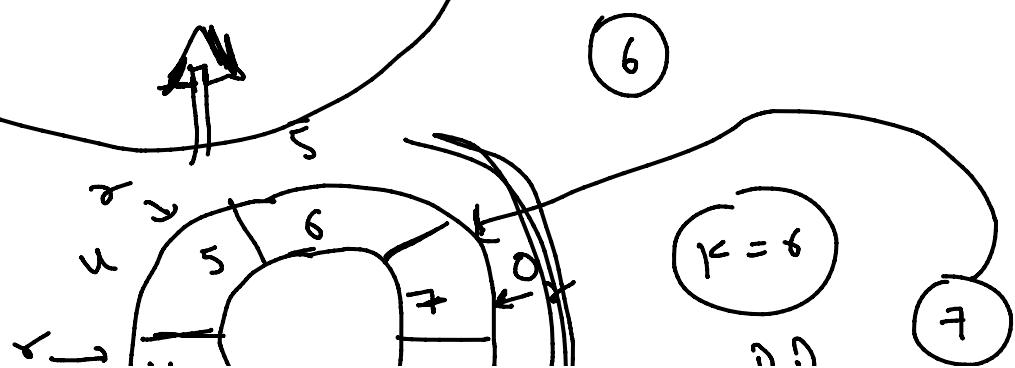
Simple queue 1D array

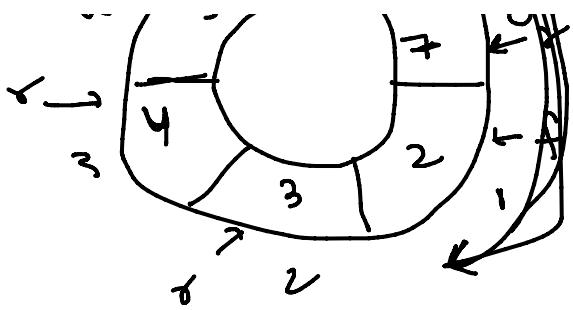


3 4 front[0]=3

using an array

front Circular Queue





Queue → Enqueue / Push

→ pop / Dequeue

$$\text{rear} = 6 \rightarrow \text{size} \rightarrow 6 + 6 = 0$$

→ front

$$\underline{\text{size}} = 6$$

$$\text{rear} = \underline{\text{front}} = \underline{5}$$



7,8

front = rear = -1

"queue is empty!"
"queue is full"

Enqueue (insert val)

↳ "is queue empty"

if (front == -1)

{ front = 0;

rear = 0;

arr[rear] = val

"is queue full?"

else if (front == 0 and rear == size - 1) or (rear == front - 1)

5

cout << "queue is full";

{ cout << "Queue is Full";

3rd step \Rightarrow Element can enter in queue

if (rear == size - 1 && front == 0)
 rear = 0; arr[rear] = value;

optional:

else {
 rear++; rear++; rear == size;
 arr[rear] = value;

int deQueue ()

{ // is queue empty

if (front == -1 and rear == -1)

{ cout << "Underflow";

}

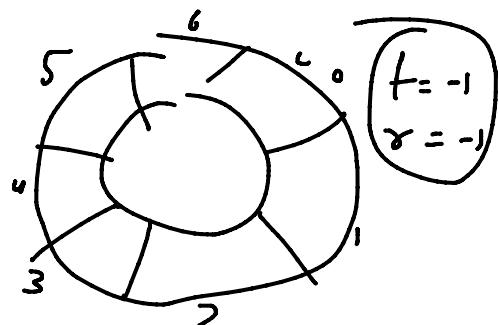
int data = arr[front];

cout << "Popped value is " << data;

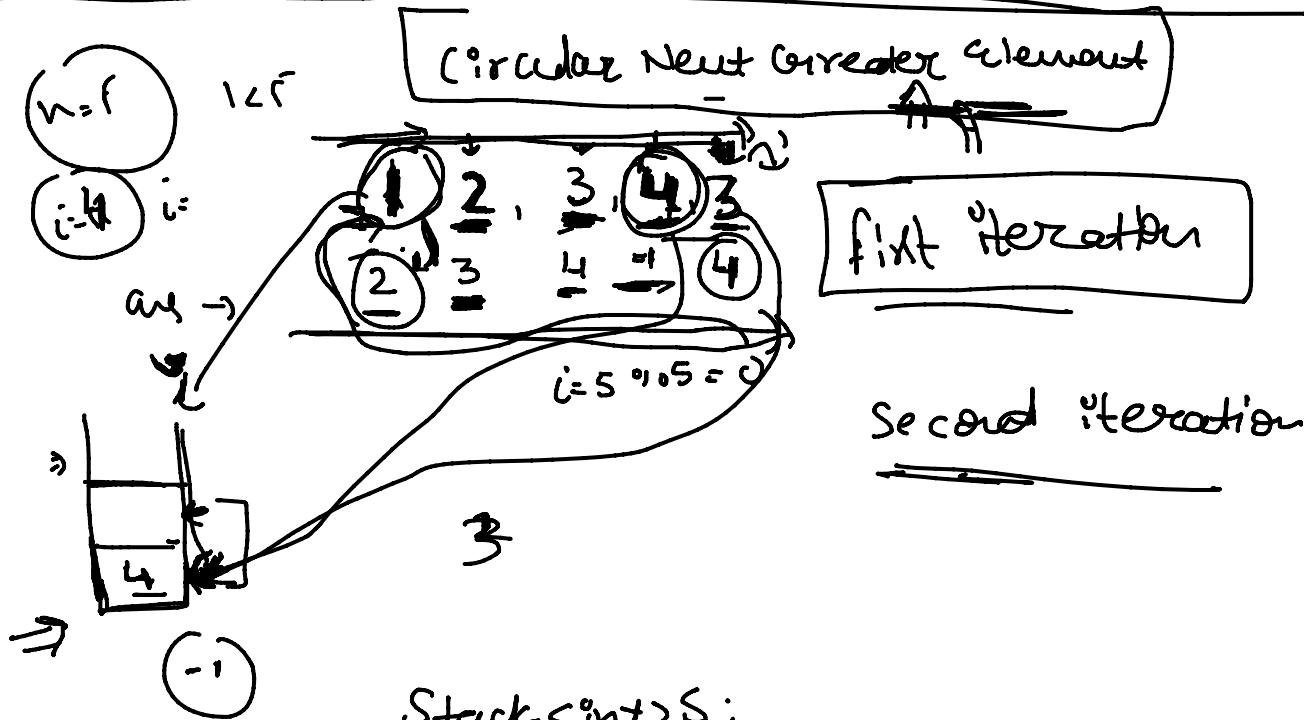
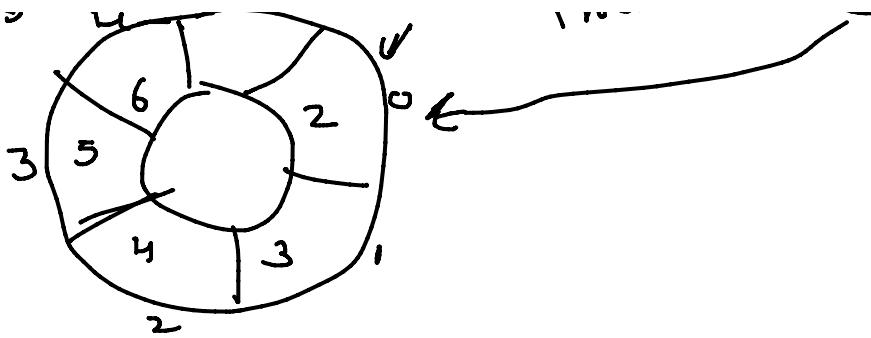
front++;

front == size;

}



$$\text{front}++ = 6 \quad 4 + 6 = 0$$



Stack<int> S;

vector<int> ans (n, -1); -1 initialize

for (i = 0; i < 2 * n; i++)

 while (!S.empty() && num[s.top()] < num[i % n])

ans[s.top()] = num[i % n];
 $\frac{S.pop();}{}$

 if (i < n)
 $\underline{s.push(i);}$

γ