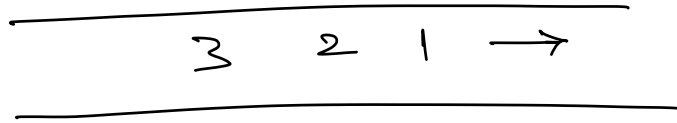


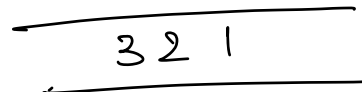
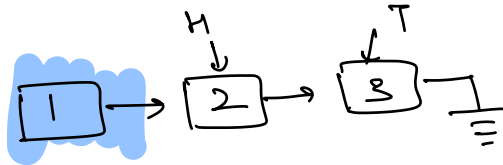
Queue Implementation (FIFO)

⇒ Using L.L



Linked List

- Add at front $\Rightarrow O(1)$
 - Add at tail $\Rightarrow O(1)$
 - Delete at front $\Rightarrow O(1)$
 - Delete at tail $\Rightarrow O(N)$ X
- } Queue

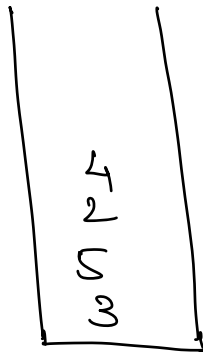


HW:- Explain internal implementation of stacks & queue in your language.

Q Given a stack, Implement a new Stack which allows us to perform getMin() funⁿ in $O(1)$ T.C

*Amazon
GS, Paytm
DMA, Unacademy*

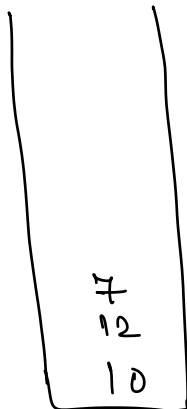
*returns the
min. of stack*



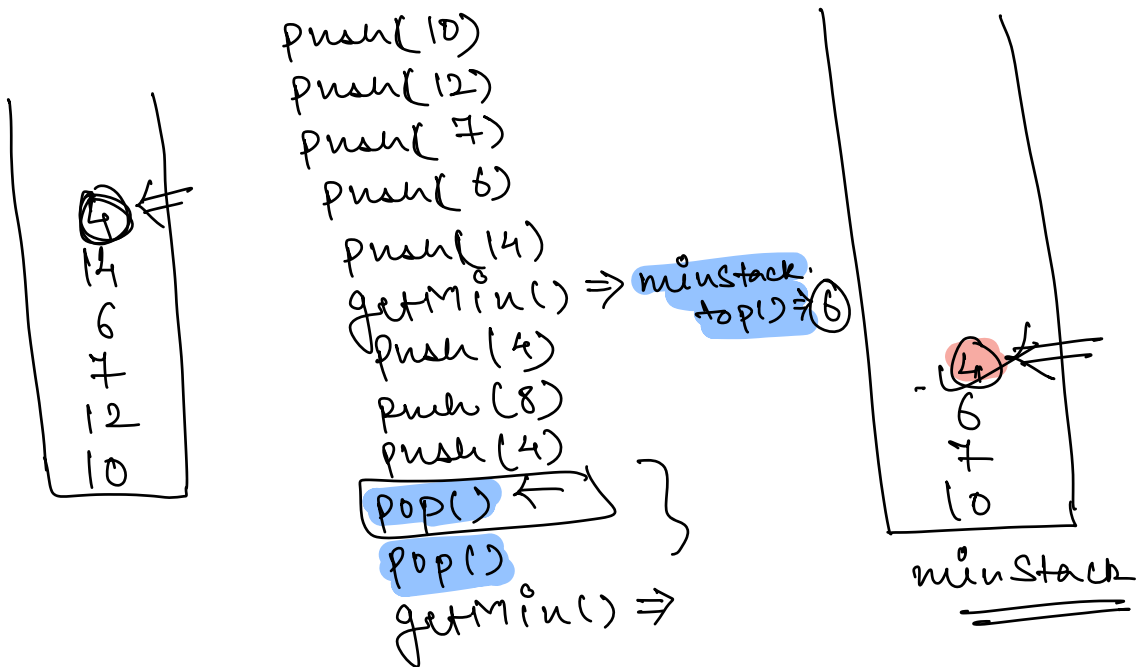
push(3)
 push(5)
 getMin() \Rightarrow 3
 push(2)
 push(4)
 getMin() \Rightarrow 2
 \hookrightarrow $O(1)$ T.C

\Rightarrow Min Variable

min = \emptyset / ∞ \neq 6



push(10)
 push(12)
 push(7)
 push(6)
 push(14)
 getMin() \Rightarrow 6 ✓
 pop()
 pop()
 getMin() \Rightarrow 6 ✗



```

Stack s1;
Stack minStack;

push(a) {
    s1.push(a)
    if( minStack.isEmpty() ||
        a <= minStack.top() ) {
        minStack.push(a)
    }
}

```

\Rightarrow $O(1)$

```

void pop() {
    ⇒ temp = st.top();
    st.pop();
    if (temp == minStack.top()) {
        minStack.pop();
    }
}

```

} ⇒ O(1)

```

int getMin() {
    return minStack.top();
}

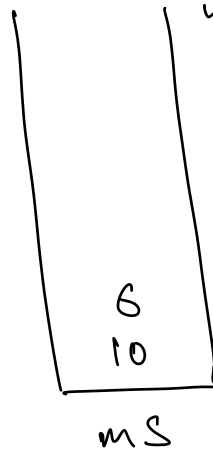
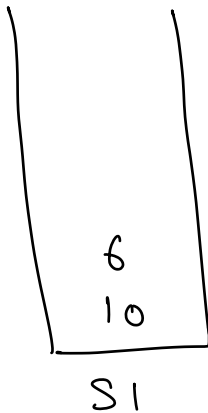
```

⇒ O(1)

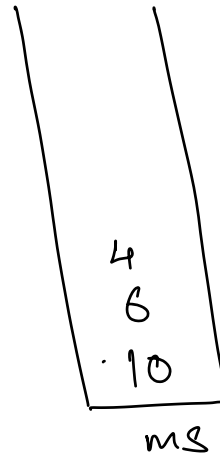
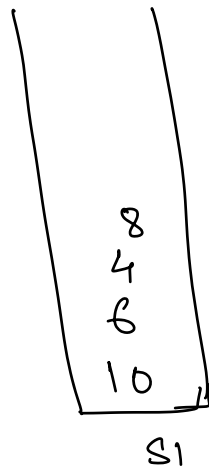
10, 6, 4, 6, getMin(), pop(), getMin(), pop()

↓
4
✓

↓
4
✓



10, 6, 4, 8, 4, pop(), pop(), pop()



$$\# \quad T(N) = 2T(N/2) + N$$

$$T(N) = 2 \left[2T(N/4) + \frac{N}{2} \right] + N$$

$$T(N) = 4T(N/4) + 2N.$$

$$= 4 \left[2T(N/8) + \frac{N}{4} \right] + 2N.$$

$$= 8T(N/8) + 3N.$$

$$= 8 \left[2T\left(\frac{N}{16}\right) + \frac{N}{8} \right] + 3N.$$

$$= 16T(N/16) + 4N.$$

After (k) steps:-

$$T(N) = 2^k T\left(\frac{N}{2^k}\right) + k \cdot N.$$

$$\frac{N}{2^k} = 1 \Rightarrow N = 2^k$$

$$\log_2 N = (k)$$

$$N \rightarrow \frac{N}{2} \rightarrow \frac{N}{4} \rightarrow \frac{N}{8} \rightarrow \dots \quad (1)$$

$\underbrace{\hspace{10em}}_{\log_2 N}$

$$T(N) = 2^{\log_2 N} T(\underbrace{1}_1) + \log N \cdot N.$$

$$T(N) = N \cdot 1 + N \log N.$$

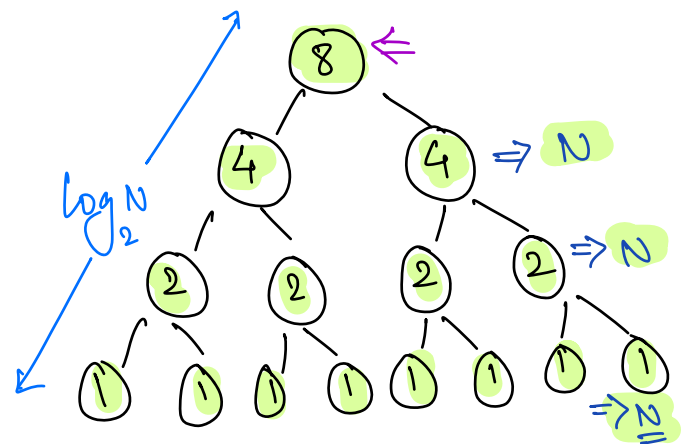
$$T.C \Rightarrow O(N \log N)$$

$$\underline{T(N) = 2T(N/2) + \underline{N}}$$

```

fun (N) {
  fun (N/2)
  fun (N/2)
  for (i  $\Rightarrow$  0 to N) {
    //
  }
}

```



$$N \log_2 N$$

Advanced DSA : Sorting

↳ Merge Sort
 ⇒ Quick Sort.