Lec-20
08/05/25 2:04 PM

# Recursion

Fun calling

```
A() {
B()
}
```

Fun calling itself

```
A() {
A()
}
```

PMI

math → proof → X

$\sum_{i=1}^{i=N} i = 1+2+3+\cdots N = \frac{N\times(N+1)}{2}$

① $i = \sum_{i=1}^{i=1} i = 1\times(1+1)/2$

$i = \sum_{i=1}^{i=2} i = 1+2 = \frac{2\times(2+1)}{2}$

(ii) $\sum_{i=1}^{i=k} i = 1+2+3+\cdots k = \frac{k\times(k+1)}{2}$

(iii) $\sum_{i=1}^{i=k+1} i = 1+2+\cdots k+(k+1) = \frac{k\times(k+1)}{2} + (k+1) = \frac{(k+1)\times(k+2)}{2}$

① Smaller formula correct

(ii) N=k correct ✓

(iii) N=k+1 proof

$\frac{(k+1)\times(k+2)}{2}$

$N! = (N-1)!\times N$

BP

S.P

Recursion ताकत LIFO LIFO है

$5! = 4!\times 5 = 120$

$4! = 3!\times 4$

$3! = 2!\times 3$

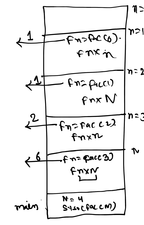
$2! = 1!\times 2$

$1! = 0!\times 1$

```
public static void main(String[] args) {
    // TODO Auto-generated method stub
    int n = 5;
    System.out.println(fac(n));
}
public static int fac(int n).{
    {
    int fn=fac(n-1);// sp
    return fn*n;
}
```

```
public static int fac(int n) {
    if (n == 0) {
        return 1;
    }

    int fn = fac(n - 1);// sp
    return fn * n;
}
```

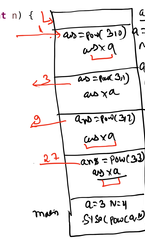



answer

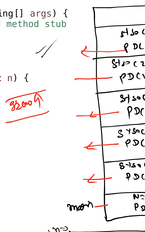
Stack

downward

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

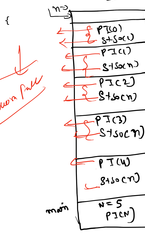



```
public static void main(String[] args) {
    // TODO Auto-generated method stub
    int n = 5;
    PD(n);
}
public static void PD(int n) {
    if (n == 0) {
        return;
    }
    System.out.println(n);
    PD(n - 1);
}
```




5
4
3
2
1

```
public static void PI(int n) {
    if (n == 0) {
        return;
    }
    PI(n - 1);
    System.out.println(n);
}
```




down fac

1
2
3
4
5

i=2

item=4

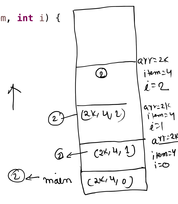
| 2 | 3 | 4 | 5 | 6 | 4 | 3 |
|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | |

1 5 0 X 1

```
public static int Index(int[] arr, int item, int i) {
    if (i == arr.length) {
        return -1;
    }
    if (arr[i] == item) {
        return i;
    }
    return Index(arr, item, i + 1);
}
```

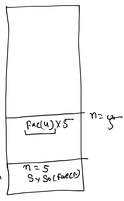
arr

| 2 | 3 | 4 | 5 | 4 |
|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 |



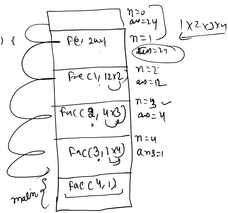

```
public static int fac(int n) {
    if (n == 0) {
        return 1;
    }

    //int fn = fac(n - 1);// sp
    return  fac(n - 1) * n;
}
}
```




```
public static int fac(int n, int ans) {
    if (n == 0) {
        return ans;
    }

    // int fn = fac(n - 1);// sp
    return fac(n - 1, ans * n);
}
```




1×2×3×4