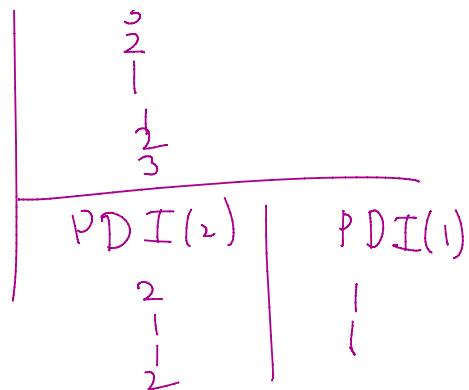


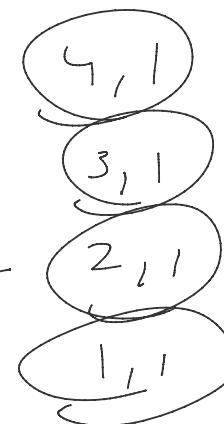
4
3
2
1
2
3
4



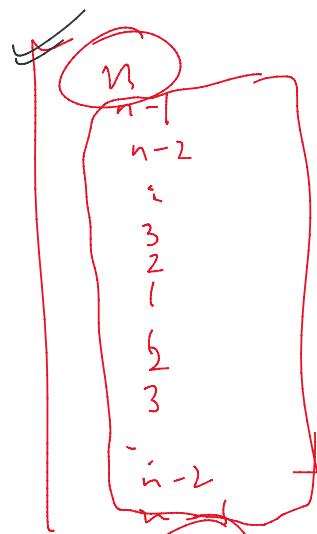
5
4
3
2
1
2
3
4

(s, c)

~~Sys₀(s)~~
~~sys₁(s)~~
~~sys₂(s)~~

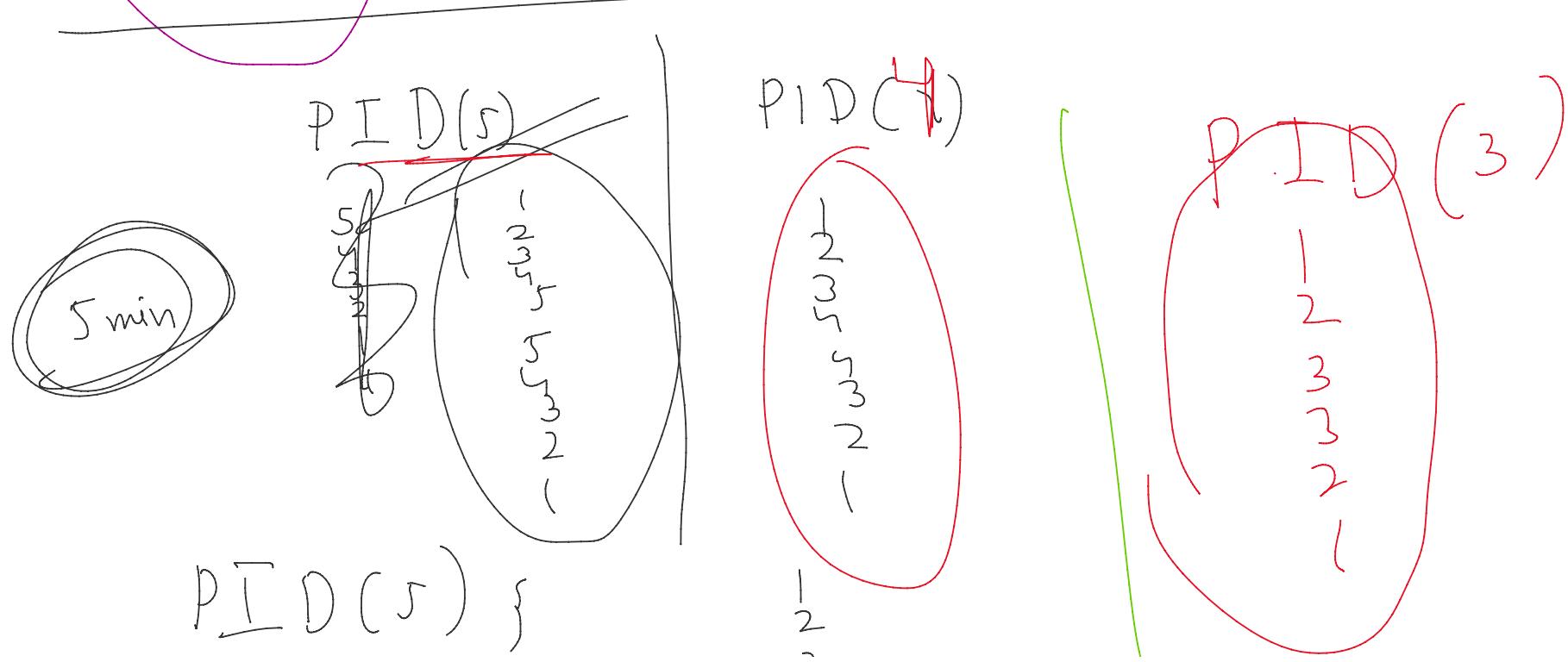
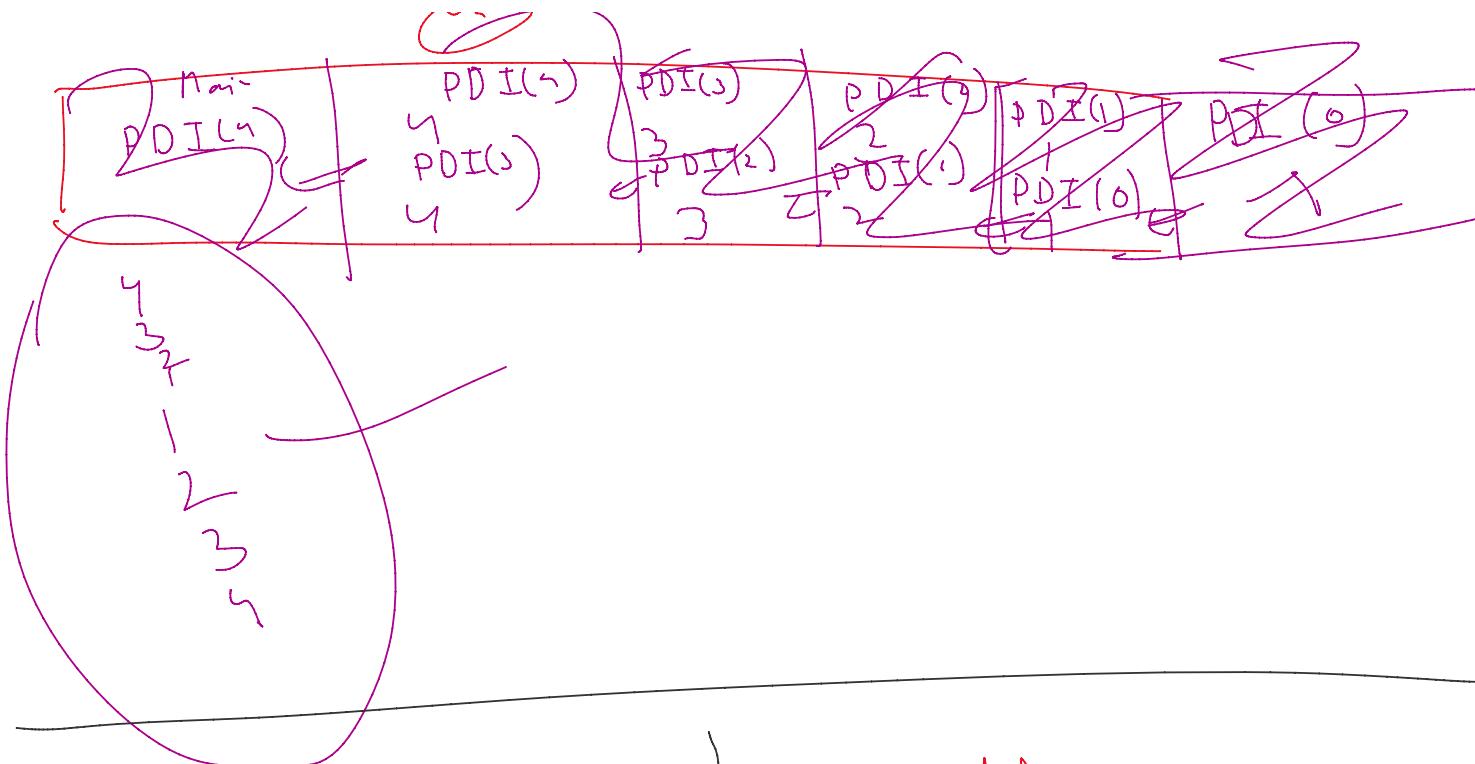


$PDI(s)$ {
 $Sys_0(s)$
 $PDI(s) \rightarrow$
 $Sys_1(s)$



$PDI(n-1)$

~~mai: | PDI(n) | PDI(i) | PDI(i) | ... |~~



PID(5)

PID(5) $\leftarrow (5, 5)$

1
2
3
4
5
6
7

1
2
3
4
5
6
7

)

PID(n)

PID(n-1) $\leftarrow n, n$

x

PID(4)

PID(3)

PID(3)

PID(2)

)

PID(2)

PID(1)

PID(1)

PID(1)

1
2
3
4
5
6
7

PID(s, e)

1
2
3
4
5

PID(2, 5)

2
3
4
5
6
7

PID(3, 5)

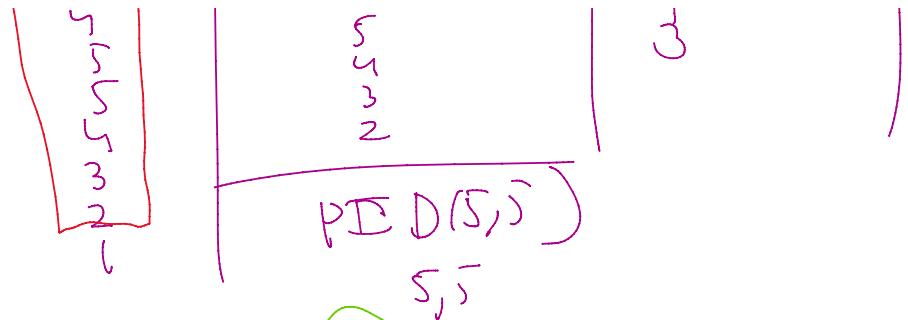
3
4
5
6
7

PID(4, 3)

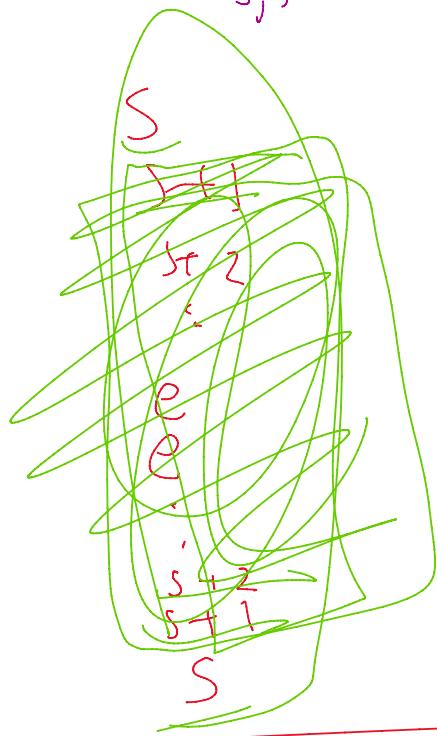
4
5
6
7

BP $\rightarrow s, e$
SP $\rightarrow st, e$

SP → S+I, e



15



PID($s + 1$)

PID) (4)

23

P I (3)

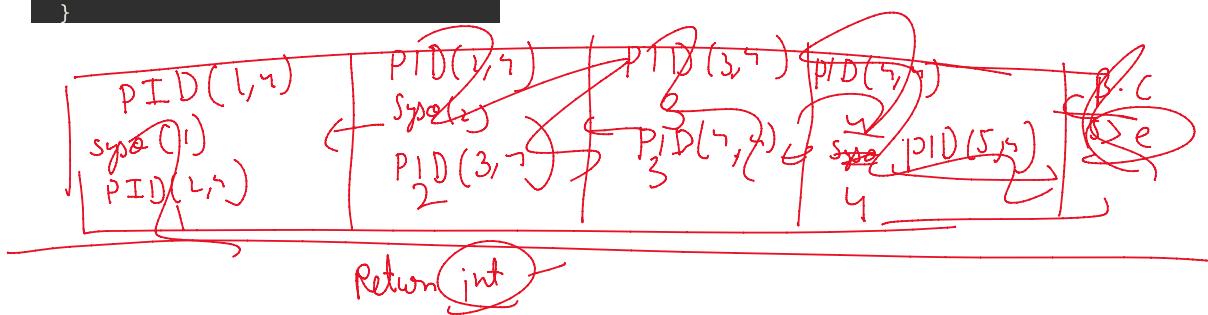
1
2

4
3
2

3
3
2
1

```
public static void PID(int s, int e) {  
    if(s>e) {  
        return;  
    }  
    BP : PID(s,e)  
    SP : PID(s+1,e)  
    System.out.println(s);  
    PID(s+1,e);  
    System.out.println(s);  
}
```

$\xrightarrow{PID(1,4)}$
1, 2, 3, 4, m, s, 2



Recursive !

$$Fac(5) = 120$$

$$Fac(4) = \frac{4 \times 3 \times 2 \times 1}{24}$$

$$Fac(3) = 6$$

$$Fac(2) = 2$$

$$Fac(1) = 1$$

$$\text{int } \text{Fac}(n) \leq 1 \times 2 \times 3 \times \dots \times n$$

$$\text{Fac}(n-1) = 1 \times 2 \times 3 \times \dots \times (n-1)$$

$$\text{Fac}(n-2) = 1 \times \dots \quad (\leftarrow \leftarrow)$$

$$\text{Fac}(n-5) = 1 \quad \dots \quad (\leftarrow \rightarrow)$$

$$a^b$$

b

(int) Pow ($\frac{a}{2}, \frac{b}{2}$)

$$\begin{matrix} 2, 1 \\ 2, 10 \end{matrix} = 10^{2^4}$$

b

$$\text{Fac}(5)$$

$$s_b = \text{Fac}(4)$$

↓

$$\text{Fac}(n)$$

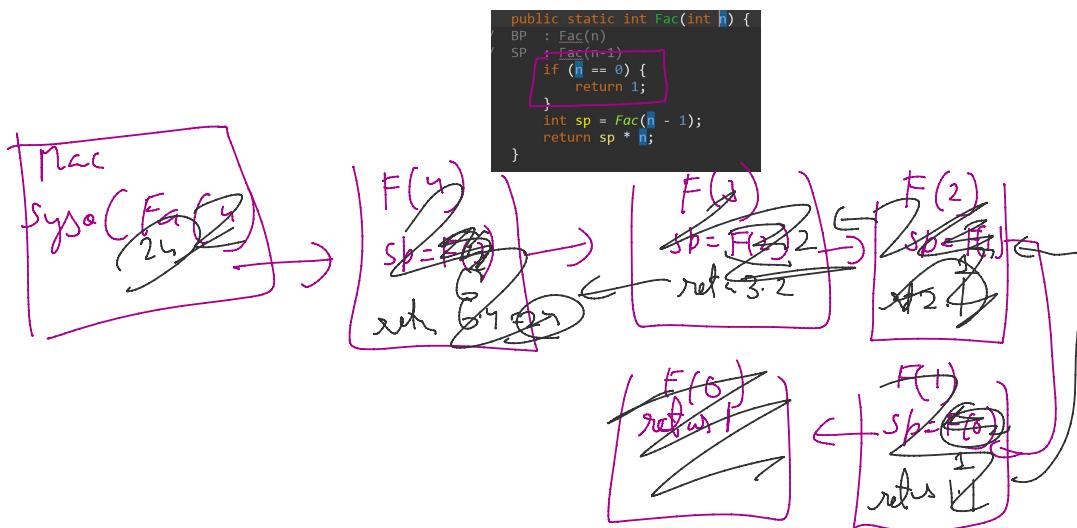
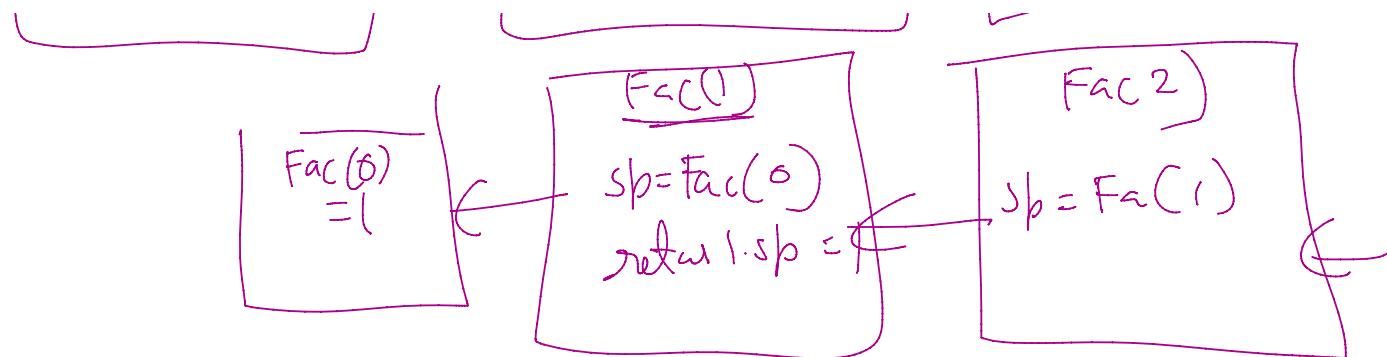
$$s_b = \text{Fac}(3)$$

$$\text{Fac}(3)$$

$$s_b = \text{Fac}(2)$$

$$\text{Fac}(1)$$

$$\text{Fac}(2)$$



$$\underbrace{a^b}_{a} \times \underbrace{a^c}_{a} \times \underbrace{a^d}_{a} = \underbrace{a^s}_{a}$$

$$a \times a \times a \times a \times a = 2^5$$

$$\overline{a^q \times a^r = a^{l_0}}$$

$$\text{Main} \rightarrow 2^3$$

```

public static int pow(int a, int b) {
    if(b==0) {
        return 1;
    }
    int sp = pow(a, b-1);
    return sp * a;
}

```

```

public static int pow(int a, int b) {
    if(b==0) {
        return 1;
    }
    int sp = pow(a, b-1);
    return sp * a;
}

```

```

public static int pow(int a, int b) {
    if(b==0) {
        return 1;
    }
    int sp = pow(a, b-1);
    return sp * a;
}

```

```

public static int pow(int a, int b) {
    if(b==0) {
        return 1;
    }
    int sp = pow(a, b-1);
    return sp * a;
}

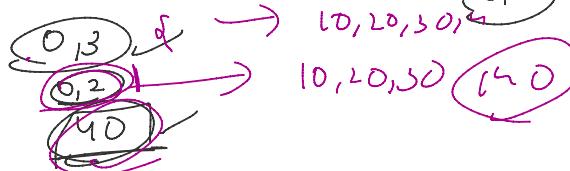
```

Array Rec

Q {10, 20, 30, 40}

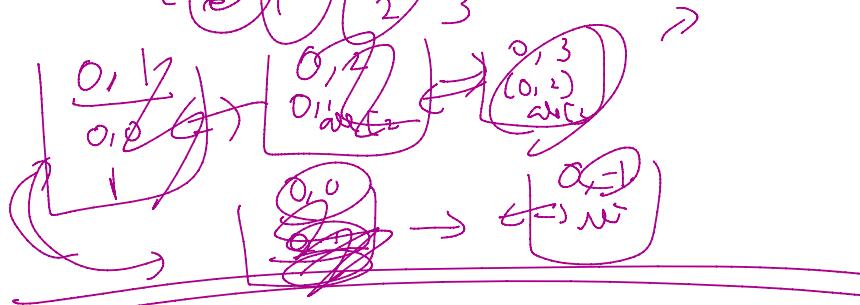
10, 20, 30, 40

4



5
{10, 20, 30, 40}

10, 20, 30, 40

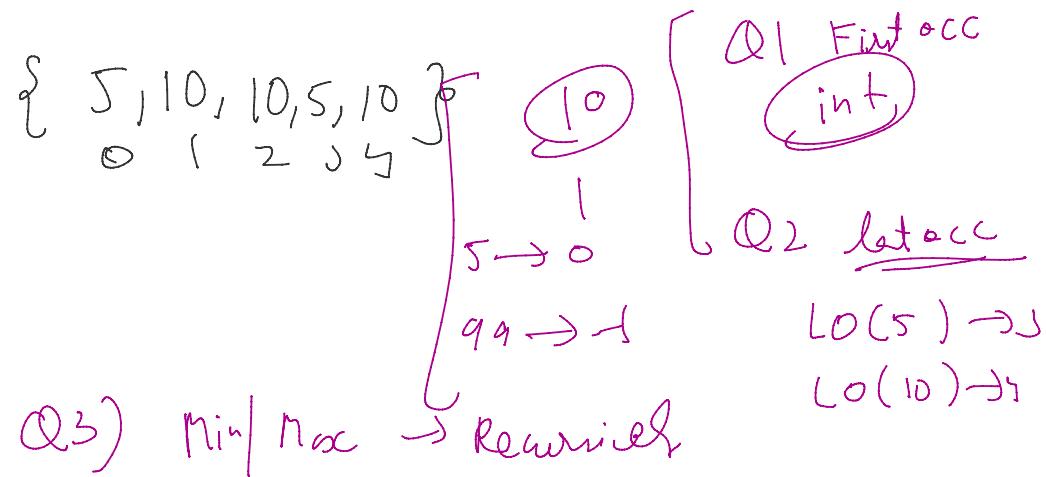
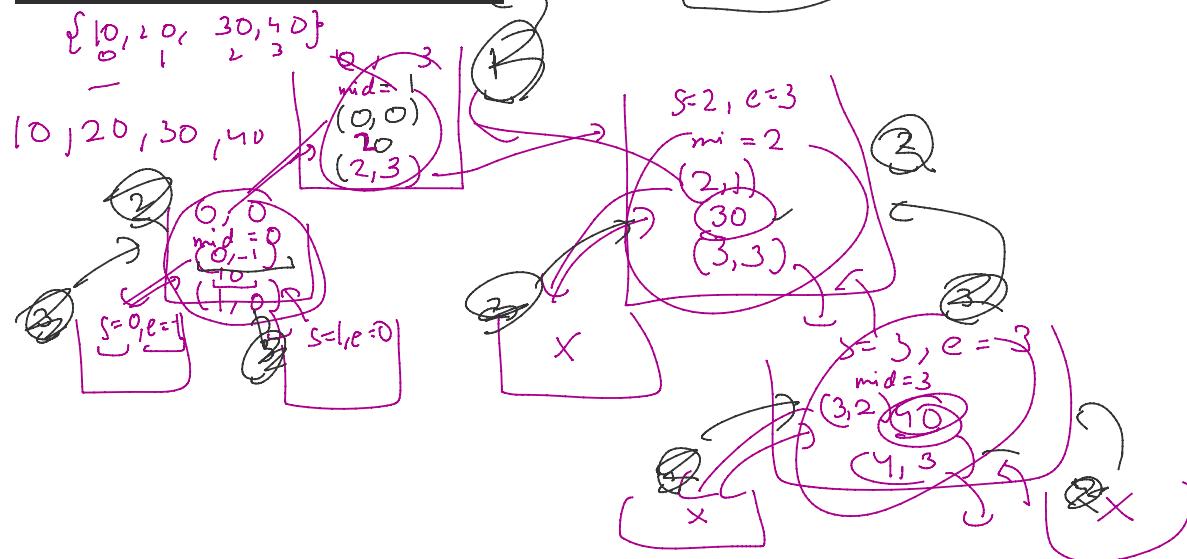


{10, 20, 30, 40}.

```

public static void print_arr3(int s, int e, int[] arr) {
    if(s>e) {
        return;
    }
    int mid = (s+e)/2;
    print_arr3(s,mid-1, arr);
    System.out.println(arr[mid]);
    print_arr3(mid+1, e, arr);
}

```



Q3) Min Max → Rewrite

Q4) Reverse Print