

(Q) Sum of 1st n numbers

ways  $\begin{cases} \rightarrow \text{parameterised} \\ \rightarrow \text{functional} \end{cases}$

} These are patterns.  
understand & learn them.

① parameterised

~~f(i, sum) {~~  
~~if (i < 1) { x~~  
~~cout << sum;~~  
~~return;~~  
~~}~~

f(i-1, sum+i)  
0+3

int main() {  
int n;  
cin >> n;  
f(n, 0)  
}

f(2, 3) {  
if ( ) x  
f(i-1, 3+2)  
}

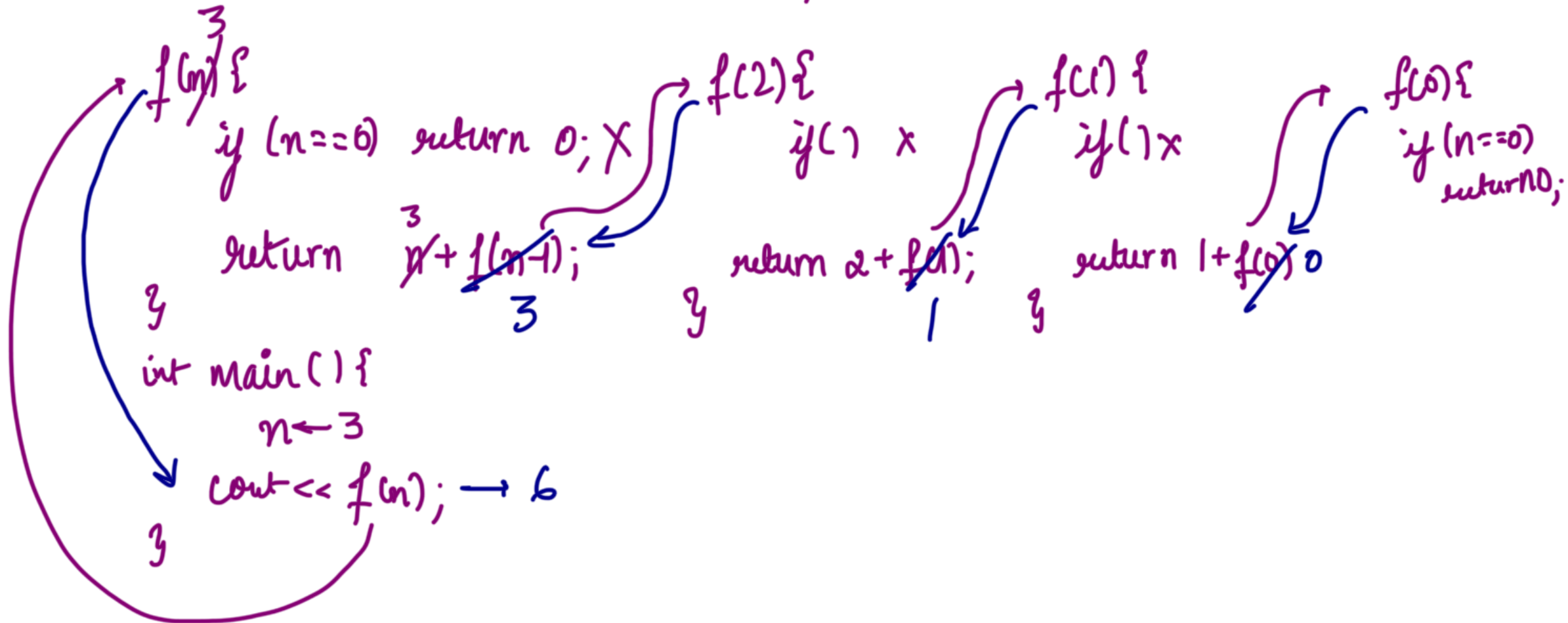
f(1, 5) {  
if ( ) x  
f(i-1, 5+1)  
}

f(0, 6) {  
if (i < 1) {  
cout << sum;  
return;  
}  
f( ) x  
}

i sum  
f(3, 0)  
✓  
f(2, 3)  
f(1, 6)  
f(0, 6)

② functional

$$\begin{array}{c}
 n=3 \\
 \downarrow \\
 3 + \frac{f(2)}{\uparrow} \\
 2 + \frac{f(1)}{\downarrow} \\
 1 + \frac{f(0)}{\nearrow 0}
 \end{array}$$



(0) factorial of n :

Parameterised

Functional

```
f (int n, int fact) {
```

```
    if (n < 1) { cout << fact;
```

```
        return;
```

```
    f (n-1, fact*n);
```

```
}
```

```
int main() {
```

```
    int n;
```

```
    cin >> n;
```

```
    f (n, 1);
```

```
}
```