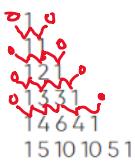


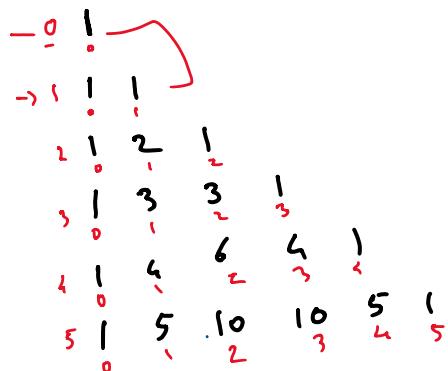
Lec13

04 February 2026 10:55 AM



nC_2

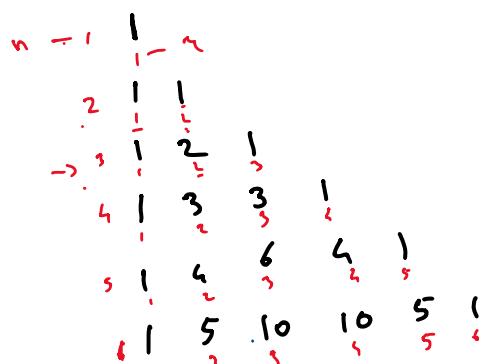
A B C D E
AB
AC
AD
AE



$$\frac{\text{row} - n}{\text{col}} = \frac{(n!)}{z!(n-z)!+1}$$

$$\frac{1}{1 \times 1 + 1} \\ 1+1 \\ = 2 \\ \frac{1}{2}$$

num = 1

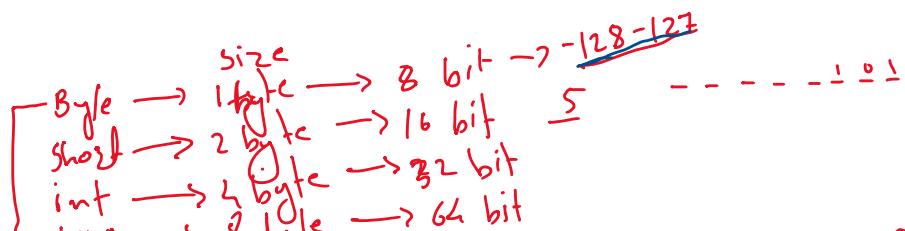


$$\frac{n!}{z!(n-z)!+1} \quad \text{num = 1}$$

$$\frac{1}{1 \times 1} \quad \frac{2}{1 \times 1 + 1} \quad \frac{2}{2 \times 0 + 1} \\ 1 \\ \frac{1}{2} \quad \frac{2}{2} \quad \frac{2}{1} \\ = 1 \quad = 2$$

$$\frac{1 \times 2 \times 1 + 1}{1 \times 2 \times 1} \\ 2 \times 1 \\ = 3$$

$$\text{num} = \text{num} + (\text{row} - \text{col}) / (\text{col} + 1)$$



short → 2 byte
 int → 4 byte → 32 bit
 long → 8 byte → 64 bit

[float
double]

[char
Boolean]

$$\begin{array}{r}
 2^8 \\
 \hline
 2 \mid 5 & -1 \\
 2 & 2 & -0 \\
 \hline
 2 & 1 & -1 \\
 \hline
 0
 \end{array}
 \quad \begin{array}{l}
 \uparrow \\
 2^8 \\
 \hline
 256
 \end{array}$$

[+ve
small +ve
large]

$$\begin{array}{c}
 \begin{array}{ccccccccc}
 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\
 \hline
 -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
 \end{array} \\
 \rightarrow \\
 \text{MSB}
 \end{array}$$

Is complement

$$\begin{array}{r}
 1 1 1 1 1 1 1 \\
 \hline
 0 0 0 0 0 0 0
 \end{array}$$

$$\begin{array}{r}
 0 1 0 0 0 0 1 0 \\
 \hline
 + 1 1 1 1 0 1 1 \\
 \hline
 1 1 1 1 1 1 1 0
 \end{array}
 \quad \begin{array}{l}
 1 \% 2 \\
 1 / 2
 \end{array}$$

$$\begin{array}{r}
 2 \mid 478 & - 0 \\
 \hline
 2 \mid 239 & - 1 \\
 \hline
 2 \mid 119 & - 1 \\
 \hline
 2 \mid 59 & - 1 \\
 \hline
 2 \mid 29 & - 0 \\
 \hline
 2 \mid 14 & - 1 \\
 \hline
 2 \mid 7 & - 1 \\
 \hline
 2 \mid 3 & - 1 \\
 \hline
 2 \mid 1 & - 1 \\
 \hline
 0
 \end{array}$$

$$\begin{array}{r}
 1 1 1 0 1 1 1 1 0
 \end{array}$$

$$\begin{array}{r}
 1 1 0 1 1 1 1 0 \\
 \hline
 0 0 1 0 0 0 1 1 \\
 \hline
 0 0 1 0 0 1 0
 \end{array}$$

$$\begin{array}{l}
 2 \% 2 = 0 \\
 2 / 2 = 1
 \end{array}$$

2 1 0

1. Widening Casting (Automatic/Implicit)

Order (Small to Large): byte → short → int → long → float → double

2 Narrowing Casting (Manual/Explicit)

Order (Large to Small): double → float → long → int → short → byte