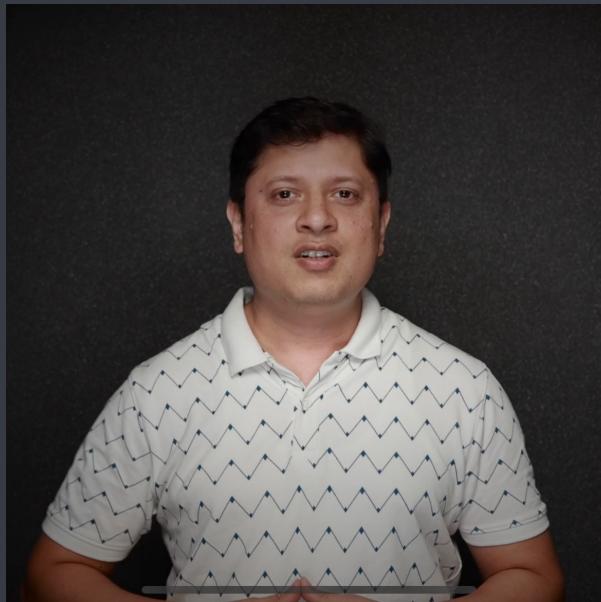


C Language

Recursion



Saurabh Shukla (MySirG)

Agenda

- ① What is a recursion?
- ② Recursion Tree | Tracing code
- ③ How to approach recursive solution?
- ④ Few examples

What is a recursion?

- Function calling itself is called recursion
- A recursive method solves a problem by calling a copy of itself to work on a smaller problem.
- It is important to ensure that the recursion terminates.

```
void f1()
{
    = f1();
}
```

```
Main()
{
    main();
}
```

```
void f1() {  
    printf("Hello"),  
    f2(); f1();  
    printf("Bye");  
}
```

```
void f2() {  
    printf("A");  
}
```

Hello A Bye

Hello Hello Hello

Code will run
infinitely

f1()

~~printf("Hello");~~
~~f1();~~
printf("Bye");

~~printf("Hello");~~
~~f1();~~
printf("Bye");

~~printf("Hello");~~
~~f1();~~
printf("Bye");

```

void f1()
{
    int a;
    printf("Enter a number");
    scanf("%d", &a);
    if(a>0)
        f1();
    printf("%d", a);
}

```

-2 105

f1() Enter a number a = 5
 5
 if(5>0)

f1();
 printf("%d", a);

f1() Enter a number a = 10
 10

if(10>0)
 f1();
 printf("%d", a);

f1() Enter a number a = -2
 -2

if(-2>0)
 f1();
 printf("%d", a);

will not run

- Each time the function call itself with a slightly simpler version of the original problem.
- Recursive code is generally shorter and easier to write than iterative code.
- Solution to some problems are easier to formulate recursively.

Recursion Tree

```
int main()
```

```
{  
    int K;  
    K=f1(3);  
    printf("%d",K);  
    return 0;  
}
```

```
int f1(int n)
```

```
{  
    int s;  
    if(n==1)  
        return(1);  
    s=n+f1(n-1);  
    return(s);  
}
```

main()

K
6
int K;
K=f1(3);
printf("%d",K);
return 0;

f1(int n)

n
3
S
6
int s;
if(n==1)
 return(1);
s=n+f1(n-1);
return(s);

f1(int n)

n
2
S
3
int s;
if(n==1)
 return(1);
s=n+f1(n-1);
return(s);

6

1

f1(int n)

n
1
S
 int s;
 if(n==1)
 return(1);
 s=n+f1(n-1);
 return(s);

3

1

How to approach a Recursive Problem?

Write a recursive function to calculate sum of first n natural numbers.

```
int sum(int n)
{
    if(n==1)
        return 1;
    return n + sum(n-1);
}
```

① sum(n) $1+2+3+4+\dots+n$

② $n + \underset{\text{RC}}{\text{sum}(n-1)}$ $1+2+3+\dots+n-1$
 $+n$

③ $\underset{\text{BC}}{\text{sum}(1)}$ 1

$$5+4+3+2+1$$

~~Sum(5)~~

$$4+3+2+1$$

$\hookrightarrow 5 + \cancel{\text{Sum}(4)}$

$$3+2+1$$

$\hookrightarrow 4 + \cancel{\text{Sum}(3)}$

$$2+1$$

$\hookrightarrow 3 + \cancel{\text{Sum}(2)}$

$\hookrightarrow 2 + \cancel{\text{Sum}(1)}$

$\text{Sum}(n)$

$$\begin{aligned} \text{Sum}(n-1) &= 1+2+\dots+n-1 \\ &+ n \end{aligned}$$

