Fibonacci Number

I/p:

F(0) = 0

F(1) = 1

F(2) = 1

F(3) = 2

F(4) = 3

F(5) = 5

$F(n) = F(n-1) + F(n-2)$

o/p: F(5) = 5

F(4) + F(3)

3 2

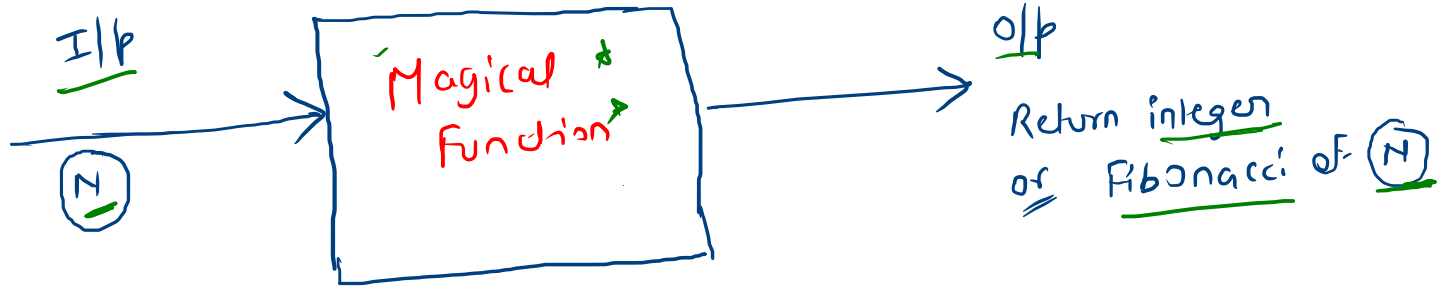
F(4) = 3

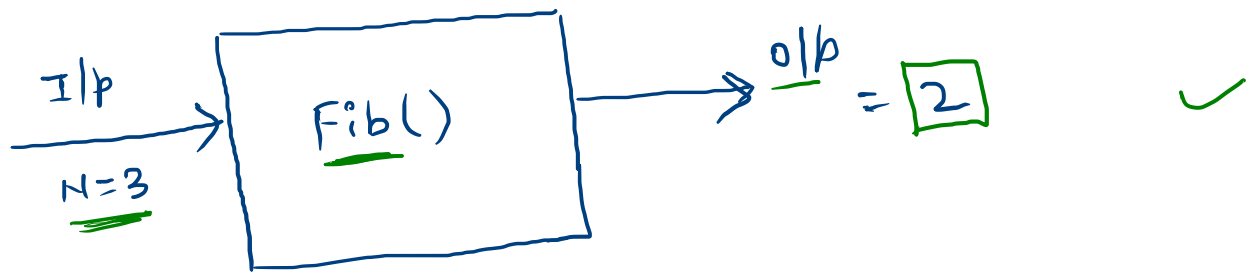
F(5)

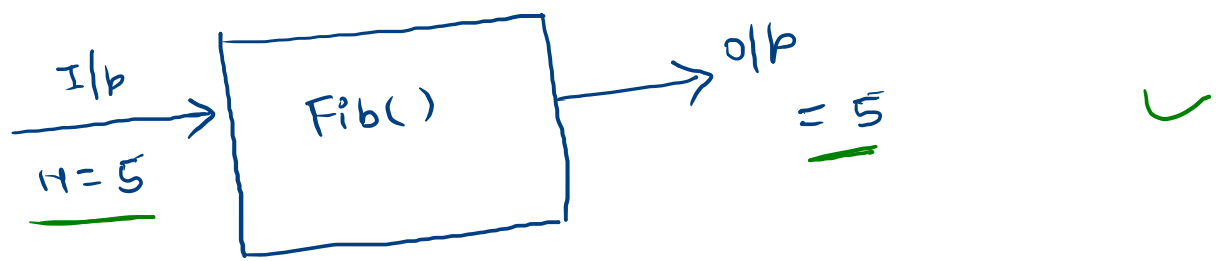
F

n >= 0

Recursive Solution

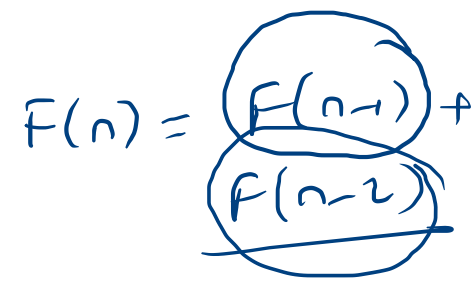






Logical part //

```
int Fib (int n)
{
    // Base condition
    return Fib(n-1) + Fib(n-2);
}
```



Logical part

```
int Fib (int n)
{
    // Base condition
    return Fib(n-1) + Fib(n-2);
}
```

↳ Example

Fib(4) = Fib(3) + Fib(2)

Fib(3) → magical

Fib(2) → magical

Base Condition

Smallest Valid Input → Check output

```
int Fib (int n)
{

}
```

n = 0, 1, 2, 3

Fib(0) = 0 Fib(1) = 1

Fib(2) = Fib(1) + Fib(0)

```
if (n==0) return 0
if (n==1) return 1
```

→

```
if(n<=1) return n;
```

0 0

Recursive solution

```
int Fib (int n)
{
    if(n<=1) return n;
    return Fib(n-1) + Fib(n-2);
}
```

Recursive call