

Lec20

Monday, 5 July 2021

6:57 PM

G

principle

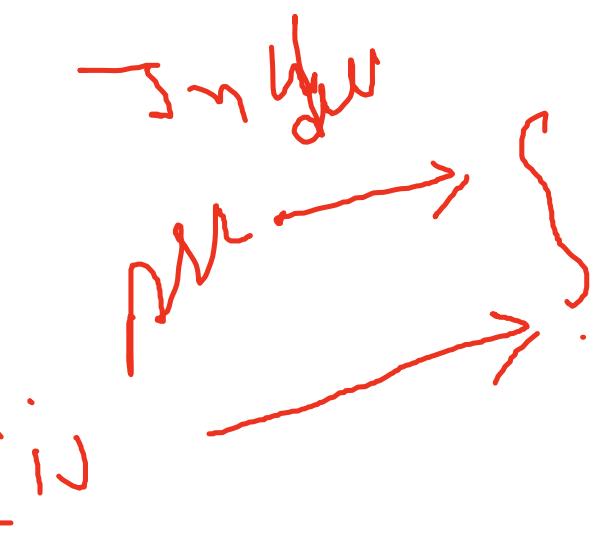
mathematic

P M II

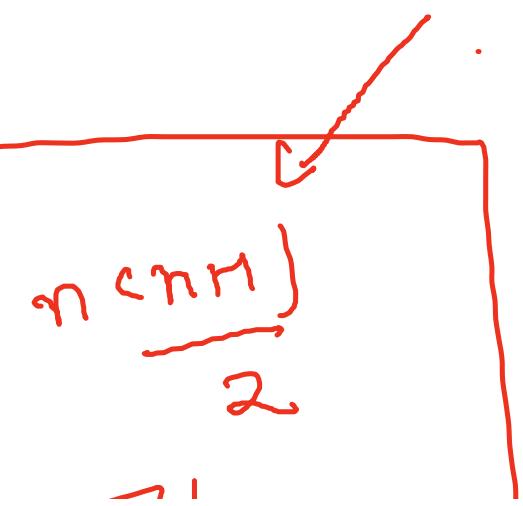
math

formula

$$\sum_{i=0}^n$$



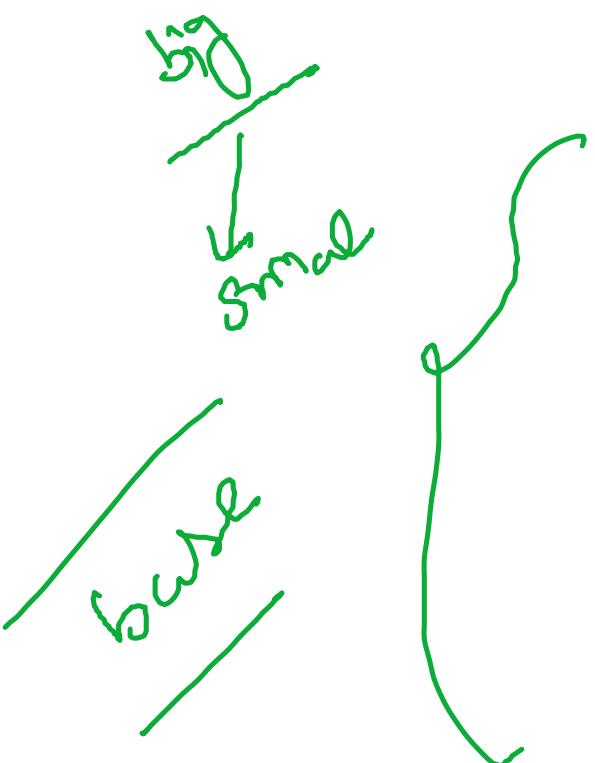
XW



$i = 1$

① For Small Input

base



$\sum_{i=1}^N i = ?$

$i = 1$

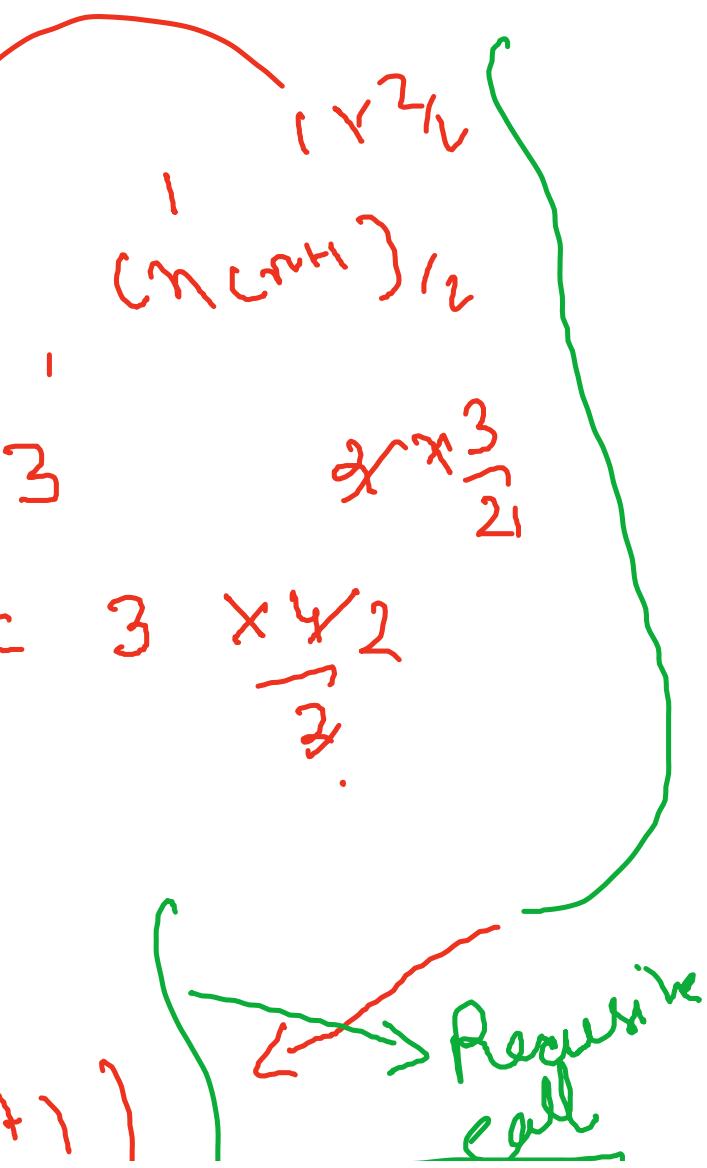
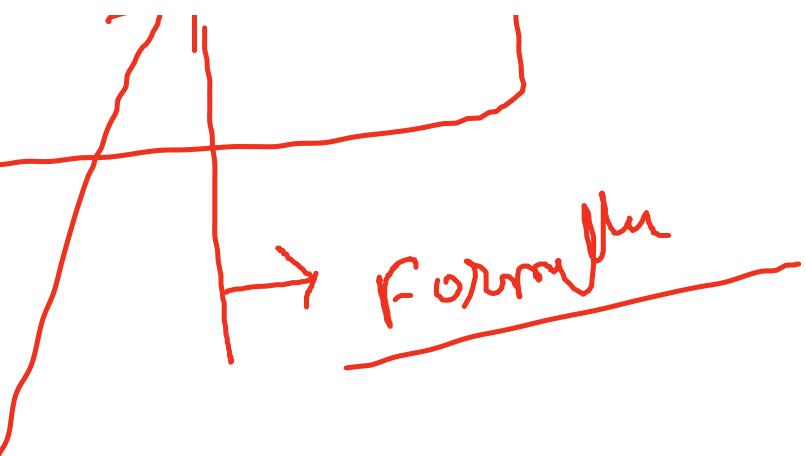
$1 + 2 =$

$1 + 2 + 3 =$

$i = k \leftarrow$

$N=12$

②



$$\begin{aligned}
 & \sum_{i=1}^n i = \\
 & 1 + 2 + \dots + k + K \\
 & \quad + \\
 & \quad \sum_{i=1}^{K+1} i = \\
 & \quad (K+1) \\
 & \quad + \\
 & \quad \sum_{i=1}^K i + (K+1) \\
 & \quad + \\
 & \quad \frac{K \times (K+1)}{2} + (K+1) \\
 & \quad + \\
 & \quad (K+1) \left(\frac{K+1}{2} \right)
 \end{aligned}$$

A hand-drawn diagram of a binary tree structure. The root node is a green horizontal line with a red '2' at its bottom. It has two children, which are red lines branching downwards. The left child has a red '2' at its bottom. The right child has a green bracketed label '(n, n+1)' above it. This pattern repeats for all nodes: each node is a red line with a red '2' at its bottom, and its right child is a green bracketed label '(n, n+1)' above it. The entire structure is enclosed in a large green bracketed label 'Selbstkomm.' at the bottom right.

R.M.S

$$= \left[(L+1) \times (L+2) \right]_2$$

$$|x+1| \times (k^2)$$

$(\langle n \rangle)$ $(\langle \langle n \rangle \rangle)$

L.H.S

$A^C \rangle \{$

$b^C \rangle :$

$\rightarrow A^C \rangle$
 $\rightarrow A^C \rangle$

3

3

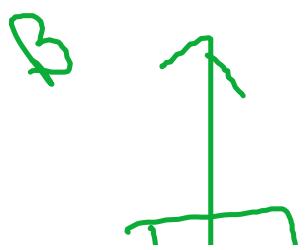
$A \uparrow$

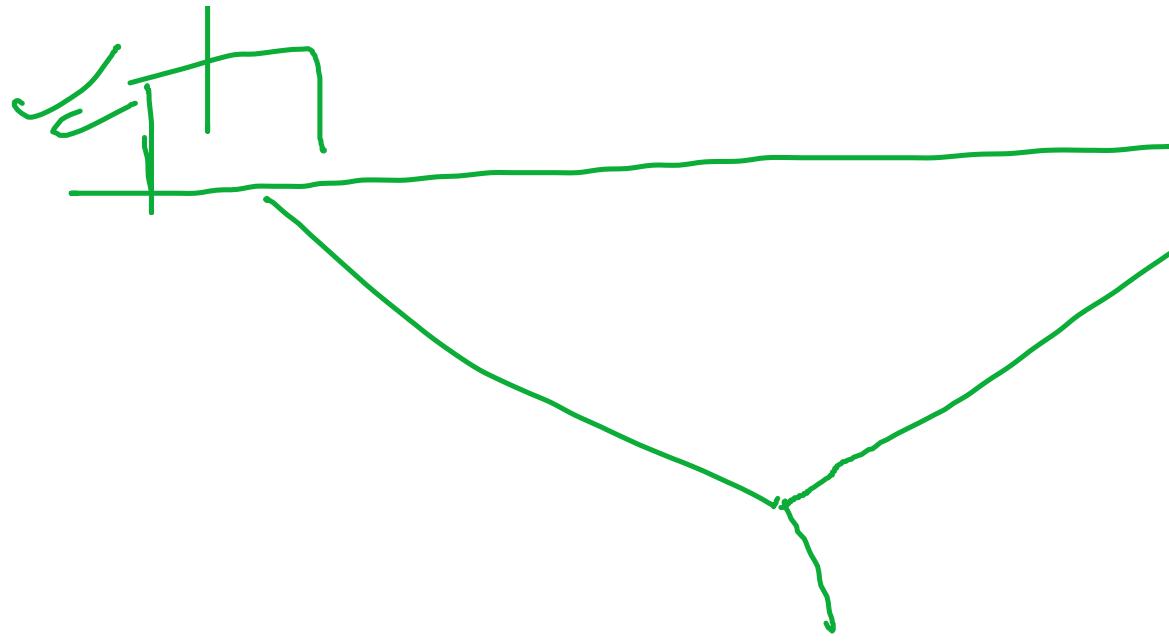
— () —
— — —
= RMS

W 0°

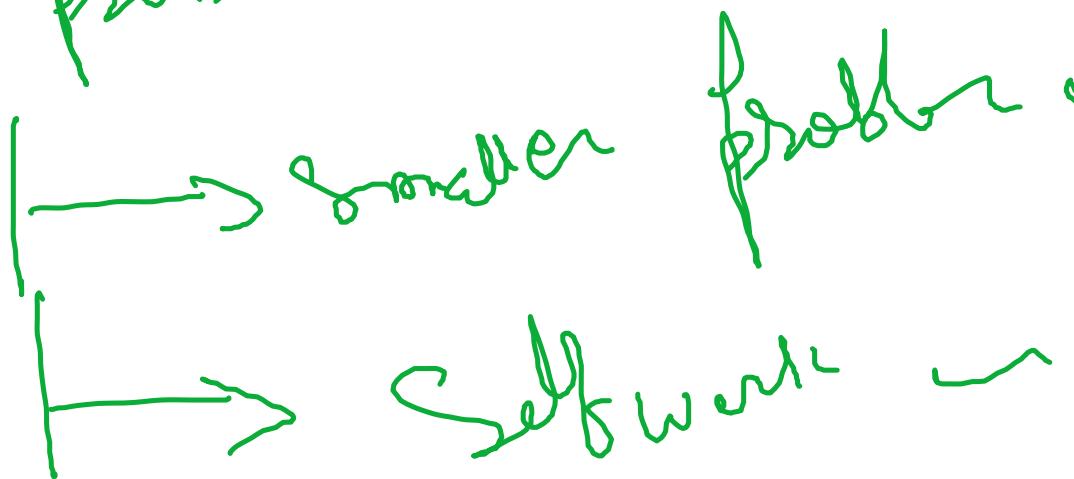
g

— . j

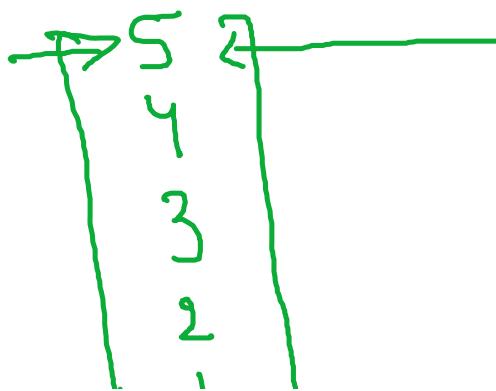




① Big problem



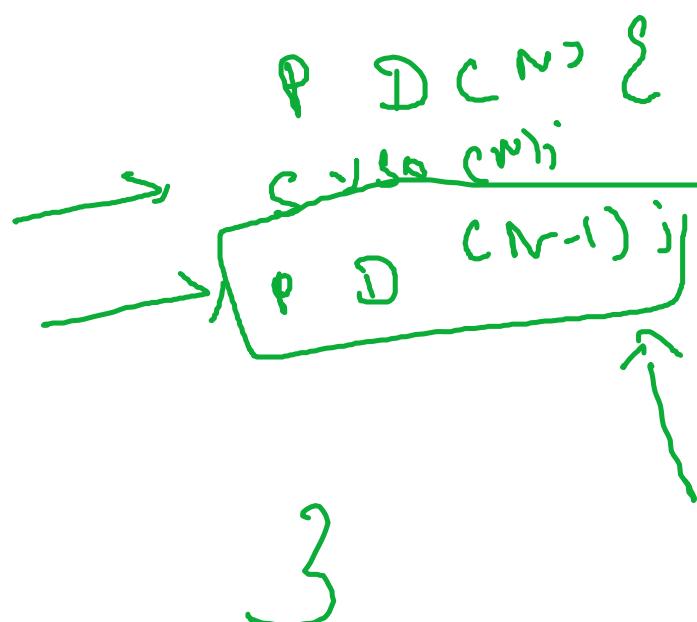
$N = 5$



PD

111

0 Box Cast

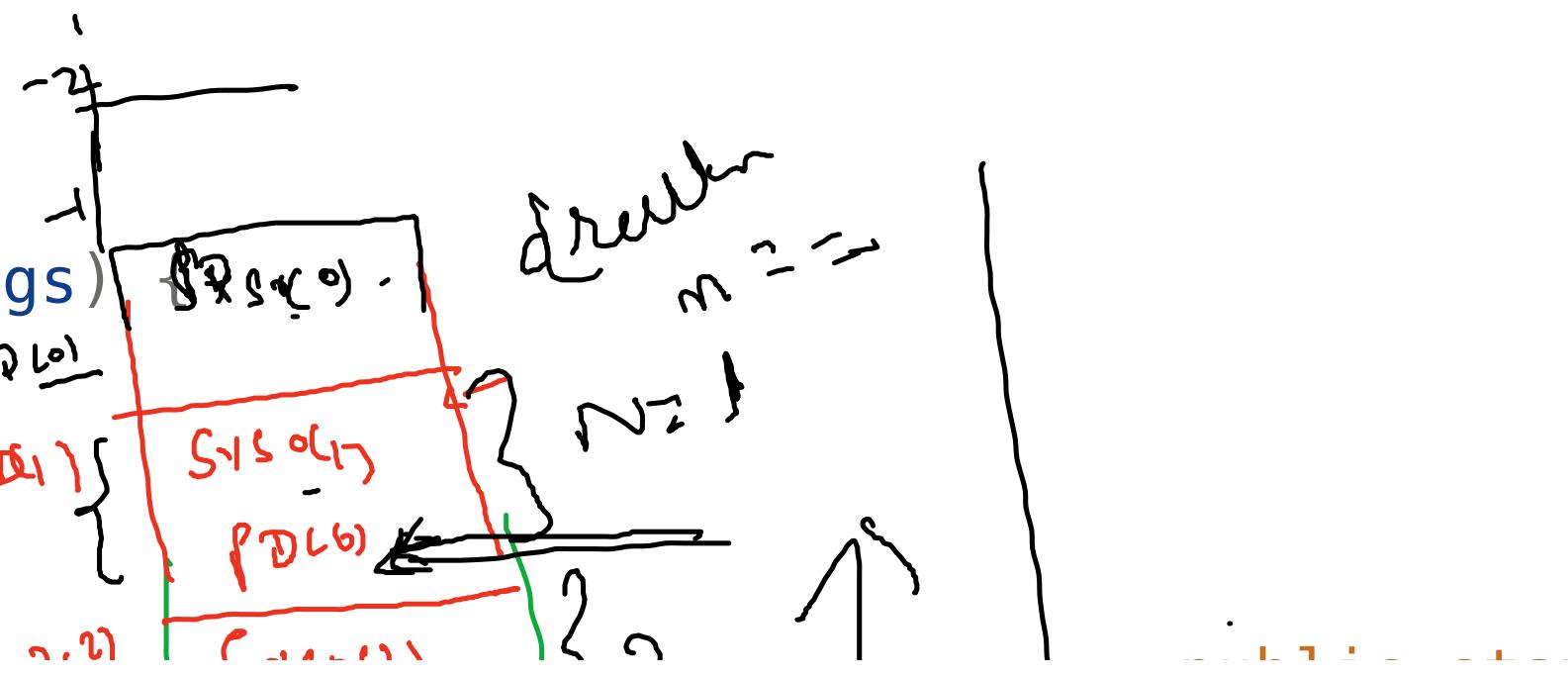


```

public class Print_Dec {
    public static void main(String[] args) {
        // TODO Auto-generated method stub
        → PD(5);
    }
    public static void PD(int n) {
        .
        .
        .
    }
}

```

5₁




```
System.out.println(n);
```

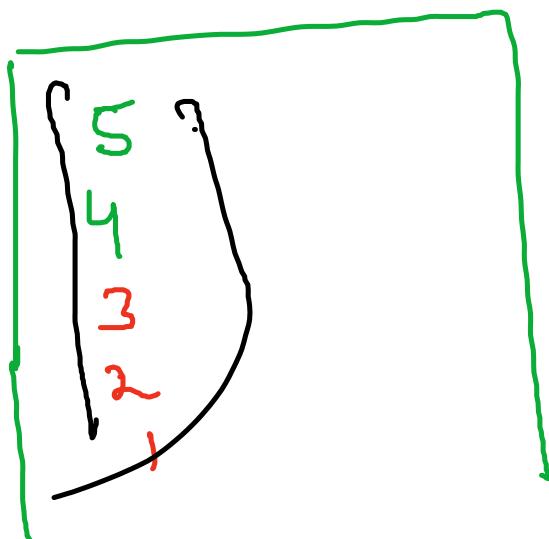
```
PD(n-1);
```

```
}
```

```
}
```

fault

PD



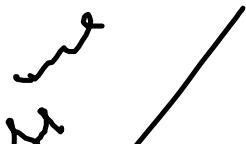
P_D(5)
P_D(5)

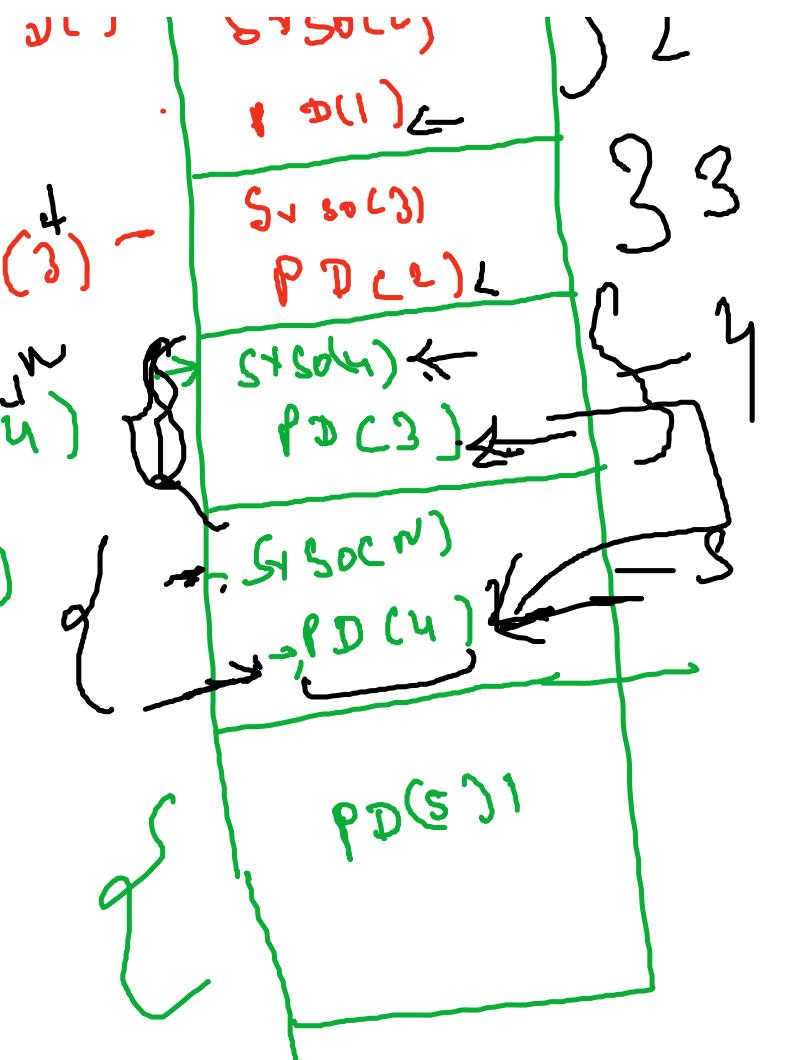
main

$\Rightarrow n = 5$

1
2
3
4
5

1
2
3
4
5





```

public static int PI(int n) {
    // Base case
    if (n == 0)
        return 1;
    // small problem
    int sum = PI(n - 1);
    // self work
    sum += n;
    System.out.println("PI(" + n + ") = " + sum);
    return sum;
}

```

```
tic void pi(int n) {\nse\n} {
```

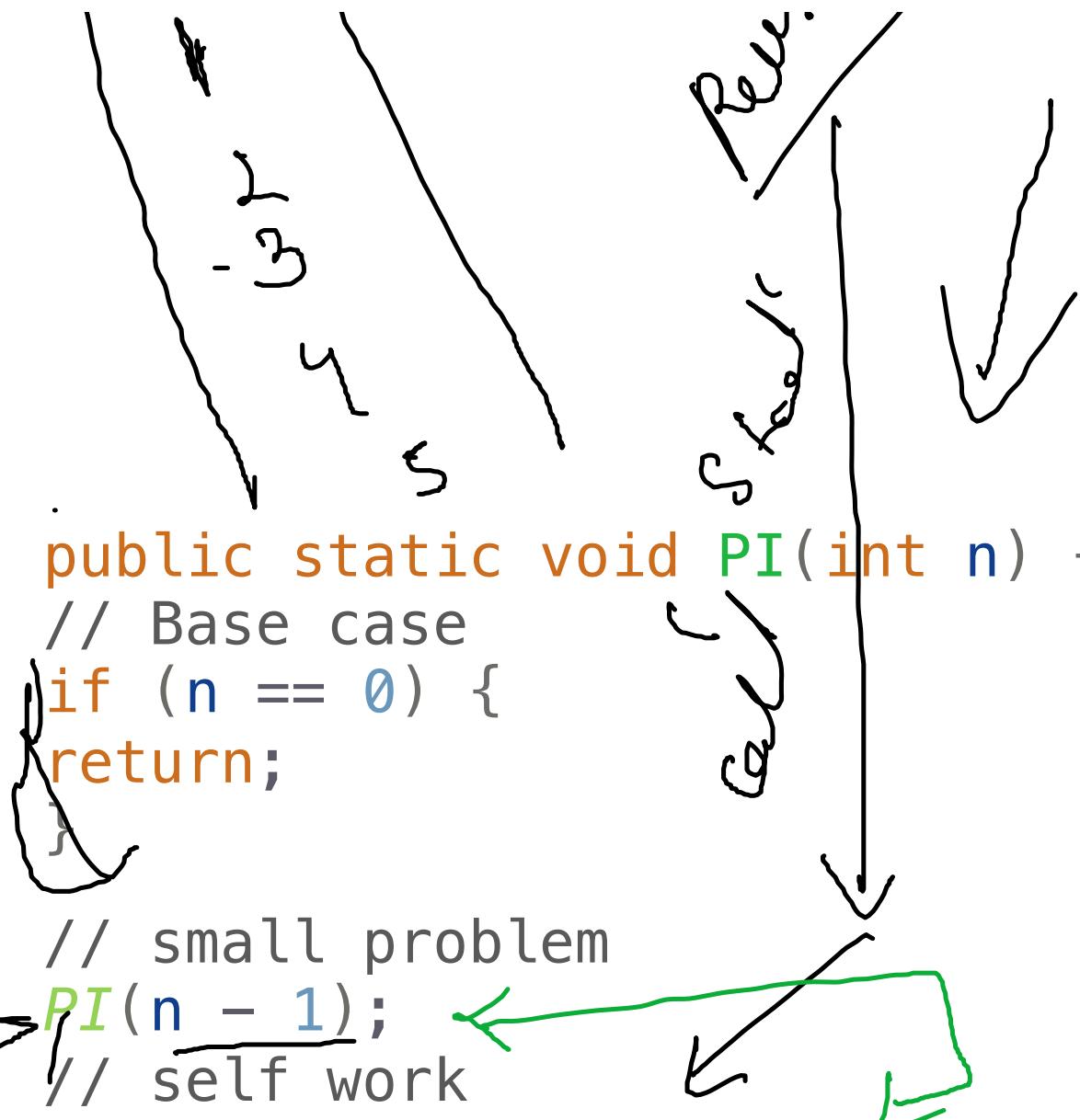
roblem

rk

```
• println(n);
```

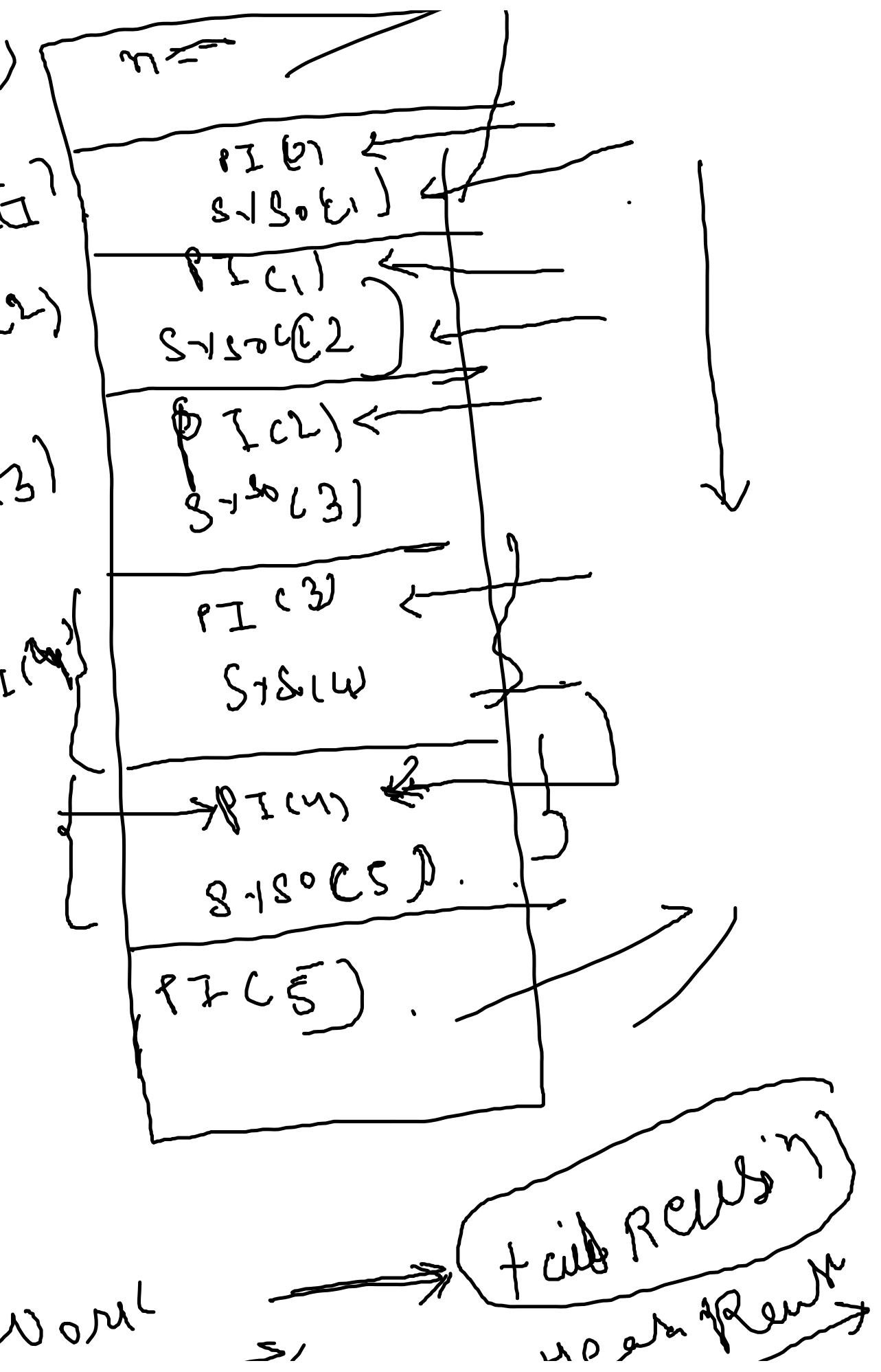


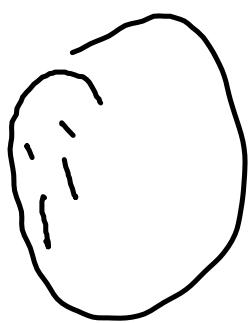
```
public static void PI(int n) {  
    // Base case  
    if (n == 0) {  
        return;  
    }  
    // small problem  
    → PI(n - 1);  
    // self work  
    System.out.println(n);  
}
```



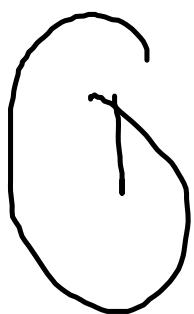
(i)

Stack ↑





Stack



fact (S)

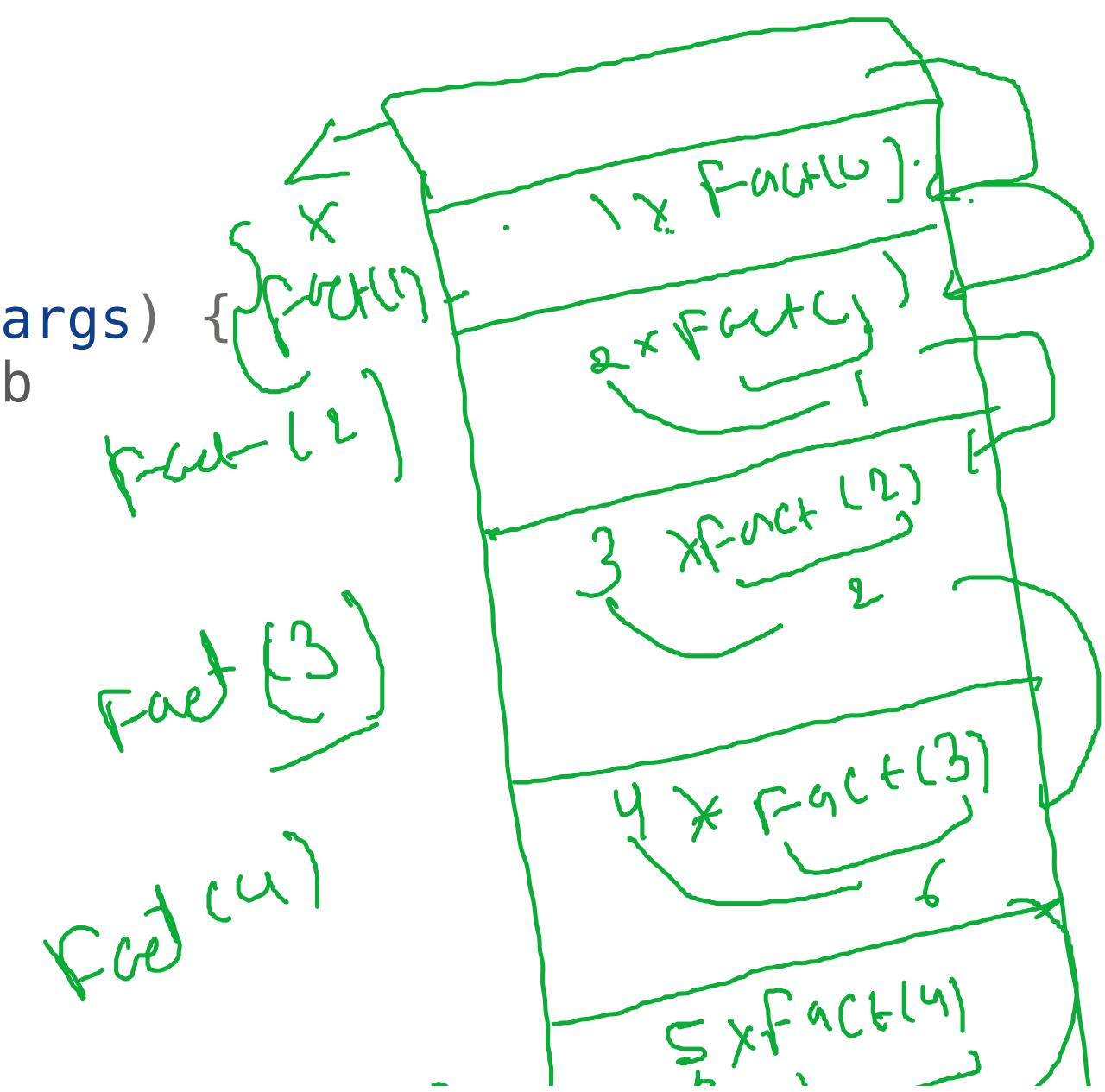
$$S' = S \times \sqrt{r}$$

$$n_{\text{ret}}(S) = S^{\downarrow} \text{fact}$$



355 → 6051

```
public class Fact {  
    public static void main(String[] args) {  
        // TODO Auto-generated method stub  
        System.out.println(fact(5));  
    }  
    public static int fact(int n) {  
        if(n==0) {  
            return 1;  
        }  
        // head  
        return n*fact(n-1);  
    }  
}
```

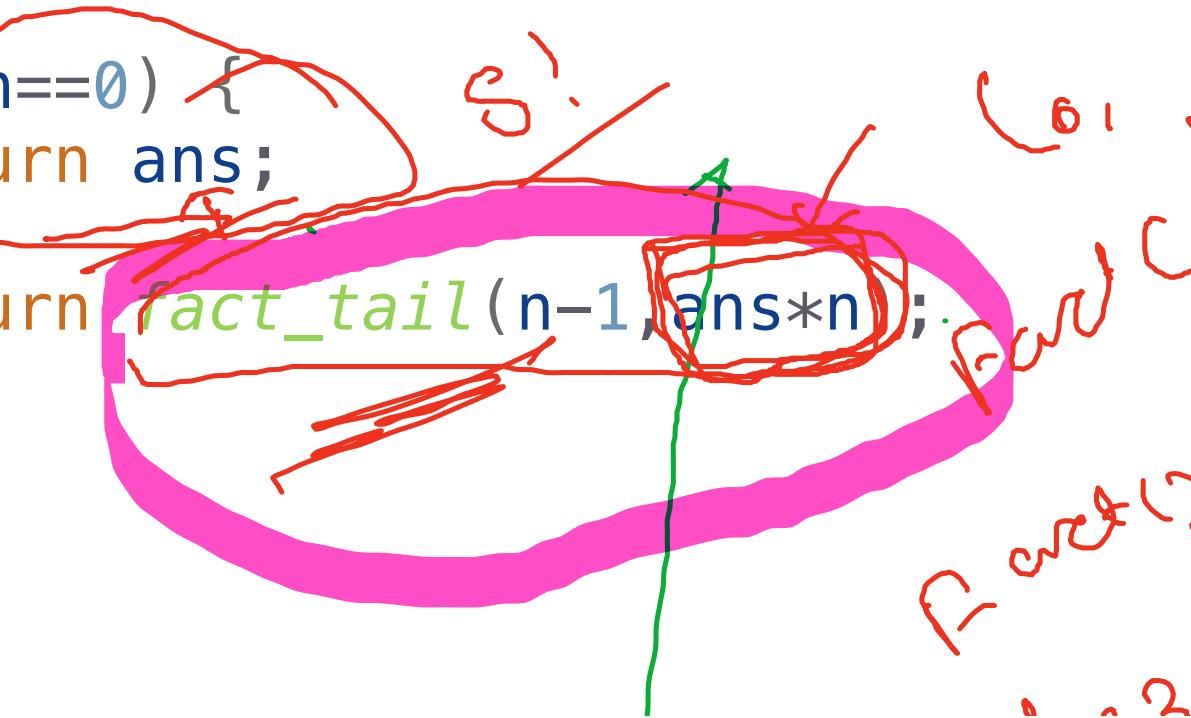


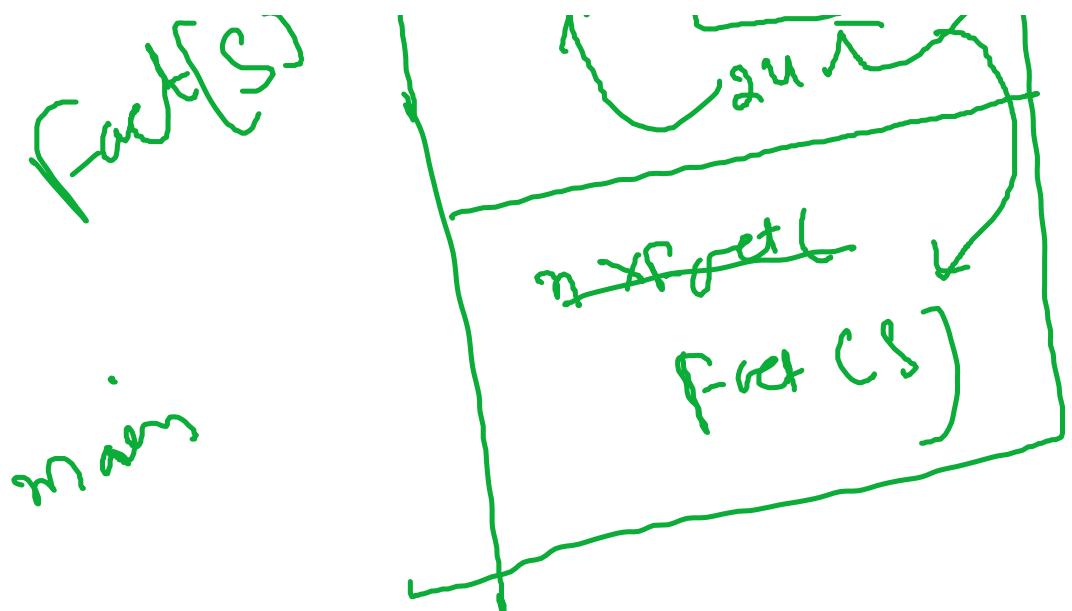
V }

CD tail n--

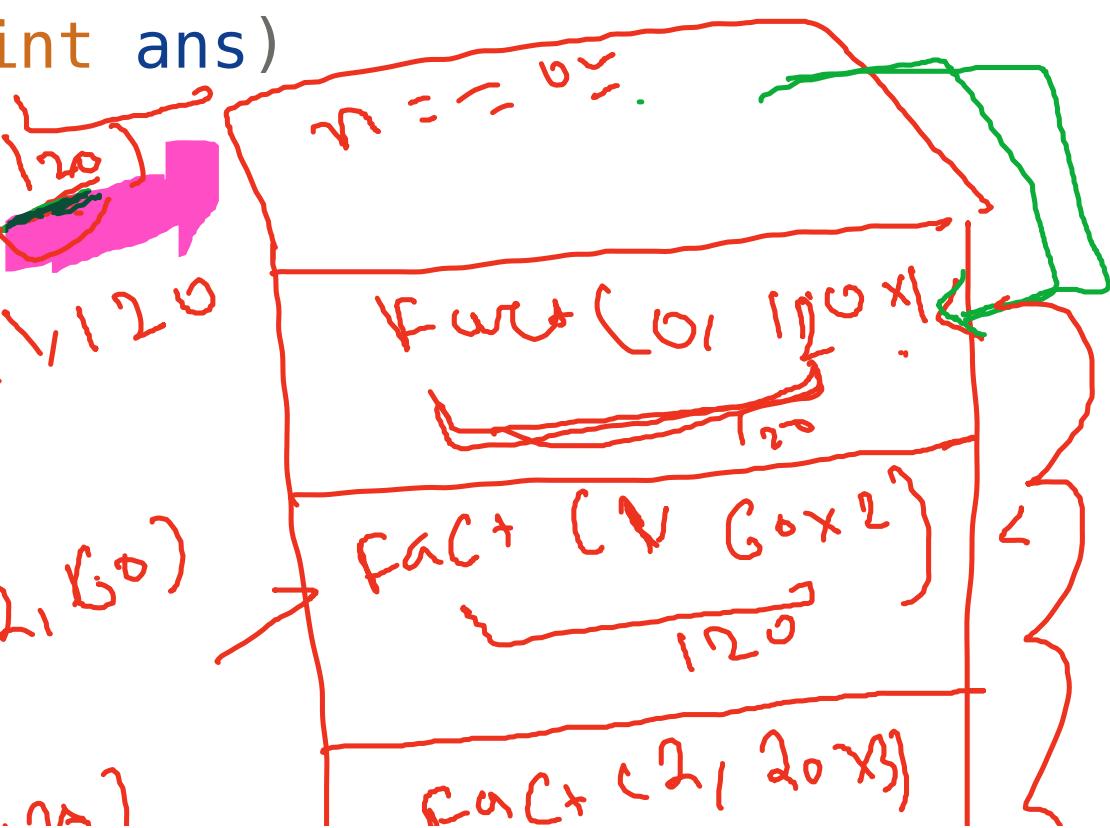
p1

```
public static int fact_tail(int n, int ans) {
    if(n==0) {
        return ans;
    }
    return fact_tail(n-1, ans*n);
```





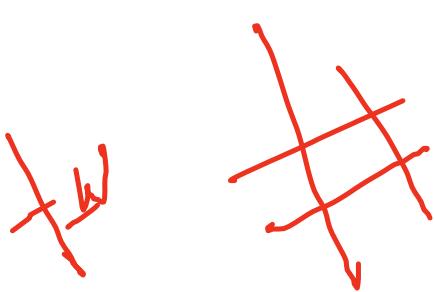
fact



2



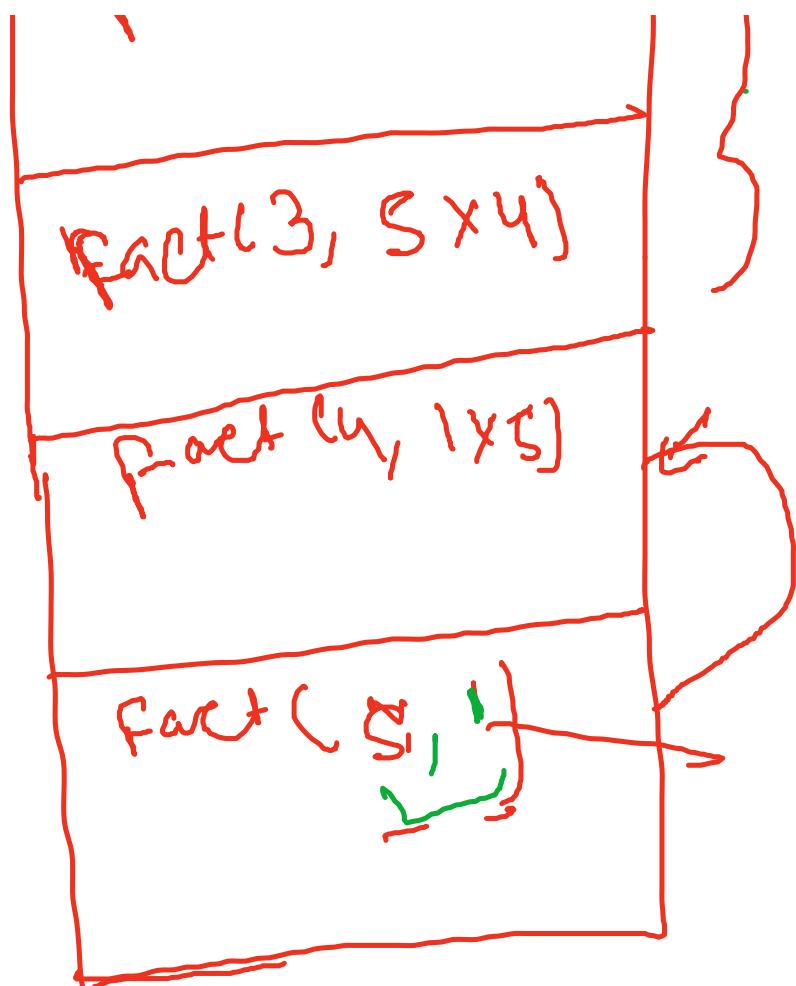
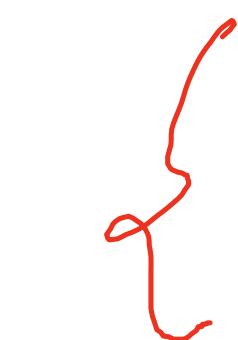
Fact C
Fact C
mean far



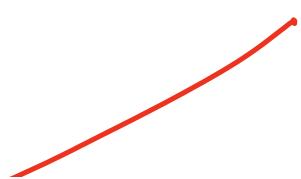
Tally Revision

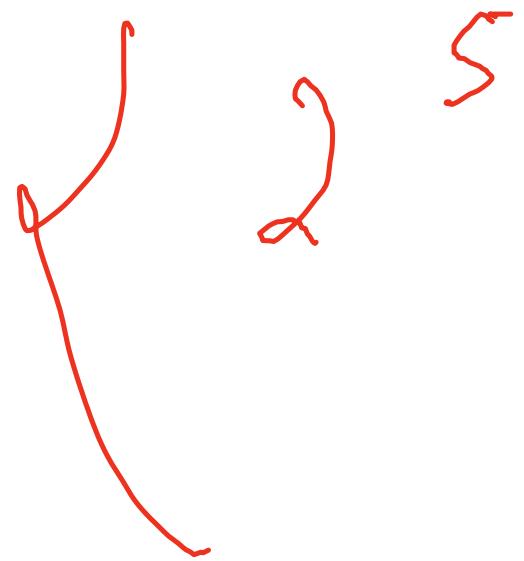
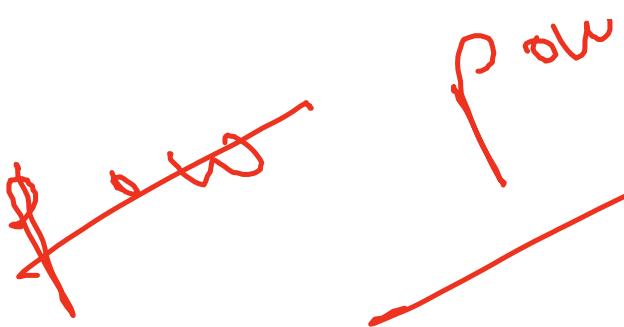
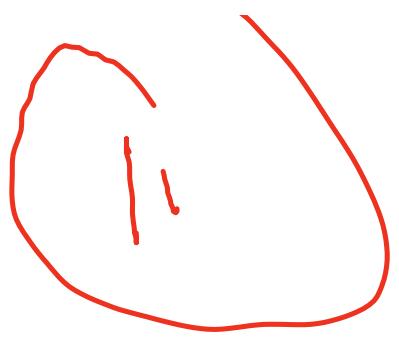
100)
is

es(1)



list
positive cell





25 *Ansatz*

$\text{pow}(\alpha, \beta) =$

