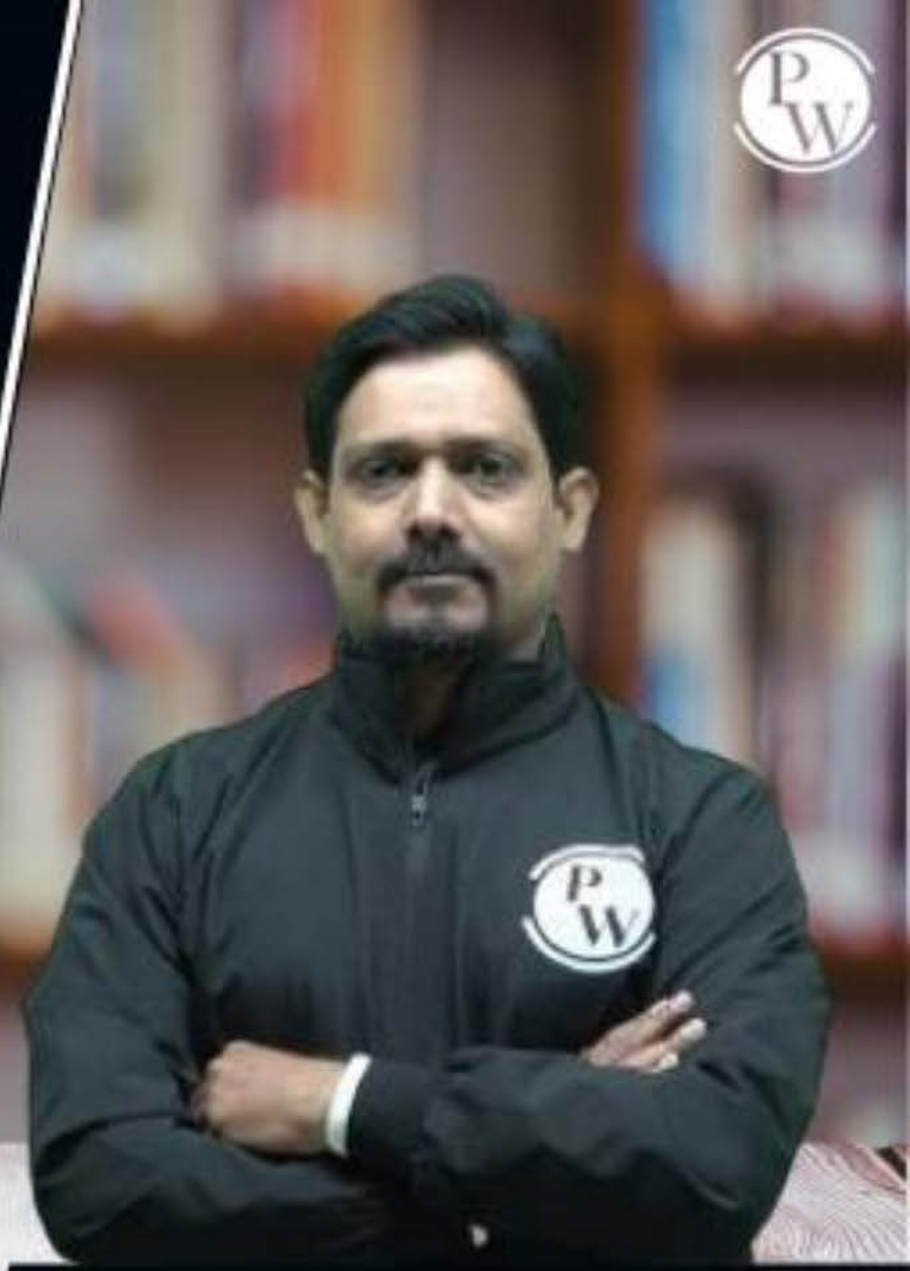
Computer Science & IT

Data Structure & Programming

Queue

Lecture No. 01



By- Abhishek Sir

Recap of Previous Lecture

- Topic
- Topic
- Topic
- Topic
- Topic

Expression

Evaluation of postfix prefix

Slide

Topics to be Covered

- Topic
- Topic
- Topic
- Topic
- Topic

Queue

linear queue

Slide

Topic : Question

#Q.

Compute the post fix equivalent of the following expression $3 \times \log (x + 1) - a / 2$

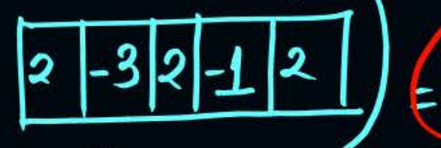
Topic : Question

A function f defined on stacks of integers satisfies the following properties. f(ϕ) = 0 and f(push (S,i)) = max (f(s), 0) +i for all stacks S and integers i. If a stack S contains the integers 2, –3, 2, –1, 2 in order from bottom to top, what is f(S)?

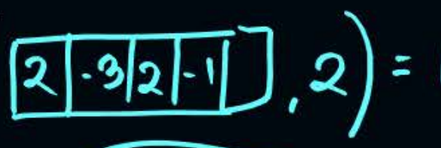
(A) 6 (B) 4

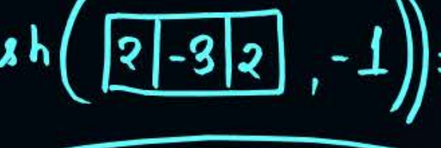
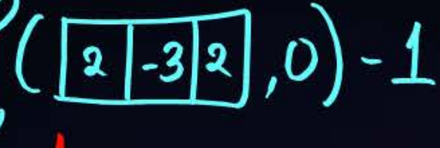
(C) 3 (D) 2

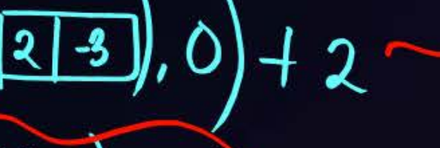
$f(\phi) = 0$, $f(push(S,i)) = \max(f(s), 0) + i$

$f($  $) = 3$

S (top)

$f($  $, 2) = \max(f($  $, 0) + 2$

$f(push($  $, -1)) = \max(f($  $, 0) - 1$

$f(push($  $, 2)) = \max(f($  $, 0) + 2$

$f(push(2), -3) = \max(f($  $, 0) - 3$

Topic : Question

A function f defined on stacks of integers satisfies the following properties. $f(\phi) = 0$ and f(push (S,i)) = max (f(s), 0) +I for all stacks S and integers i. If a stack S contains the integers 2, –3, 2, –1, 2 in order from bottom to top, what is f(S)?

(A) 6

(B) 4

(C) 3

(D) 2

Queue

end

front

Queue

Insert

∴ End

rear

Serve

deletion

front

Insert

deleted

Slide

Topic : Queue

1. Queue is a linear Data structure
2. Queue is 2 ended data structure
   front (Deletion) rear (Insertion)
3. Queue is based on logical property "first in first out"

Slide

Topic : Queue

4. Insertion operation Enqueue & Deletion operation is called dequeue

5.

Slide

Topic : Queue

global variable

```
# define Max 100
int a[Max];
int front;
int rear;
```

void init()

Initialize value of front & rear:

rear = -1

front = -1

Slide

Topic : Queue

global variable

```
# define Max 100
int a[Max];
int front;
int rear;
```

int IsEmpty()

it return 1 if queue is Empty.

Empty condition is front = -1

if front = -1 then queue is Empty

Slide

Topic : Queue

```
void init() {
    front = rear = -1
}

int IsEmpty() {
    if(front == -1)
        return 1;
    else
        return 0;
}
```

Slide

Topic : Queue

Enqueue : 1 if queue is full then return

2. Increment the rear

3. Insert the value

4. if first Insertion then Increment the front also.

Slide

Topic : Queue

```
void Enqueue(int data){
    if(rear == Max-1){
        printf("queue is full");
        return;
    }
    rear = rear + 1;
    a[rear] = data;
    if(front == -1)
        front=0
}
```

if(IsEmpty())

Slide

Topic : Queue

Dequeue : 1 if queue is Empty then return (underflow)

2. delete the element from queue (frontInde

3. if front == rear then front & rear
   initialized to -1,

4 else Increment the front

5. return the value

Slide

Topic : Queue

```
int Dequeue() {
    int data;
    if (IsEmpty()) {
        printf("queue is Empty");
        return -1;
    }
    data = a[front];
    if (front == rear)
        front = rear = -1
    else
        front = front + 1;
    return data;
}
```

Slide

Queue

front = -1, rear = -1

Enqueue(1)

front = 0, rear = 0

Enqueue(2)

rear = 0
front = 0

First Insertion
front Incremented to 0

front = 0
rear = 1

Slide

Queue

front = 0, rear = 1

front = 0

rear = 2

front = 0 Enqueue(3)

rear = 2

front = 0

rear = 3

Enqueue(4)

Slide

Queue

front = 0, rear = 3

1	2	3	4	
0	1	2	3	4

front = 0
rear = 4

1	2	3	4	5
0	1	2	3	4

Enqueue (5)

front = 0
rear = 4

1	2	3	4	5
0	1	2	3	4

Enqueue (6)

front = 0
rear = 4 (Max - 1)

1	2	3	4	5
0	1	2	3	4

overflow

Slide

Queue

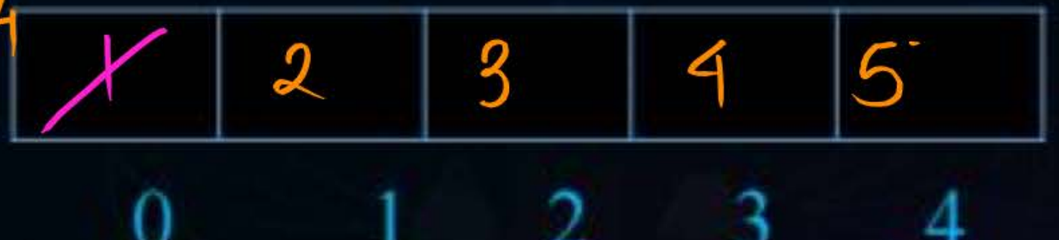
front = -1, rear = -1

queue is Empty

Dequeue ( )

front = 0, rear = 4

front = 1
rear = 4



underflow

Dequeue( )

first Delete then Increment

Slide

Queue

front(1), rear = 4

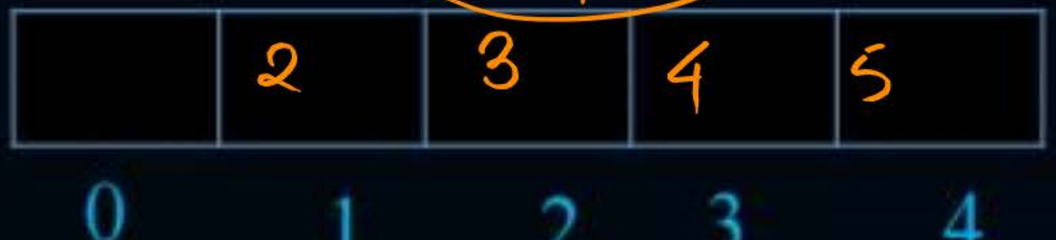
Enqueue(6) ✓

front = 1, rear = 4

Dequeue()

overflow

rear = Max - 1



front = 2

rear = 4

Slide

Queue

front = 2, rear = 4

front = 3
rear = 4

Dequeue

front = 3
rear = 4

front = 4
rear = 4

Dequeue()

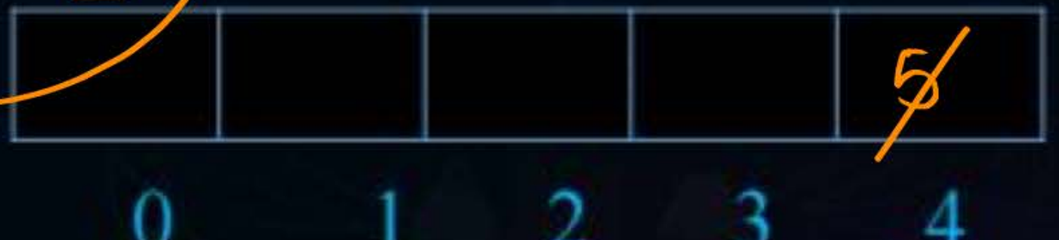
Slide

Queue

front = 4
rear = 4

Dequeue ()

front = -1
rear = -1



after deletion before Incrementing
if front & rear points to same Index
both initialized to -1

Slide

(15), Enque(60) x = dequeue(), y = dequeue

Enqueue(2), Enqueue(3), z = dequeue

alue of x+2y-z is ____

$15^2 = 225$

60

2

225 + 120 - 2

345 - 2 = 343

Topic: GATE 2021 Set-I, 1-Mark

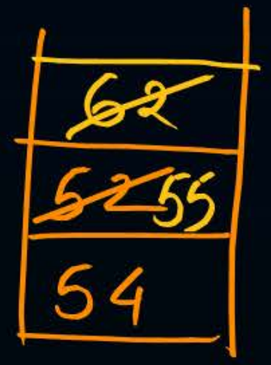
Consider the following sequence of operations on an empty stack.

```
push (54); push (52); pop (); push (55); push (62); s = pop ();
```

Consider the following sequence of operations on an empty queue.

```
enqueue (21), enqueue (24), deque (); enqueue (28); enqueue (32); q = dequeue();
```

The value of s+q is 62+24 = 86



x = pop()

x = 62

q = dequeue

q = 24

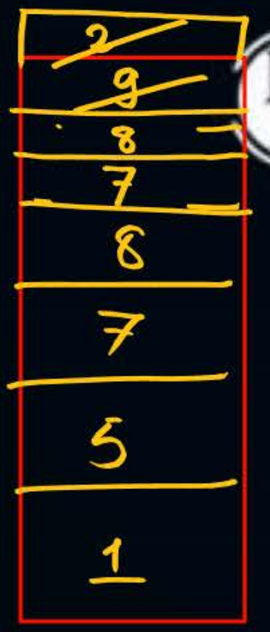
Slide

Topic: GATE 2023, 2-Marks

Consider a sequence a of elements $a_0 = 1, a_1 = 5, a_2 = 7, a_3 = 8, a_4 = 9$, and $a_5 = 2$. The following operations are performed on a stack S and a queue Q, both of which are initially empty.

I. push the elements of a from a_0 to a_5 in that order into S.
II. enqueue the elements of a from a_0 to a_5 in that order into Q.
III. pop an element from S.
IV. dequeue an element from Q.
V. pop an element from S.
VI. dequeue an element from Q.
VII. dequeue an element from Q and push the same element into S.
VIII. Repeat operation VII three times.
IX. pop an element from S.
X. pop an element from S.

The top element of S after executing the above operations is _____.



Slide

Consider the following function

```
void f() {
  int i;
  if(!isEmpty()) {
      i = dequeue()
      f();
      Enqueue(i);
  }
}
```

What operation performed by above function

(A) Leave the queue unchanged

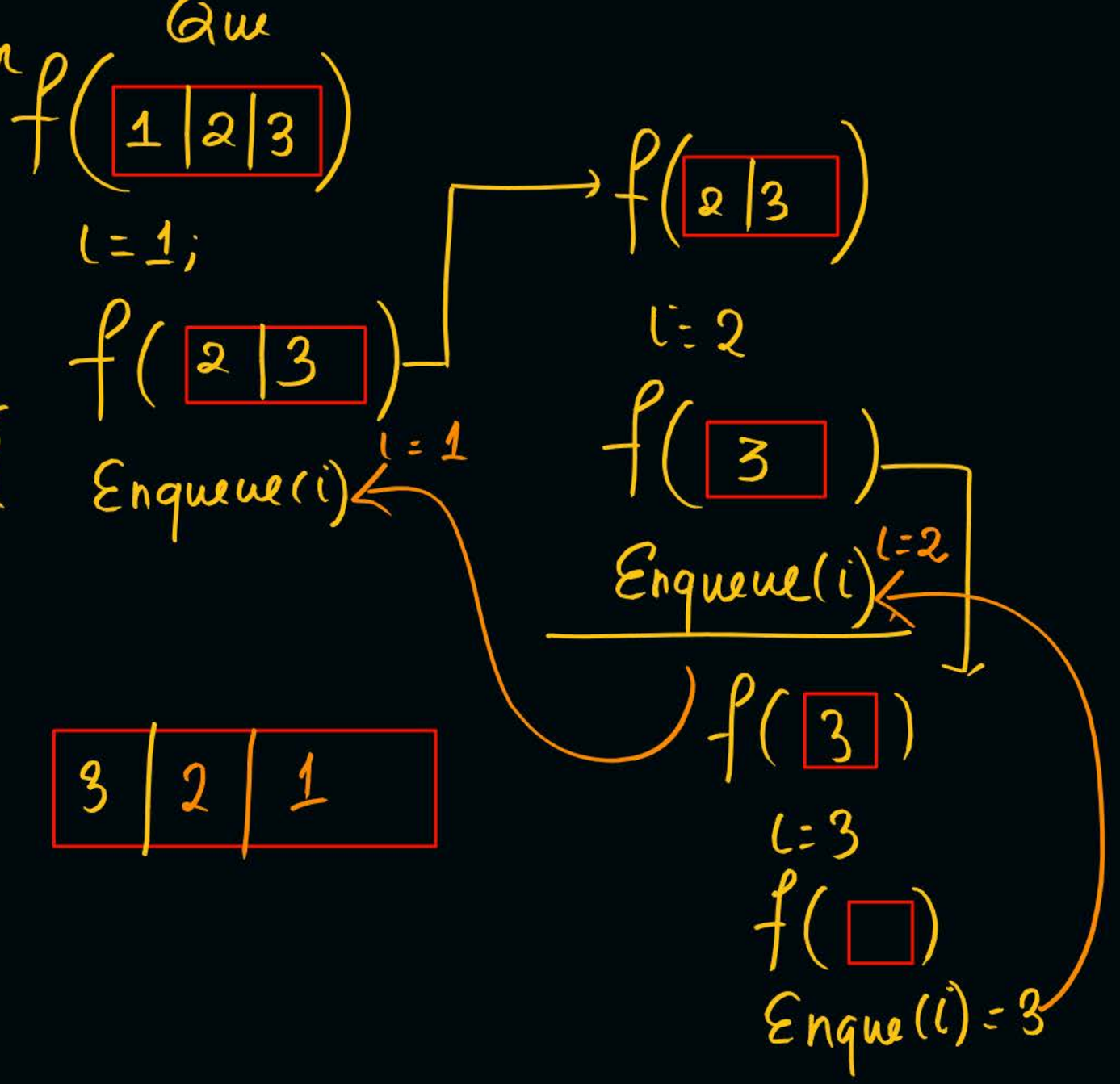
(B) Reverse order of element of the Queue

(C) Delete element from front of queue and inseot in rear.

(D) Empties the queue

Consider the following function

```
void f(){
  int i;
  if(!isEmpty()){
    i=dequeue()
    f();
    Enqueue(i);
  }
}
```



2 mins Summary

Topic

Topic

Topic

Topic

Topic

Expression

Evaluation of postfix prefix

Slide

THANK - YOU