Recursion

→ when a function calling itself

```
f(){
    f();
}
```

```
suman(){
    suman();
}
```

PMI

Sum of first n natural number = $\frac{n(n+1)}{2}$

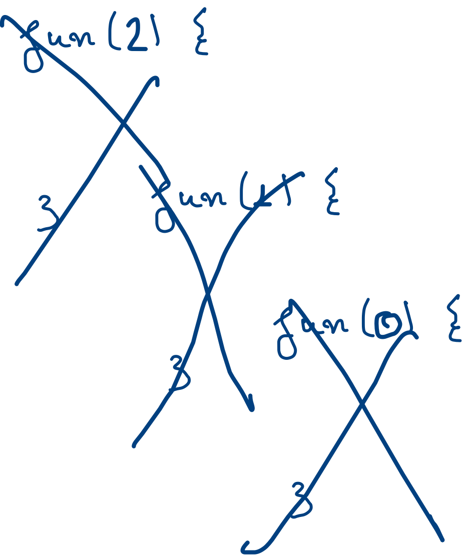
1. Prove for $n=1$ → sum = 1
2. Prove for x

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

3. for $(x+1)$

$$[1+2+3+4+ \dots + (x-2)+(x-1)+x] + (x+1)$$

$$= \frac{x(x+1)}{2} + (x+1) = \frac{(x+1)(x+1+1)}{2}$$



Print Decreasing

Input → 5

Output → 5 4 3 2 1

Expectation	Faith
n	n-1
n-1	n-2
n-2	.
.	.
3	3
2	2
1	1

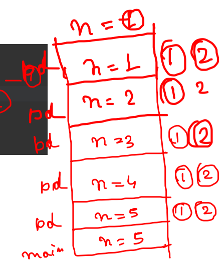
while (n >= 1)

```
void pd(4){
    printf("%d ...);
    pd(n-1);
}
```

pd(3){ pd(2){

```
void pd(int n){
    printf("%d\n",n);
    pd(n-1);
}
```

5
4
3
2
1

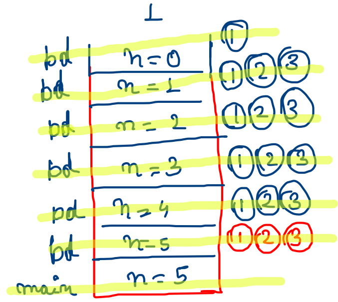


pd(1)

pd(0)

```
void pd(int n){
    if(n<1) return;
    printf("%d\n",n);
    pd(n-1);
}
```

5
4
3
2
1



Print Increasing

Input → 5
Output → 1 2 3 4 5

Exp: 1, 2, 3, …, n

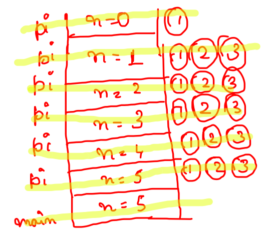
Faith: 1, 2, 3, …, (n-2), (n-1)

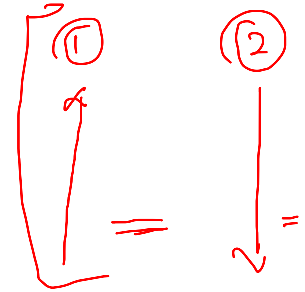
```c
void pi(int n){
    if(n<1) return;   // (1)
    pi(n-1);          // (2)
    printf("%d",n);   // (3)

}
```

pi(1) {

}

1
2
3
4
5





Print Decreasing Increasing

Input → 5

Output → 5 4 3 2 1 1 2 3 4 5

Factorial

Input → 5
Output → 120

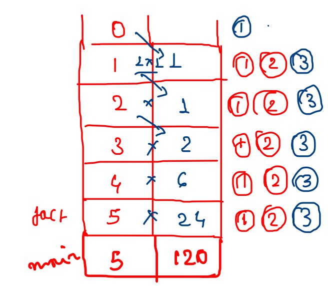
Exp: n × n-1 × … 1

Faith: [n-1 × … 1] = k

Result → k * n

```c
int fact(int n){
    if(n<1) return 1;    // (1)
    int k = fact(n-1);   // (2)
    return n*k;          // (3)

}
```

fact(1) {
int k = fact(0);
}



fact n → k main ( ) ⇒ 120