

Language :- C++ (Basics must be clear)

Input & Output

The following lines of code make input & output more efficient.

```
sync_with_stdio(0);  
cin.tie(0);
```

In work faster than endl, because endl always causes a flush operation

To read a whole line

```
string s;  
getline(cin, s);
```

To read & write from file :-

```
freopen ("input.txt", "r", stdin)
```

```
freopen ("output.txt", "w", stdout)
```

Modular Arithmetic

$x \bmod m = \text{remainder of } x \text{ divided by } m$

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

$$(a-b) \bmod m = (a \bmod m - b \bmod m) \bmod m$$

$$(a \cdot b) \bmod m = (a \bmod m \cdot b \bmod m) \bmod m$$

Note :- $x = x \bmod m$, if $x < 0$ then $x + m$

Floating point number

double :- 64 bit

long double :- 80 bit

To compare floating point number with equality operator is risky, due to precision error.

So,

$$|a - b| < 1e-9$$

The value of x with 9 decimal places

```
printf( "%.9f", x);
```

Jo typedef

h wo h baas
datatype k saath
kaam karta h

Shortening Code

typedef :- shorter name to datatype **(only)**

```
typedef long long ll;
```

It can also be used with more complex types.

```
typedef vector<int> vi; (compiler is responsible)
```

Macros :- A macro means that certain string in the code will be changed before the compilation.

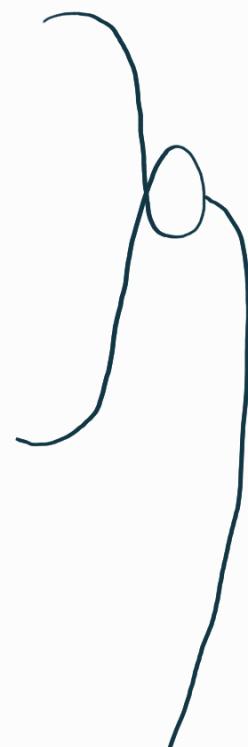
```
#define pb push_back
```

```
#define mp make_pair
```

(Pre-processor is responsible)

Use of macros

```
v.push_back(make_pair(y1,x1));  
v.push_back(make_pair(y2,x2));  
int d = v[i].first + v[i].second;
```

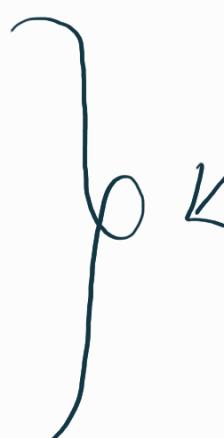


After defining macros

```
#define F first  
#define S second  
#define PB push_back  
#define MP make_pair
```

The code will be

```
v.PB(MP(y1,x1));  
v.PB(MP(y2,x2));  
int d = v[i].F + v[i].S
```



To shorten loops

```
for(int i=1; i<=n; i++) {  
    search(i);  
}
```

Macro: #define REP(i,a,b) for(int i=a; i<=b; i++)

```
REP(i,1,n) {  
    search(i);  
}
```

To perform short operation

define SQ(a) (a)*(a)

Mathematics

Sum Formulas

$$1+2+3+4+\dots+n = \frac{n(n+1)}{2}$$

$$1^2+2^2+3^2+4^2+\dots+n^2 = \frac{n(n+1)(2n+1)}{6}$$

Arithmetic Progression

Sequence of numbers where the difference between any two consecutive numbers is constant.

For eg :- 3, 7, 11, 15

$$\underbrace{a+\dots+b}_{n \text{ numbers}} = \frac{n(a+b)}{2}$$

Geometric Progression

Ratio between any two consecutive numbers is constant.

For ex:- 3, 6, 12, 24

Sum of GP

$$a + ak + ak^2 + \dots + b = \frac{bk - a}{k - 1}$$

$$3 + 6 + 12 + 24 = \frac{24 \cdot 2 - 3}{2 - 1} = 45$$

$$S = a + ak + ak^2 + \dots + b$$

Multiplying, both side by K

$$KS = aK + ak^2 + ak^3 + \dots + bK$$

$$KS - S = bK - a$$

$$S = \frac{bK - a}{K - 1}$$

Functions

$$\lfloor x \rfloor \downarrow \lceil x \rceil \uparrow$$

$$\lfloor 3/2 \rfloor = 1 \quad \lceil 3/2 \rceil = 2$$

Logarithms

$\log_K(x) = a$ exactly when $K^a = x$

A useful property of logarithms is that $\log_K(x)$ equals the number of time we have to divide x by K before we reach the number 1.

$$\log_2(32) = \underbrace{32, 16, 8, 4, 2, 1}_{5}$$

$$\log_u(x) = \frac{\log_K(x)}{\log_K(u)}$$

Contest

ICPC

Codeforces

Hacker Rank

Last Month

Interview Bit

Leetcode