Lecture 8

Saturday, 25 January 2020 2:42 PM

Recursion .

Mathematical Induction :-

$$Sum = n(n+1)/2.$$

For $n = 1$, $Sum = 1$

For $n = K$, $Sum = (K)(K+1)/2$ ✓

For $n = K+1$, $Sum = (K+1)(K+2)/2$.

$$\underbrace{1+2+3+4+\dots+K}+K+1 = Sum$$

$$\frac{(K)(K+1)}{2} + (K+1) = Sum$$

$$\frac{(K+1)(K+2)}{2} = Sum$$

Q=>

```
p s void PD(int n){        // 5
    if(n==0){              // PD(0)
        return;
    }
    Syso(n);               // xx  -- 1
    PD(n-1);               // *
}
```

PD(5), PD(4) . . .

PD(3), PD(2)

PD(1)

5

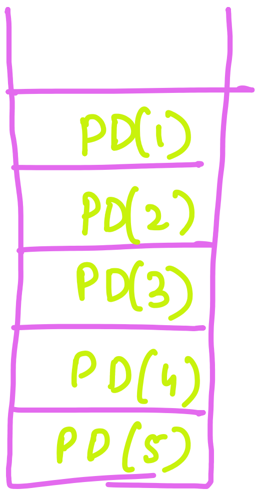
4

3

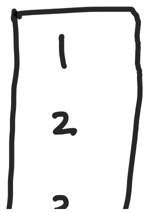
2

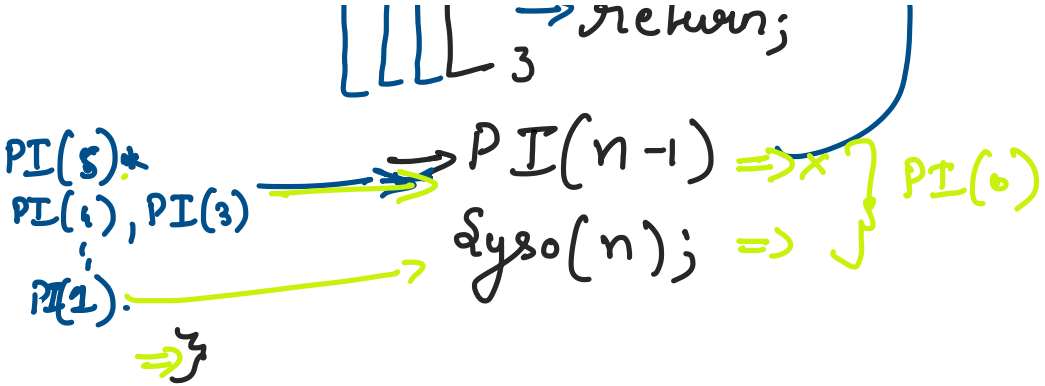
1

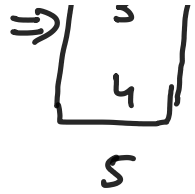
PD ->



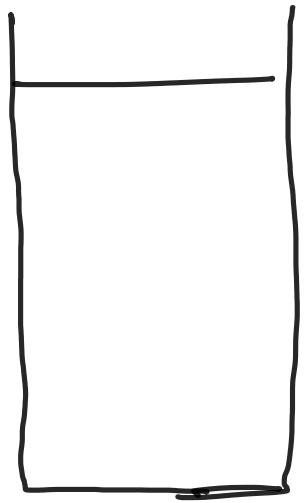

```
p s void PI(int n){        // 5
    if(n==0){
```







1
2
3
4
5



2 chances. < stack is build
stack is finishing.

p s

```
void PDI(int n){
if(n==0){
return;
3
fyso(n);
```

3 ✓
4 ✓
5 ✓

PDI(n-1);
↳ Syso(n); ✓

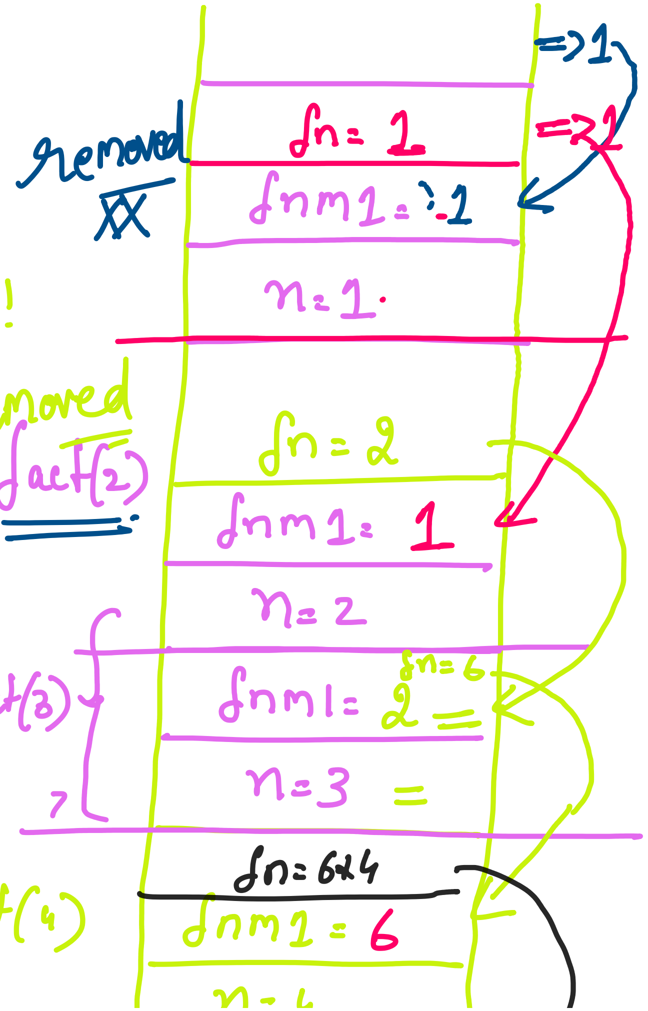
}

```
p s int fact (int n){   4
  if (n==0){
     return 1;
  }
* int fnm1 = fact(n-1);     3!
  ↳ int fn = n * fnm1;      4*3!
  return fn;
}
```

4! => 4 * 3! → Main.
↳ 3 * 2!
↳ 2 * 1!
↳ 1 * 0!

=> (24)

fact(4)



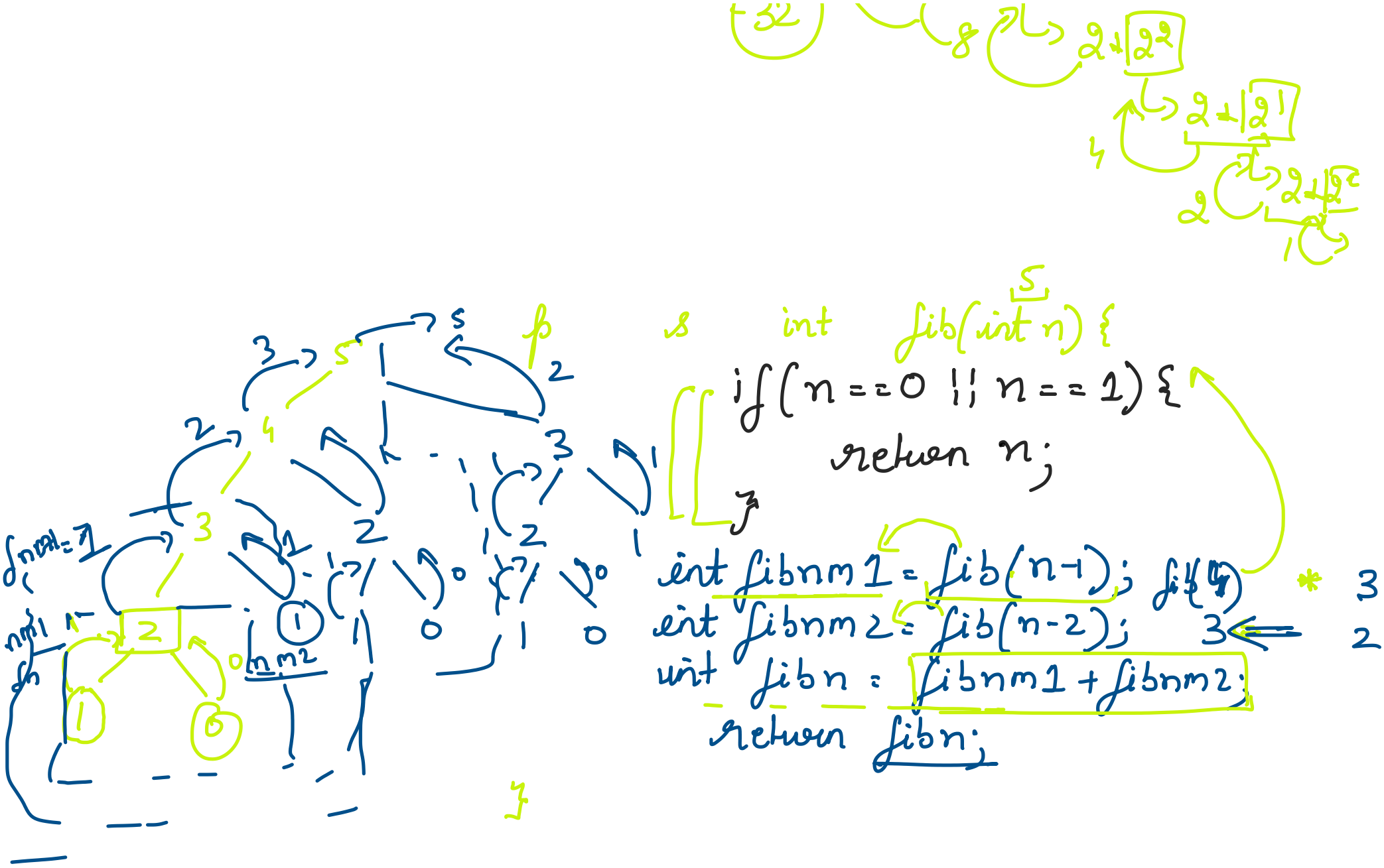

```
int power(int x, int n){
    if (n==0){
        return 1;
    }
    int pnm1 = power(x, n-1);
    int pn = x * pnm1
    return pn;
}
```

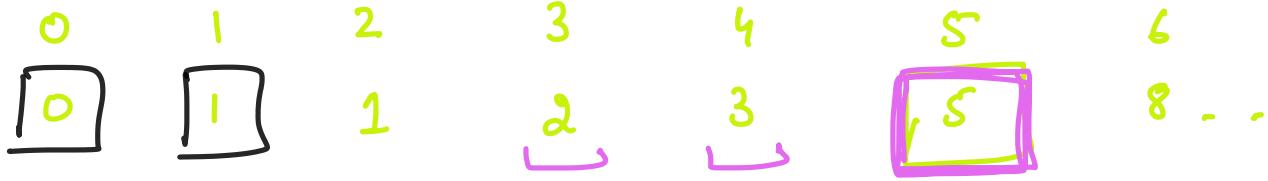
x^n

$2^5 \Rightarrow x^n \Rightarrow x * x^{n-1}$

$2^5 = 2 * 2^4$

$2^4 \hookrightarrow 2 * 2^3$


int fib(int n) {
if(n==0 || n==1) {
return n;
}
int fibnm1 = fib(n-1);
int fibnm2 = fib(n-2);
int fibn = fibnm1 + fibnm2;
return fibn;
}



n = 35, 45

fib(5) = fib(4) + fib(3)