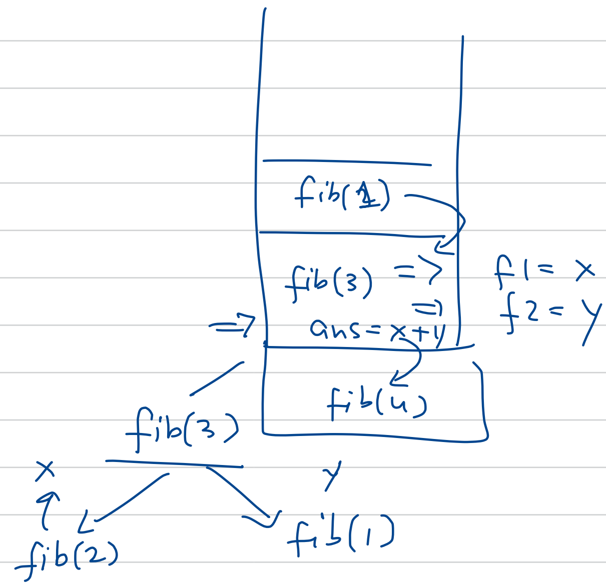
Maths

Recursion Doubts

I

```
fib ( int n ) {
  // —x—

  int f1 = fib(n-1)
  int f2 = fib(n-2)
  int ans = f1 + f2;
  return ans
}
```

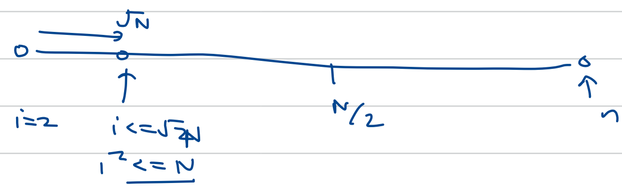


II Prime No using Rec

III Binary to Decimal

IV GCD

V TOH Steps



Loop → Rec

maintain this info in the call itself

↳ fn parameter

```
for(i=2; i*i<=n; i++){
    if(n%i==0)
        return false;
}
return true;
```

=> 2

```
boolean isPrime (int n, int i) {

    if ( i*i > n) {
            return true;
    }

    if (n % i == 0){
            return false;
    }
    return isPrime(n, i+1);
}
```

div -> 2

15
↓
2

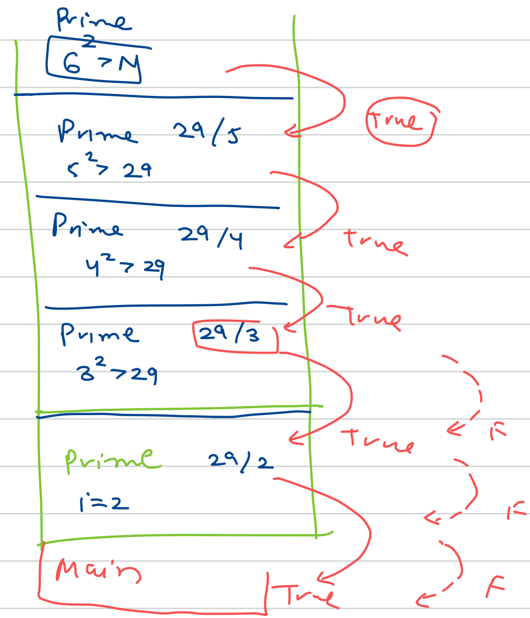
```java
static boolean isPrime(int n,int i){
    if(i*i>n){
        return true;
    }
    if(n%i==0){
        return false;
    }
    return isPrime(n,i+1);
}
```

N = 29

27%3 == 0

5

$\sqrt{29}$



Number Conversion

- Binary To Decimal
- Decimal To ~~Octal~~ Binary

$N = 1\,0\,1\,1\,\cancel{0}$

powers of 2: $2^4\ 2^3\ 2^2\ 2^1\ \cancel{2^0}$

16 8 4 2 1

p

ans = 0 + 16 + 4 + 2 = $\boxed{22}$

Code

→ Extract digit

→ gen powers of 2

p = $2^0 = 1$

ans = 0

while (N > 0) {

Logical steps

last_digit = N % 10

ans = ans + last_digit * p;

N = N/10; //remove LD

p = p * 2;

}

Sout (ans);

Rec

rem part

1011 | $\boxed{0}$

sub prob

one step

11 ×2 + 0×1

= 22+0

= 22

8 4 2 1

10 11

Breakdown

N

f(N/10) N%10

10 11 | 1

p = 2° = 1

2 (Rec ~~Rec~~ work) + my work = N%10

Not exactly addition (Concatenating)

2 (2³ 2² 2¹ 2⁰) 2

1011 = 11

1

2^4 2^3 2^2 2^1 2^0

1 0 1 1 0

2^4 2^3 2^2 2^1 2^0

10110 = 22

+ 1 0 got added

10111

```
static int binaryToDecimalRec(int n){
    //base case
    if(n==0){
      return 0;
    }
    int myWork =  n%10;
    int recWork = binaryToDecimal(n/10);
    int ans = 2*recWork + myWork;
    return ans;
}
```

BD	0	m = 1
BD	1	2×0 + 1 = 1
BD	10	mw = 0 ; 2×1 + 0
BD	101	mw = 1 ; 2×2+1
BD	1010	mw = 0 ; 2×5 + 0
BD =	10101 0	mw = 1 ; 2×10 + 1 ; ×2
BD	n = 101010	mw = 0 ; 2×21 + 0

32 16 8 4 2 1

101010 = 42

2x21 + 0 = 42

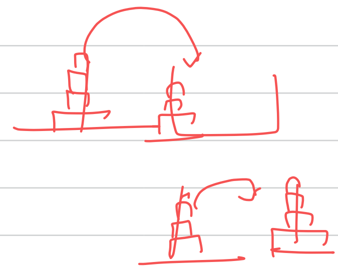
toh (n) {

x = toh(n-1) ;

+ 1 step

x = toh(n-1)

}



toh(n) { if(n==0) { return 0 }

→ $\underbrace{2 * toh(n-1) + 1}$

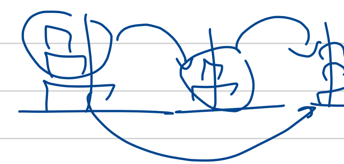
}

3

3 steps

+1

+3 steps



= 7 steps

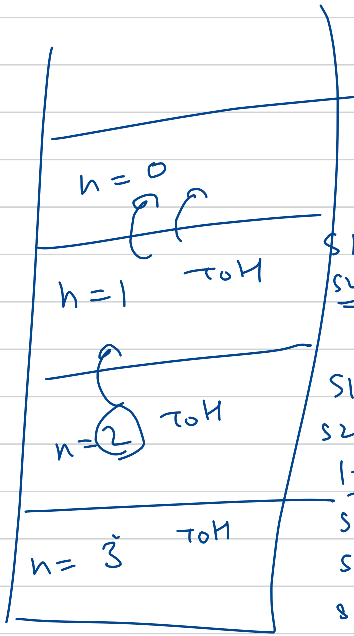
Two calls

equal?

1. $\Rightarrow$ $toh(n-1) + toh(n-1) + 1$
2. $\Rightarrow$ $2 * \boxed{toh(n-1)} + 1$

$toh(n-1)$

$toh(n-1)$



$s1 = 0$

$s2 = 0$

$= 0+0+1$

$s1 = 1$

$s2 = 1$

$1+1+1$

$s1 = 3$

$s2 = 3$

$s1+s2+1$

$3+3+1$

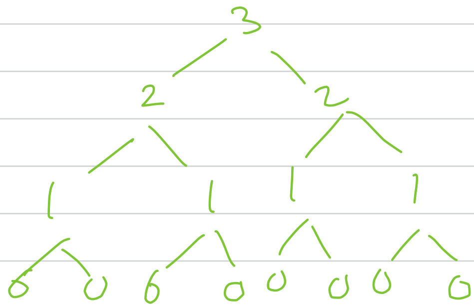
$= 7$

$toh(1)$

$toh(1)$

$toh(3)$

$toh(2)$ $toh(2)$



7

3 $2 \times 3 + 1$

2 $= 2(1) + 1$

1 $= 2 \times 0 + 1$

0 0

GCD

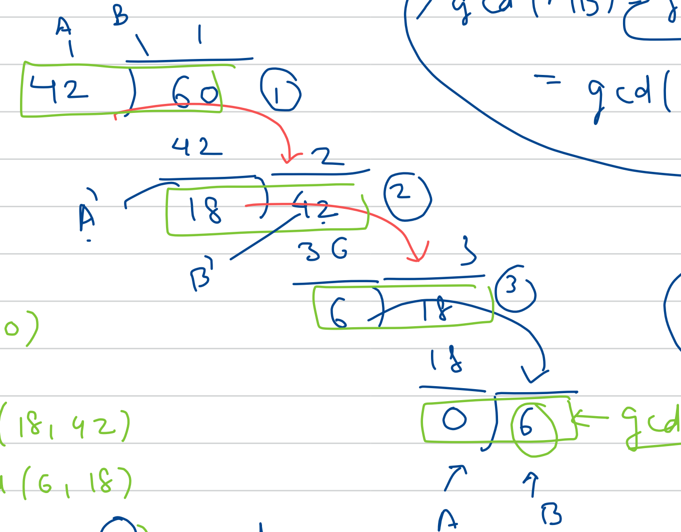
$B = 60$

$A = 42$

gcd

$$\gcd(A, B) = \gcd(A', B')$$
$$= \gcd(B \% A, A)$$

Pen & Paper Method



Atmost 4 steps

Lesser steps

Brute

42 steps

1
2
⋮

min(42,60)

$\gcd(42, 60)$

$= \gcd(18, 42)$

$= \gcd(6, 18)$

$= \gcd(0, 6) \rightarrow$ gcd

Rec formula

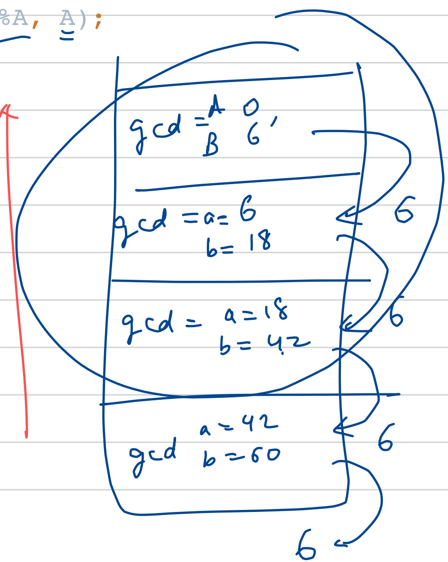
- $gcd(A, B) = gcd(B \% A, A)$

- if $(A == 0)$ return B;

Euclid's Algorithm

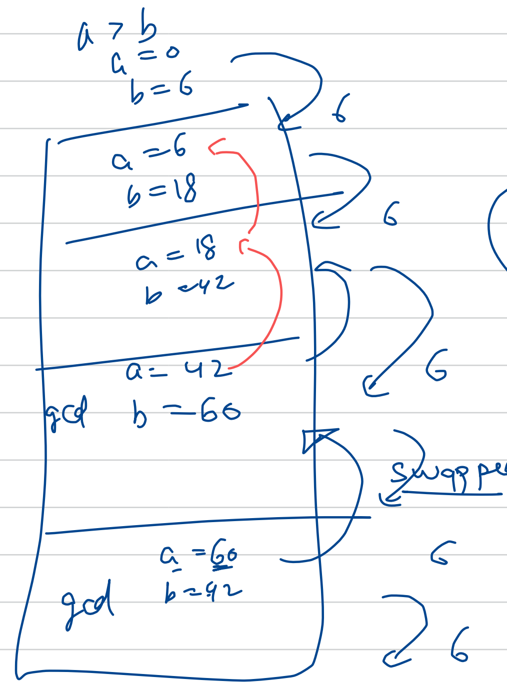
```
static int gcd(int A,int B){
    if(A==0){
        return B;
    }
    return gcd(B%A, A);
}
```

42, 60



Log A

∝ Log N



1 additional step

Logarithm

mathematical fn

$\sin(x)$
$\cos(x)$
$\log(x)$

$$\log_B N = val$$

$$\Rightarrow B^{val} = N$$

$\log_{10} 1000 = 3$ $\quad 10^3 = 1000$

$\log_5 125 = 3$

$\log_5 120 = 2.99..$

$e = 2.7$

Base

Base - 2

$\log_{2} 100 = 6.7$

$\ln$ Natural log

$\log_e N$

$\log_2 1000 = 9.xxx \approx \boxed{10}$

$2^{10} = 1024$

$\log_2 1024 = 10$

$\log_B N^K = K \log_B N$

$\log_2 1000\,000 = \log_2 (1000)^2$

$= 2 \log 1000$

$= 20$

100 $\downarrow$ 50 $\downarrow$ 25 $\downarrow$ 12 $\downarrow$ 6 $\downarrow$ 3 $\downarrow$ 1 $\downarrow$ 0

N = some val

while (N > 0) → exponential decay

{ N = N/2 }

7–8 steps

$\log_2 100$ = 7 steps

gcd → after every 2 steps A → less than A/2

$\sim \log_2 A$ steps

approximation

$A \downarrow A/2 \downarrow A/4 \downarrow A/8 \downarrow 0$

18 r

Time of execution → No of steps → how Big the input is

10 , 77

1
2
.
.
.
10

10 steps

10
↓
0

log 10 steps = 3 steps

CPU $\rightarrow$ Capability

$1s = 10^8$ steps

gcd $\quad 10^{18}$, long $\quad 10^{18} + 5$

A $\quad$ B

1
2
⋮
10^{18}

10^{18} steps
N steps

10^8 steps $\rightarrow$ 1 second

$$\boxed{1 \text{ step} \rightarrow \frac{1}{10^8} \text{ seconds}}$$

10^{18} steps $\rightarrow \dfrac{10^{18}}{10^8}$

$= 10^{10}$ seconds

317 years

Euclid

Log N steps

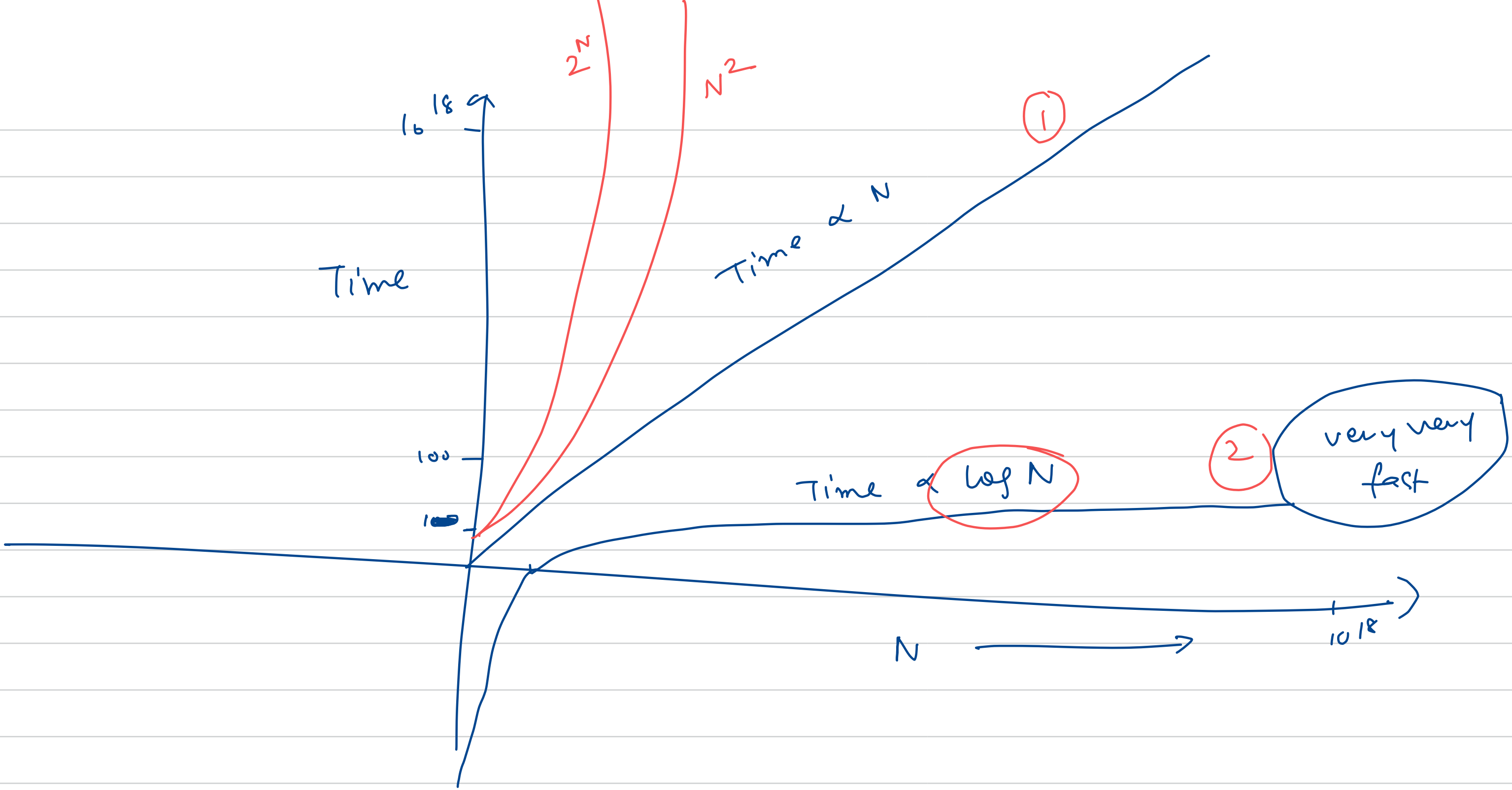
$= \log 10^{18}$

$= \log_2 (1000)^6$

$= 6 \log 1000 \simeq 60$ steps

60 steps $\rightarrow 60 \times 10^{-8}$ seconds

$= 6 \times 10^{-7}$ seconds



2^N
N^2
10^{18}
Time
Time ∝ N
1
100
10
Time ∝ log N
2
very very fast
N
10^{18}

$\Rightarrow$ AP, GP, BC, PC

$\underbrace{\text{AP, GP, BC, PC}}$

Computing Space & Time Compute

10.40

Airthmetic Progression

$$\overset{25}{\downarrow}$$

$$\Rightarrow \underset{1}{5} \overset{d}{\longleftrightarrow} \underset{2}{9} \overset{d}{\longleftrightarrow} \underset{3}{13} \overset{d}{\longleftrightarrow} \underset{4}{17}, \underset{5}{21}, \underset{6}{25}$$

a

+4   +4   +4   +4   +4

$$T_6 = a + (6-1)d$$
$$= 5 + 5 * 4$$
$$= 5 + 20$$
$$= 25$$

$$\boxed{S_N = \frac{n}{2}\left(2a + (n-1)d\right)}$$

$$= \frac{n}{2}\left(a + \underline{a + (n-1)d}\right)$$

$$\boxed{S_N = \frac{n}{2}(a + a_n)}$$

$$S_6 = \frac{6}{2}(2 \times 5 + 5\ 4)$$
$$= 3(10 + 20)$$
$$= 90$$

$$S_8 = \frac{8}{2}(6 + 20)$$
$$= 4(26) = \boxed{104}$$

$$\boxed{T_N = a + (n-1)d}$$

$$6, \ldots\ldots, 20$$

a T_8 a_8

$20 = -6 + (7)d$

$\Rightarrow \frac{20-6}{7} = d \Rightarrow 2$

$\overset{5}{\longleftrightarrow}$

17, 22, 27 .....

5^{th} $T_5 = a + (n-1)d$

$= 17 + 4 \times 5$

$= 37$ ✓

Geometric Progression → every term bears a constant ratio (r) to prev term

$r = 2$

$a = 3, 6, 12, 24$

$\frac{6}{3} = 2$ $\frac{12}{6} = 2$ $\frac{24}{12} = 2$

r^0

$a, \ ar, \ ar^2, \ ar^3, \ \ldots\ldots \ ar^{n-1}$

1st $\qquad\qquad$ nth

$T_n = ar^{n-1}$

Finite GP

$r = 1 \qquad a, a, a, a \ldots\ldots a$

$S_n = an$

$5, 5, 5, 5 \ldots 5$ $\underbrace{\qquad}_{5n}$

$r \neq 1 \qquad a, ar, ar^2 \ldots\ldots ar^n$

$$\boxed{S_n = \frac{a(r^n - 1)}{(r - 1)}}$$

5, 10, 20, 40, 80 ...

5 Terms

$a = 5$

$r = 2$

works :)

$$S_5 = \frac{5(2^5 - 1)}{2 - 1}$$

$$= 5(31)$$

$$= 155$$

Infinite GP

$n \to \infty$

$r < 1$

$2, 1, \frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \frac{1}{16} + \ldots \infty$

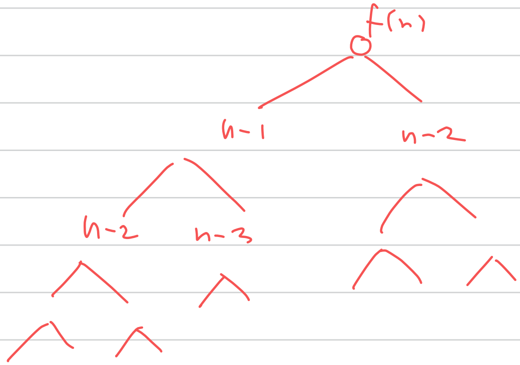
$$S_\infty = \frac{a}{1-r} \qquad \frac{2}{1-\frac{1}{2}} = 4$$

$r \geq 1$

$2, 4, 8, 16, \ldots$

$S_\infty \Rightarrow \infty$

divergent



1

2

4

8

⋮

2^{n-1}

$1+2+4+8+\dots 2^{n-1}$

$= \frac{a(r^n-1)}{r-1}$

$= \frac{1(2^n-1)}{2-1}$

$= 2^n$

Time $\propto 2^N$

N = 10

1024 steps

N = 100

2^{100} steps

Binomial coefficient

$$^{N}C_{R} = \frac{N!}{(N-R)!\,R!}$$

Combination

$^{N}C_{R}$ signify: No of ways of choosing R object out of collection of N objects

Ex ⇒ 5 Boys A, B, C, D, E

Choose 2 Boys

$$^{5}C_{2} = \frac{5!}{(5-2)!\,2!} = \frac{5!}{3!\,2!} = \frac{\cancel{3!} \times \overset{2}{\cancel{4}} \times 5}{\cancel{3!} \times \cancel{2!}} = 10$$

AB	BC	CD	DE
AC	BD	CE	
AD	BE		
AE			

$4 + 3 + 2 + 1 = \boxed{10 \text{ ways}}$

Permutation : Choose + Arrange

2 Boys out of 5 Boys for the post Captain & V Captain.

2 options

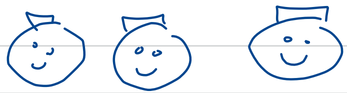
$^{N}P_{R} = {}^{N}C_{R} * R!$

$= 10 \times 2!$

$= 20$

$$^{N}P_{R} = {}^{N}C_{R} \times R!$$

$$= \frac{N!}{(N-R)! \times \cancel{R!}} \cancel{R!} = \frac{N!}{(N-R)!}$$



B1 B2 B3

$^{N}C_{3}$ ways $\times 3!$

3 ppl → 3! ways

C1	C2	C3
100%	50%	20%
B1	B2	B3
B1	B3	B2
B2	B1	B3
B2	B3	B1
B3	B1	B2
B3	B2	B1

6 ways

B1, B2, ---- B10

⇒ Choosing   Combination

⇒ hint : → Arrangement

B1 B2 →

B2, B1 → Arrangement

5 Boys

3 girls

Product Rule

⇒ Choose 2 Boys & 1 girl

$^5C_2 \times {}^3C_1$

(X) → 3

(A, B)

→ G1

G2

G3

Sum Rule

⇒ Choose either 2 Boys or 1 girl.

$^5C_2 + {}^3C_1$

$= 10 + 3 = 13$ ✓

Pascal's Triangle

I

$^4C_2 = \frac{4 \times 3}{2} = 6$

forever term

$^NC_R = \frac{N!}{(N-R)!\,R!}$

	R: 0	1	2	3	4
N: 0	$^0C_0$ 1				
1	$^1C_0$ 1	$^1C_1$ 1			
2	$^2C_0$ 1	$^2C_1$ 2	$^2C_2$ 1		
3	$^3C_0$ 1	$^3C_1$ 3	$^3C_2$ 3	$^3C_3$ 1	
4	$^4C_0$ 1	$^4C_1$ 4	$^4C_2$ 6	$^4C_3$ 4	$^4C_4$ 1

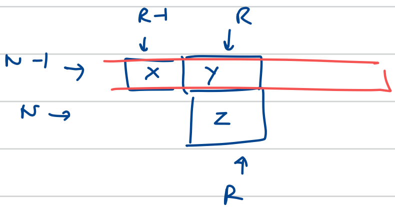
```
for( i=0; i<=N; i++) {
    for( j=0; j<=i; j++) {
        print -> iCj
    }
}
```

$\Rightarrow \frac{i!}{(i-j)!\,j!}$ $\boxed{\text{Loop}}$

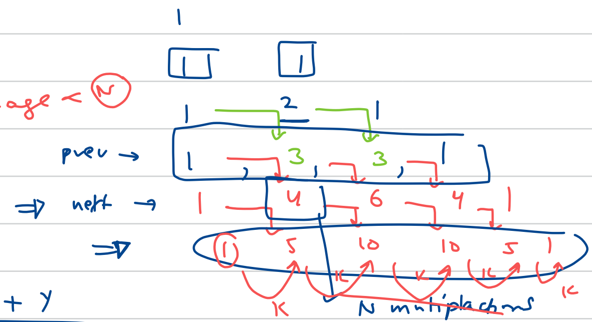
1

Q

better than prev



Storage < N



$z = x + y$

$$\boxed{{}^{N}C_{R} = {}^{N-1}C_{R-1} + {}^{N-1}C_{R}}$$

prev Row

term ⇒ 1 5 10 10 5 1

${}^{N}C_{R-1} \leftarrow {}^{N}C_{R}$

Next = K prev

$\Rightarrow {}^{N}C_{R} = K \cdot {}^{N}C_{R-1}$

Row − col + 1 / col

$\Rightarrow K = \left(\frac{N-R+1}{R}\right)$

③ Less Time, less Storage

$$ {}^{N}C_{R} = f\left(\cancel{R}, {}^{N}C_{R-1} \right) $$

$$ {}^{N}C_{R} = k \cdot {}^{N}C_{R-1} $$ $\boxed{\text{Next} = K \; \text{prev}}$

$$ \Rightarrow \frac{\cancel{N!}}{(N-R)!\,R!} = k \frac{\cancel{N!}}{(N-R+1)!\,(R-1)!} $$

$$ \Rightarrow \frac{1}{\cancel{(N-R)!}\; R \times \cancel{(R-1)!}} = \frac{K}{(N-R+1)\cancel{(N-R)!}\;\cancel{(R-1)!}} $$

$$ \Rightarrow K = \left(\frac{N-R+1}{R}\right) $$

Row → col → cont

R → col.

1 multiplication with no extra space

```
for ( i=0; i<=n; i++ ){
    term = 1;
    for (r=1, r<=i; r++){
        print(term)
        term = k . term
    }
}
```

$$ \left(\frac{i-r+1}{r}\right) $$

N = 4

R = ~~0~~, 1, 2, 3, 4

term ① ④ ⑥ ④ ①

$\frac{4-1+1}{1}$ $\left(\frac{4-2+1}{2}\right) 4 \times 2$ $\left(\frac{4-3+1}{3}\right) \cancel{6}\,2$

$4 - \frac{4+1}{4} \times 4 = ①$

loop → Rec

gcd (A, B)

||

gcd (B % A , A)

↳ Rec ↓ Space

↓

to think of rec

for (, a > 0 ;)

[a = B % A
b = A]