Lec-5
29/08/24 2:32 PM

2^{56}

$\frac{5}{V}$ $\cancel{\frac{21}{C}}$ Unicode

$\frac{19}{C}$

$\frac{2}{S \cdot V}$

2^{16} →

2^{8}

a	97	A	65	0	48
b	98		66	1	49
c	99	B	67	2	50
d	100	C			
z		Z		9	

Ch=Ch+1

-128<128
127<128
126<128

```java
public static void main(String[] args) {
	// TODO Auto-generated method stub
	for (byte b = 0; b < 128; b++) {
		System.out.println(b);
	}
}
```

126
127
(128)

1600 0000

0
1
125
126
127
-128 ✓

$\frac{4!}{2!\,2}$.

nCR → (0 to n)

nCR

$\left[{}^{n}C_{R}\right]$

$^{0}C_0$ → 0
$^{1}C_0$ $^{1}C_1$ ← → 1
$^{2}C_0$ $^{2}C_1$ $^{2}C_2$ ← → 2
$^{3}C_0$ $^{3}C_1$ $^{3}C_2$ $^{3}C_3$ ← → 3
$^{4}C_0$ $^{4}C_1$ $^{4}C_2$ $^{4}C_3$ $^{4}C_4$ ← → 4
← → 5

$\frac{5!}{2!\,3!}$

n=150
R=20

A BCDE

$^{N}C_R$

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

$^{150}C_{20}$

ACD

BEP

ACE

2 ABC

$^{5}C_3 = \frac{5!}{(5-3)!\,3!} = \frac{5!}{2!\times 3!} = \frac{5\times 4\times \cancel{3!}}{2!\times \cancel{3!}} = 10$ ✓

$^{n}C_R \rightarrow \left[{}^{n}C_{R+1}\right]$

$^{N}C_R = \frac{N!}{(N-R)!\,R!}$ — ①

$^{N}C_{R+1} = \frac{N!}{(N-(R+1))!\,(R+1)!}$ — ⑪

⑪ ÷ ①

$\frac{^{N}C_{R+1}}{^{N}C_R} = \frac{\cancel{N!}}{(N-R-1)!\times(R+1)!} \times \frac{(N-R)!\times R!}{\cancel{N!}}$

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

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

$\frac{p}{q} \div \frac{R}{S} = \frac{p}{q}\times\frac{S}{R}$

$(N-R)! = (N-R)(N-R-1)!$

$N! = N\times(N-1)$

$\frac{^{n}C_{R+1}}{^{n}C_R} = \frac{n-R}{R+1}$

$\boxed{^{n}C_{R+1} = \left(\frac{n-R}{R+1}\right)\times {}^{n}C_R}$

0
1 0
2 0 1
Row → 3 0 1 2
4 0 1 2 3
5

Row ← $^{n}C_{R+1} = \frac{(n-R)\times {}^{n}C_R}{(R+1)}$
i ←

$^{n}C_R = \left[\frac{(Row-i)\times {}^{n}C_R}{i+1}\right]$

```java
int i = 0; int ncr=1;
		while (i < star) {
			System.out.print(ncr+" ");
			i++;
		}
```

```java
public static void main(String[] args) {
	// TODO Auto-generated method stub
	Scanner sc = new Scanner(System.in);
	int n = sc.nextInt();
	int row = 0;
	int star = 1;
	while (row < n) {
		// star
		int i = 0;
		int ncr=1;
		while (i < star) {
			System.out.print(ncr+" ");
			ncr = ((row - i) * ncr) / (i + 1);
			i++;
		}
		// next row ki prep
		System.out.println();
		row++;
		star++;
	}
```

ncr=4
6
4

$^{n}C_1 = \frac{(4-0)\times 1}{0+1}$

ncr = $\frac{(4-1)\times 4}{1+1}$ 2

$\frac{(4-2)\times 6}{2+1}$ 2

$\frac{4-3\times 4}{(3+1)}$

0 1
1 1 1
2 1 2 1
3 1 3 3 1
4 $\boxed{1\ 4\ 6\ 4\ 1}$
1 5 10 10 5 1

1 4 6 4 1

32145

2143

2187

2132 X

3214 5 = 1 2 5 4 3 ✓
5 4 3 2 1 ← 5 4 3 2 1

(12543)

30000

$32145 = 3\times10^{5-1} + 2\times10^{4-1} + 1\times10^{3-1} + 4\times10^{2-1} + 5\times10^{1-1}$

= 30000
2000
100
40
5
32145 ✓

3 2 1 4 5
5 4 3 2 1

3214
321
32
3
0

$x += 1 \rightarrow \boxed{x = (int)(x+1)}$

$1\times10^{5-1} + 2\times10^{4-1}$
$+ 3\times10^{3-1} + 4\times10^{2-1}$
$+ 5\times10^{1-1}$

= 10000
2000
300
40
5
12543 ✓

6 5 4 3

sum=0 pos=1

32145

while(n>0){
int rem=n%10
sum=sum + pos×10^rem-1
pos++;
n=n/10;
}