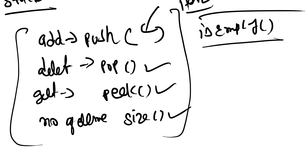
Lec-36
03/04/24 10:20 AM

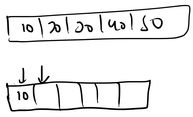
Stack and other

LIFO

Stack

data size
top delete
front





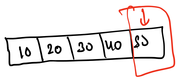
idx=2 2 3 4 → 4

```
public void push(int item) {

}
```

st.push(10);
st.push(20)
st.push(30)
st.push(40)
st.push(50)

idx--

arr[idx]=item;
idx++



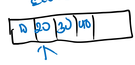
Dequeue (1)

[ii]

FIFO

add → enqueue
remove → dequeue
get → getFront
size
isEmpty
isFull

front=0
size=0



(1)

idx = (front+size)%arr.length

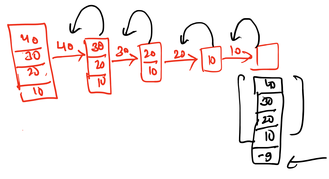
```
Public void Enqueue(intitem) {
   arr[size] = item;
     size++;
  }

  publicintDequeue() {
     intv= arr[front];
     front++;
     size--;
     return v;
  }
```

Size=2+1
front=3+1=4

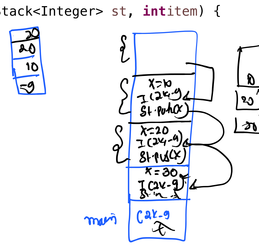
30 40 50 60

front=(front+1)%arr.length

2nd=2
3rd=3
4th=4
5th=5




```
public static void Insert(Stack<Integer> st, intitem) {
   if(st.isEmpty()) {
      st.push(item);
      return;
   }
   intx= st.pop();
   Insert(st, item);
   st.push(x);
}
```




```
Public static void Reverse(Stack<Integer> st) {
    if(st.isEmpty()) {
       return;
    }
    intx= st.pop();
    Reverse(st);
    Insert(st, x);
}
public static void Insert(Stack<Integer> st, intitem) {
    if(st.isEmpty()) {
       st.push(item);
       return;
    }
    intx= st.pop();
    Insert(st, item);
    st.push(x);
}
```