Power of Numbers

I/p : N = 2

o/p : $(N)^R$ Reverse of N is 2

$(2)^2 = 4$

I/p : N = 12

o/p : $(N)^R$ Reverse of N is 21

$(12)^{21} = 864354781$

12 → 21

123 → 321

NAIVE SOLUTION

I/p: (12)^{21} $\quad$ (N)^{R} → Reverse

long long ans = 1;

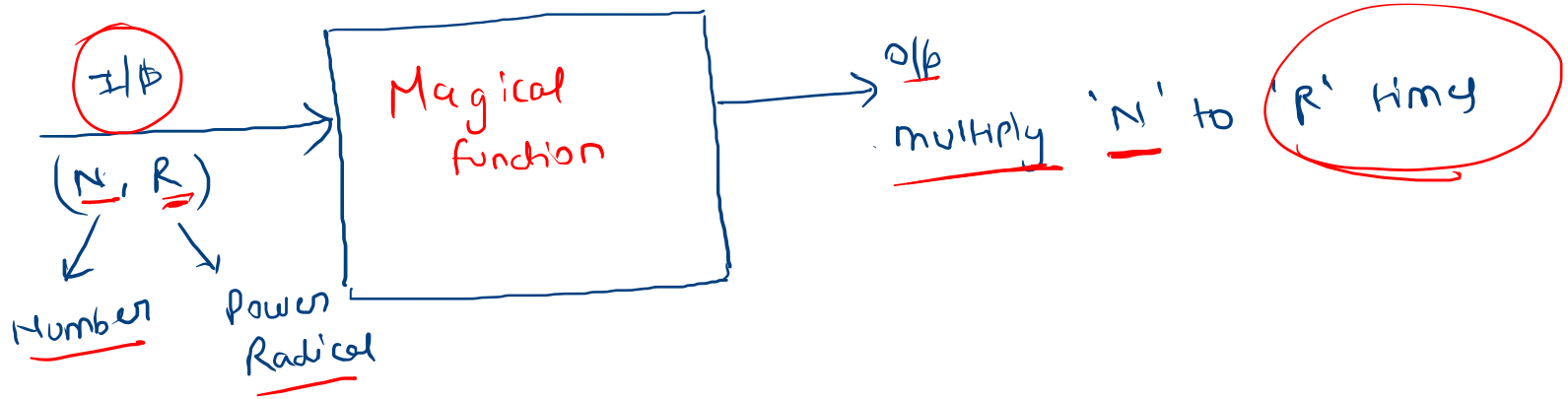
solution:

```
for( int i=1 ; i<=21 ; i++)
{
    ans = ans * 12;
}
cout << ans <<endl;
```

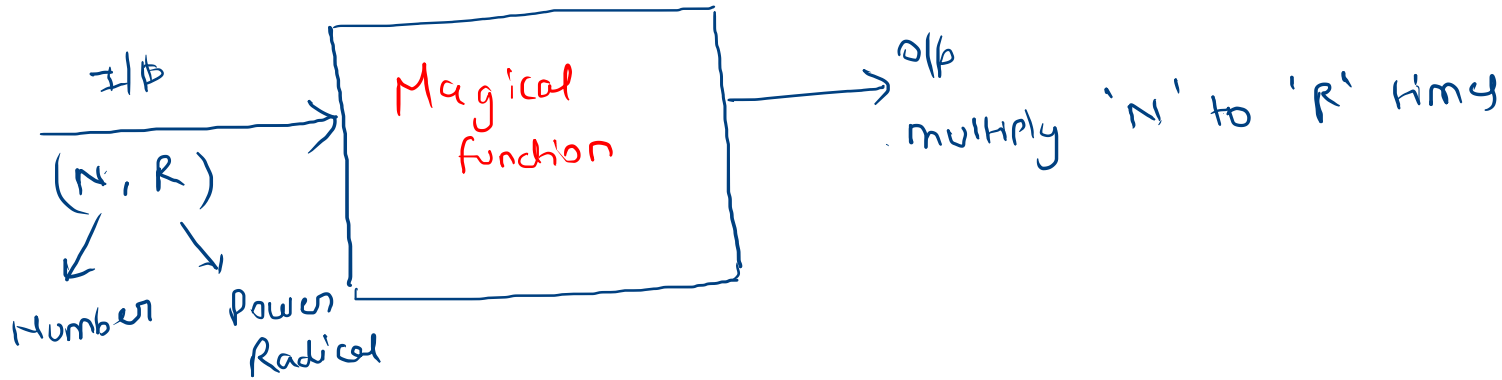
O(N) Time

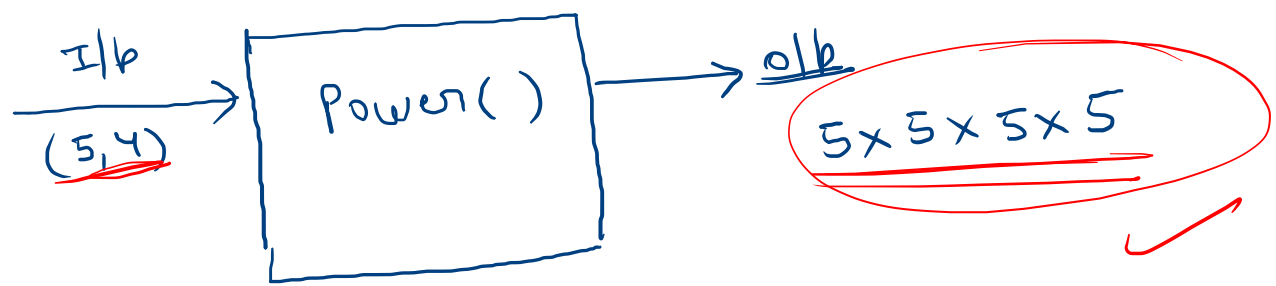
R

Recursive solution



Recursive solution





Now think of Recursive function

Now think of Recursive function

I/p: $(5)^4$

$5 \times 5 \times 5 \times 5$

$5 \times pow(5, 3)$

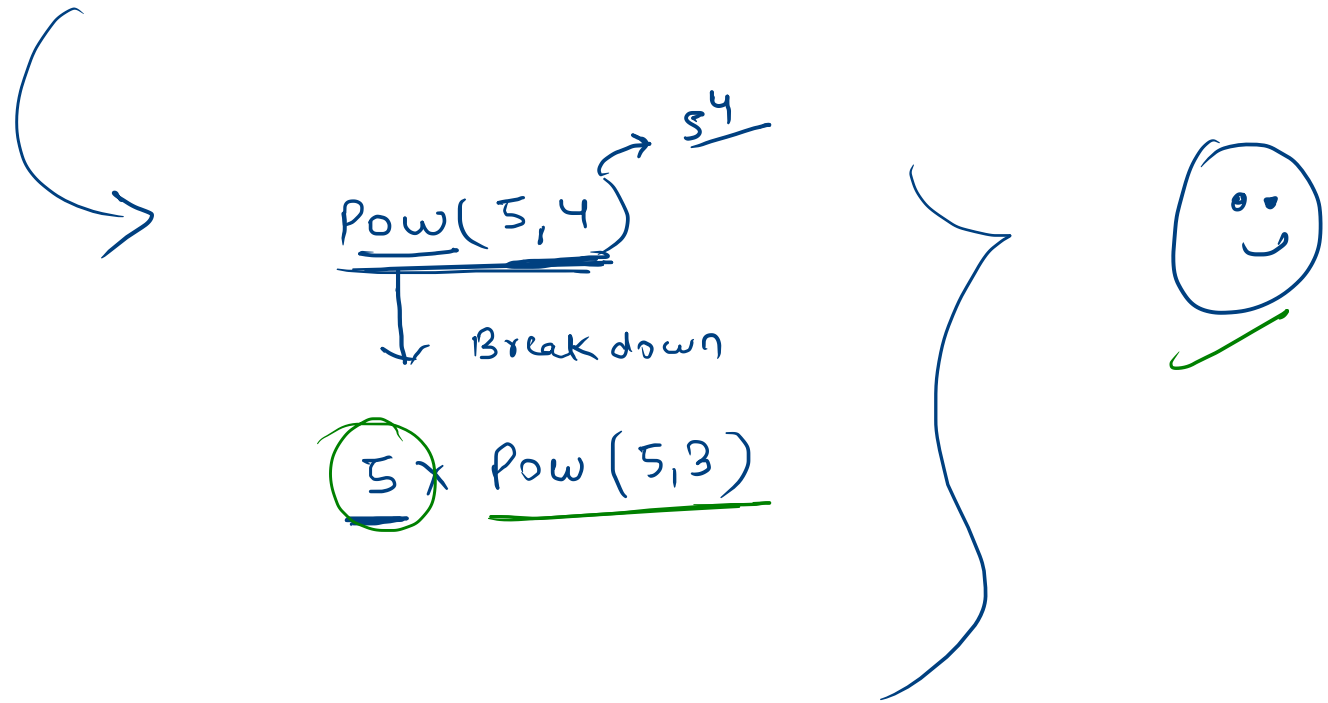
Recursive fun^n

Now think of Recursive function

I/p: $(5)^4$

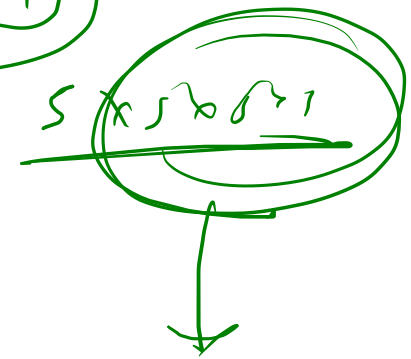
↳ $5 \times 5 \times 5 \times 5$

$5 \times pow(5, 3)$



Recursive function

```
long long Pow( int N , int R)
{
    // Base cond?
    return N * Pow (N, R-1);
}
```



Base cond^n :

Smallest Valid I/p → Check output

I/p : N , R

N = 0, 1, 2, 3 ...

if N = 0, 1, 2, 3 ...

↓ Smallest

R = 0, 1, 2, 3

↓ smallest

```
if ( N==0) return 0;
if ( R==0) return 1;
```

} Base Cond^n

Recursive function

```
long long Pow( int N , int R)
{
    if ( N==0) return 0;
    if( R==0) return 1;
    return N * Pow (N, R-1);
}
```

Recursive Tree

I/p: $(5)^4$

if(n==0) → 0

n==0 → 1

↳ pow(5,4)

o/p 625

5 x pow(5,3)

5 x pow(5,2)

125

5 x pow(5,1)

25

5 x pow(5,0)

1

5

pow(5,4)

↓

5 x pow(5,3)

Now

Given a number and its reverse. Find that number raised to the power of its own reverse.

Note: As answers can be very large, print the result modulo $10^9 + 7$.

↳ Answers are very large

↳ Ans % 10^9+7

↓

int MOD = 1e9 + 7 ;

(1000000007);

Modular Equation

$$(a * b) \% m = ((a \% m) * (b \% m)) \% m$$

$$(a + b) \% m = ((a \% m) + (b \% m)) \% m$$

Modular Equation

$$(a * b) \% m = ((a \% m) * (b \% m)) \% m$$

$$(a + b) \% m = ((a \% m) + (b \% m)) \% m$$

Below equations are valid

```
(a * b) % m = ((a % m) * (b % m)) % m
(a + b) % m = ((a % m) + (b % m)) % m

// m is added to handle negative numbers
(a - b + m) % m = ((a % m) - (b % m) + m) % m

But,
(a / b) % m may NOT be same as ((a % m)/(b % m)) % m

For example, a = 10, b = 5, m = 5.
   (a / b) % m is 2, but ((a % m) / (b % m)) % m
                              is not defined.
```

Recursive function

```
long long Pow( int N , int R)
{
    if ( N==0) return 0;
    if( R==0) return 1;
    return N * Pow ( N, R-1);
}
```

Ans

a * b

(a*b) %. M

(a%.m * b%.m) %.m

Ans %. MOD

(a * b) %. MOD

(a%.MOD * b%.MOD) %.MOD

Example :

$(5)^6$

↳ $5 \times pow(5,5)$

↳ $5 \times pow(5,4)$

↳ $5 \times pow(5,3)$

↳ $5 \times pow(5,2)$

↳ $5 \times pow(5,1)$

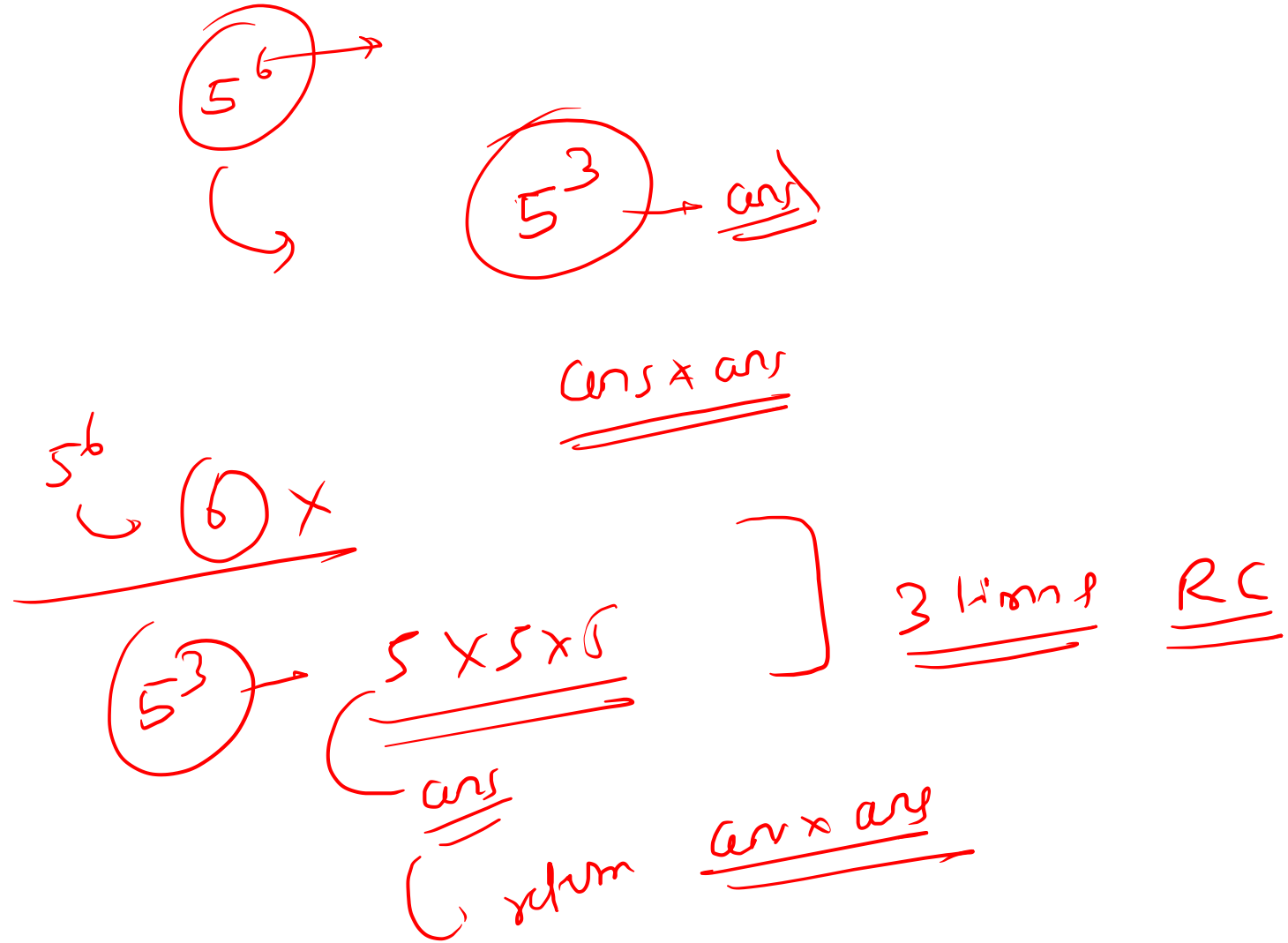
↓

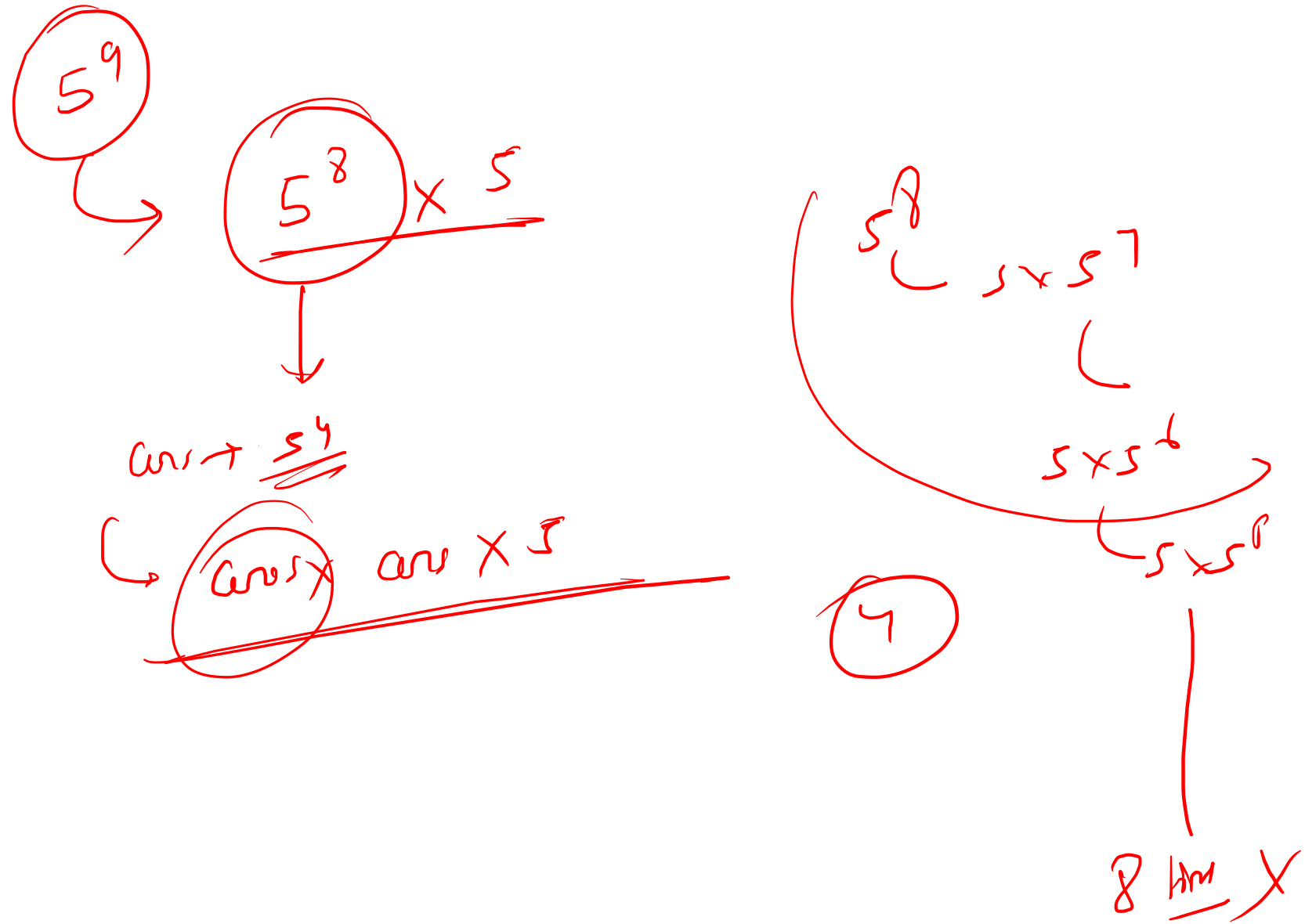
$5 \times pow(5,0)$

6 time

DP

Optimisation over Recursive code


5^6
5^3
ans
ans * ans
5^6
6 X
5^3
5 X 5 X 5
3 times RC
ans
return ans * ans


5^9
$5^8 \times 5$
ans → 5^4
ans × 5
ans × 5
5^8
5×5^7
5×5^6
5×5^5
4
8 times

Now, we will optimise solution

Ik: $(5)^6$

↳ $5 \times 5 \times 5 \times 5 \times 5 \times 5$

Optimal

↳ if (R % 2 == 0)

↳ long long ans = $(5)^3$ → $(N)^{R/2}$

return ans * ans ;

Ex $(5)^9$

↳ if (R % 2 != 0)

↳ long long ans = $(5)^4$ → $(N)^{(R-1)/2}$

return ans * ans * 5