Date _____

CODING NINJA (Competitive Programming)

```
#include<bits/stdc++.h>
using namespace std;
int main(){
    freeopen("file.txt","r",stdin);
    freeopen("file2.txt","w",stdout);
    cin>>
    cout<<
}
```

to take i/p and give o/p to a file

How to approach Problem

1) Paragraph → Think of soln
2) i/p format
3) o/p "
4) constraints
5) sample i/p, o/p
6) Explanation

→ Code → Submit.

{ 4×10^8 operations are done per second }

Scanned with CamScanner

Size of pointer - 8
(can be 4)
depends on comp.

Date

Constraints	time
10^8	$O(N)$
10^4	$O(N^2)$
10^8	$O(\log n)$
10^7	$O(n \log n)$
10^2	$O(n^3)$
30	$O(n^4)$
20	$O(2^n)$
11	$n!$

we can do
p1 < p2 } - p1, p2 - pointers
pointers me
p++ ~~vgera work~~
if p - integer then add 4 to hexadec. value

diff bw array & pointer
a - array - int a[5];
int i=2; we want a[i]
can be like
*(a+i), i[a]

int *p;
sizeof(p) → 4
sizeof(a) → 20

diff b/w array and pointer

cout << a;
cout << &a; } both are same

cout << p;
cout << &p; } diff.

p=p+1 → work

cant assign address of a variable. error ← a = a+3;

Scanned with CamScanner

Date _____

```
char b[] = "abc";

cout << b;
```

because cout for char pointer implemented ~~other~~ differently

will not print address but the value i.e. abc

```
char * c = &b[0];
cout << c;
```

print abc

```
char cl = 'k';
char *pc = &cl;
cout << cl;
cout << *pc;
```

prints k

doesn't print add but print *k 12WL

garbage till it will get \0

```
char str[] = "abcde";
char *pstr = "abcde";
```

pstr temp mem create then stored to perm. and return add.

pointing starts to add of temp array.

may or may not give an error.

Date _____

→ passed as a pointer only

```
int sum (int a[]) {
    cout << sizeof (a);  → 4
}
void main ()
{
    int a[10];
    cout << sizeof (a);  → 40
}
```

void pointer can store address of any variable.

```
int a = 65
char * pc = (char *)&a;
cout << * pc;
cout << *(pc+1);
```

prints a ek byte as a char and ek byte as deg.

prints next line

65	0	0	0

a ↑ → 4 byte

(big endian and little endian system)

↳ in dono ki ugh se farq pada

we assumed 0 0 0 65 hone chahie the

```
int i = 10, k = 50;
int & j = i;
```

↳ its just like ki ye i ka doosra naam bana deya (refrence variable

```
j = k;
```

↳ dono i, j ki value 50

Date _____

sirf ki kaam nhi karta.

int * p = new int;

address in stack

address in heap

→ Stack ki memory out of scope jaate hi khatam ho jaati hai
→ not in heap.

delete p; → to delete variable in heap whose address is with p.

delete [] p; → to delete array

```
int ** p = new int *[10];
p[0] = new int [20];
p[1] = new int [20];
:
p[10] = new int [20];
```

2D array

```
cout << p[2][3];
```

```
inline int max(int a, int b)
{
    return (a>b)? a : b;
}
```

to make inline fn if no. of lines goes greater than 3, compiler ignores inline, and do as normal fn

Scanned with CamScanner

int j = 5.

const int & k = j;

j++ ;

↳ work.

k wala path become constant. not value.

int const i = 10; → works

int const j3 = k3;

int & k3 = j3; → error.

j3 is read only,

k3 can read and write,

ambiguity uske paas access nhi hai toh kaise provide kr sktaha

error.

int const & k3 = j3;

works

int const * p = & i;

works → p = & j;

↳ it means ek pointer p is pointing to constant integer.

int * const p = & i;

↓

it means

p is a constant variable of type int *

now we cant do p = & j

but can do (*p)++.

we can change p but (*p)++ cant do.

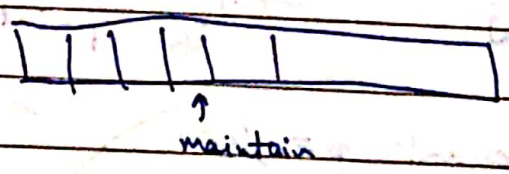
Scanned with CamScanner

Think recursion ≡ PMI
(principal of mathematical induction)
1) prove f(0) - true
2) assume f(k) is true)
3) using 2) prove f(k+1) is true

Space complexity includes only extra space.
function calls requires constant space, recursion me isliye ~~extra~~ space lagti hai.

Kadane's algorithm
↳ largest subarray sum.

max_sum
↓
will give max array so far.



current sum, if curr<0, curr=0
if curr>0 then add ith element.
if curr > max, update max.

Scanned with CamScanner

Date _____

problem → KCON (codechef)
→ Max. subarray product.

STL (standard template library)
→ vector (dynamic array)
→ String (collection of char)
→ pair
→ set
→ map
↳ unordered_map.
→ list (Doubly linked list)
→ Stack
→ Queue
→ priority queue

functions in STL
→ sort
→ binary-search. → (array must be in asc. order
→ lower_bound (if element found uska pointer, vrna usse just next bade ka pointer)
→ upper_bound (just bade element ka pointer dega)

→ __gcd (to find gcd)
→ pow
→ swap
→ reverse
→ max
→ min

problem → Hussain set (codechef)
→ Discrepancies in voters list (Codechef)
→ Permutation and palindrome (Codechef)

Scanned with CamScanner

Date

→ if complexity of solⁿ going high try to incorporate binary search thing in solⁿ.

Problem

→ Chef and Restraunt (codechef)

→ Variation (codechef)(code - ZCOPRAC)

merge sort approach can also help. converts $n^2 \rightarrow n\log(n)$ do if possible

recursion -

merge, quick

string has find function.

→ backtracking.

Problem

→ Sudoku (codechef)

Bit manipulation

→ left shift (<<)

→ right shift (>>)

→ and (&)

→ or (|)

→ xor (^)

→ not (~)

Scanned with CamScanner

Date _____

efficient way to multiply by 7

3x (left shift) − 1

{ right shift is division by two then floor } { 1 time left shift means multiply by 2 }

to check i^{th} bit is 1

$x \& (1 << i) != 0$.

to flip i^{th} bit

$x = x \wedge (1 << i)$

to check even or odd

$(n \& 1) == 0$.

to check if a no. is power of 2.

$N \& (N-1) == 0$

to clear all bits from i to LSB

$x = x \& ((1 << i) - 1)$

Greedy technique.

gen - min/max

- sorting yaad me chlta hai.

problem

→ Save energy (codechef)

→ Equalize (codeforces)

Scanned with CamScanner

Modulo operation.

1) $(a+b)\%c = (a\%c + b\%c)\%c$

↳ x_1 - k liye bhi valid.

if

$a\%m < 0$ then $a\%m = (a\%m + m)\%m$

Dynamic programming

Problems

→ Longest increasing subsequence
→ Staircase problem
→ Coin change problem.
→ Construct the array (hackerrank)
→ Maximum sum rectangle in 2D grid.
→ Sam and substring (hacker rank.
→ longest common subsequence
→ Knapsack.

Segment tree

↳ Jab hame ranges k beech me output nikalna ho.

for eg.
array me 2 se 5 k beech me ka sum krna ho
→ and vo baar baar modify ho raha ho.

Scanned with CamScanner

Date _____

lets take two op^n. -

query - to find sum from i^{th} to j^{th} element

update - to do $a[i]=k$.

normal soche

toh query k liye $O(n)$ time

update k liye $O(1)$ time

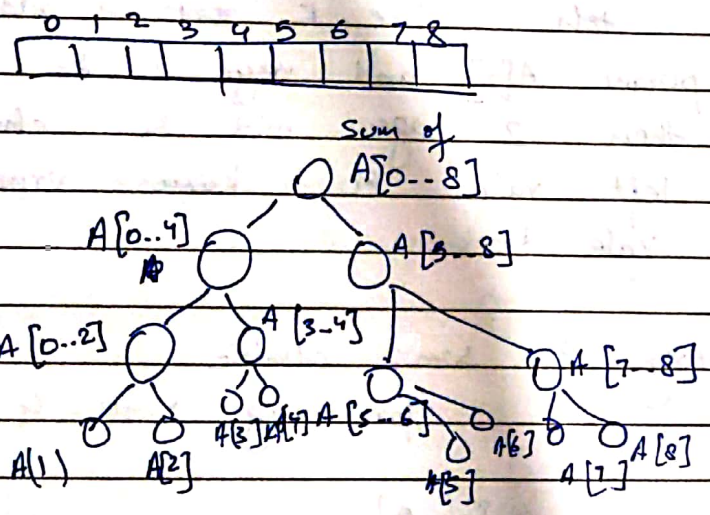
maintain sum array where sum[i] has sum of elements from 0 to i

query k liye $O(1)$ time

update k liye $O(n)$ time

Segment tree does

query, update both in $\log(n)$



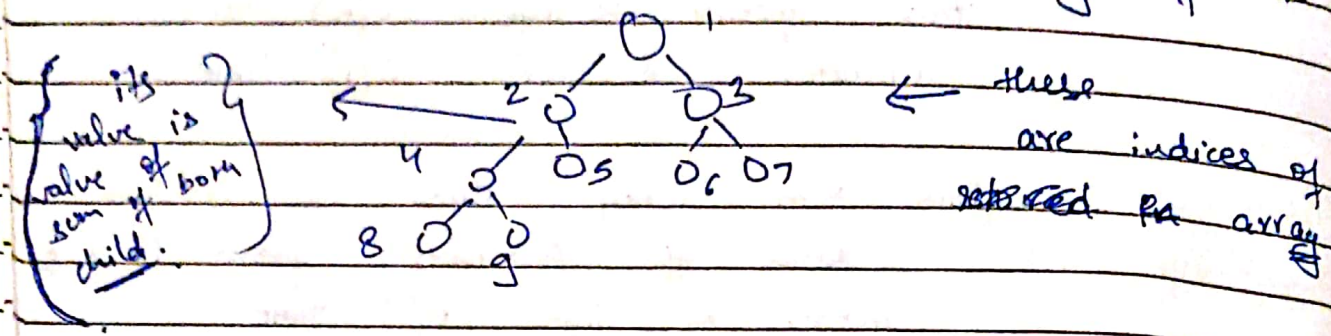
leaves are indices of array.

number of total nodes in segment tree = 4N.

0 - (→ so that kbhi dikkat nhi aaegi.)

O(N) - to construct segment tree

~~Solution~~ so what we do is construct array of 2N

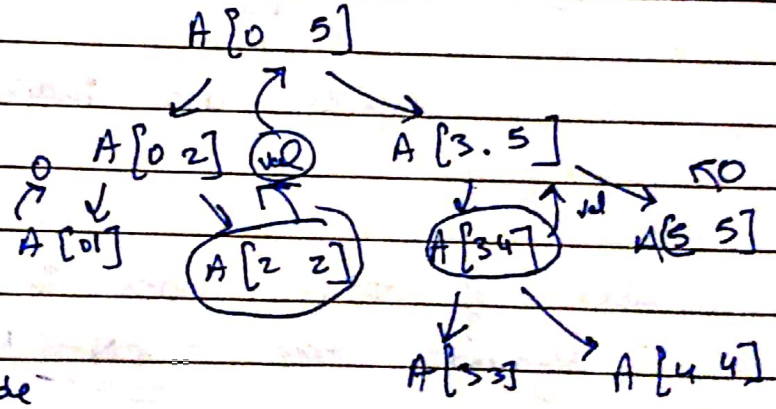


left child = 2 * i
right child = 2 * i - 1
↳ build this tree recursively

update for A[2]
Krna hai
toh tree k root p add kro
purane A[2] and naye 2 ka. then
then 2 agr mid se chota then
left vale ko karo varna right
vale ko.

query ka sum chahie
for eg. 2 to 4 ka chahie

agr 24 is partially inside toh dono child p Jao. dono taraf ek ans aaega jiska sum our ans. then if completely outside return 0, if completely inside return value



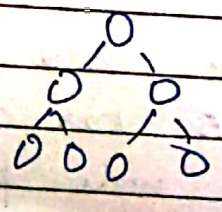
Scanned with CamScanner

Problem

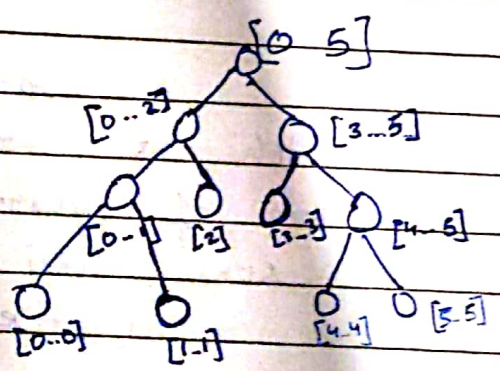
- We need to find max pair sum. (updates are also happening) for eg i se j k beech me max do element chahie
  → will make segment tree each node has max, second max.
  now think

- We need to find max sum subarray with query in update.
  - we need max sum, sum, Best prefix sum, Best sufix sum in each node.
    now think.

→ Lazy propogation
  ↳ update on a range
    eg. i to j tak do +2.



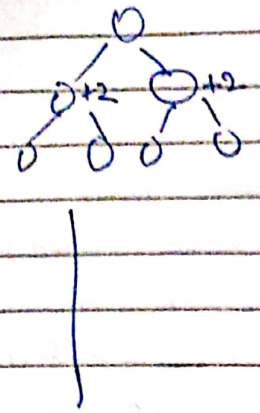
another tree initially all with 0.

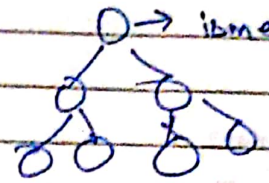


lets assume each has min val. in range. is tree me

and now if someone said [0..3] we do +2 kr do
[toh what we did]

Scanned with CamScanner





→ isme +2 kr dia, abhi child pr nhi gaya

and is lazy propogation ~~ke~~ child me update kr dia

then right tree me kuch update ya kuch kiya hai toh we will check lazy propogation tree agr ~~update~~ ~~kaona~~ dhundna hai kuch toh we update that node first lazy k us node ki val. then ~~uske~~ lazy k child ki values ~~ko~~ kr do vo.... carry on like this ...(check ~~ythe~~ first)

(virtually ~~help~~ save time)

Problem

→ Sum of squares (baaki segment tree vali problem)

update ne kaha har jgh +5 kar do toh we want har node pr sum of squares k ilava normal sum bhi hoz ~~uoz~~ ~~kate~~

$(a+k)^2 = a^2 + k^2 + 2ak$

noga toh for a, b, c, d (0 to 3)

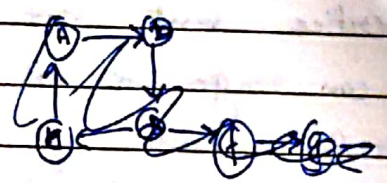
2 ~~(k~~ (a+b+c+d) ~~noga~~

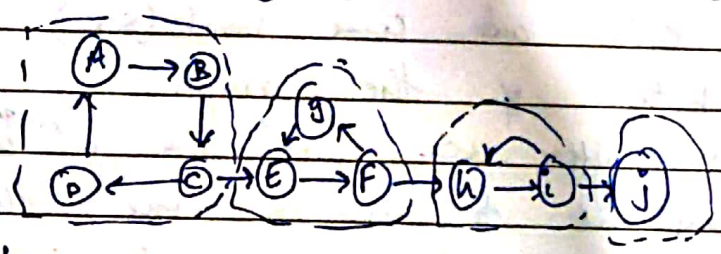
~~baaki~~

socho

Graphs

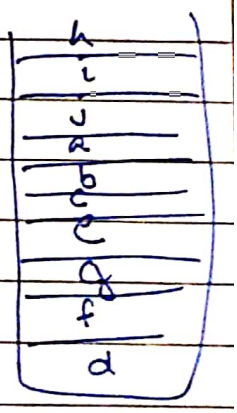
→ BFS
→ DFS
↳ (for disconnected graphs too)
→ Prim
→ Kruskal
→ Dijkstra's
→ Connected components.
→ Strongly connected components.
↳ for directed graph, ek component me ek vertice can go to any other vertice.
→ Kosaraju's algorithm.





these four are strongly connected comp.

to find them
use DFS, visit all the vertices and if for a vertice no more edge left or with unvisited vertice add it in the stack.
now transpose the above graph (all vertices reverse) refresh visited array, take vertice from stack, all unvisited vertices it can reach is a strongly connected component.



Scanned with CamScanner

Date

bottom of the graph.

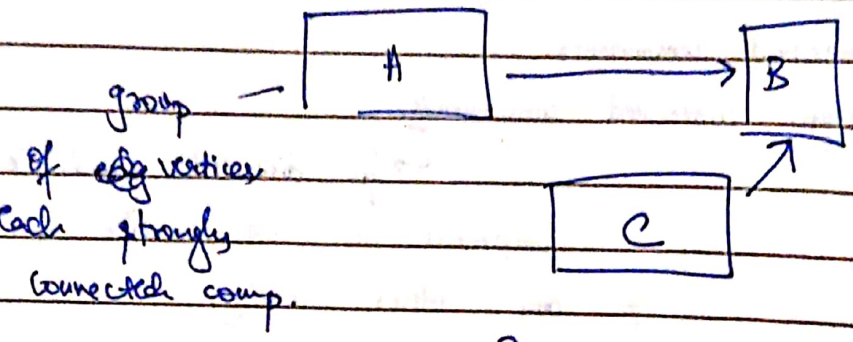
if

a → b pe ~~edge~~ path hai

then b → a bhi honi chahie then

a - bottom of the graph.

use SCC for this



— if like this

B - bottom of the graph.

A k kisi vertice se B ki koi vertice but not vice versa. toh A vale can go to B but not come. In B all can go ~~in the~~ to each each and come back (scc) toh it is bottom of the graph

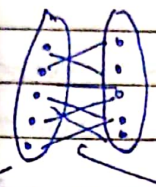
toh after scc just find a outgoing edge from a vertice of a component to another one, then it is not bottom of graph baaki kise se me se hoga

Date _____

→ Bipartite graph.

↓

if we can split a graph in two sets



each vertice in a set can't have edge ~~from~~ to a ~~ed~~ vertice from same set.

all the edges are from one set to other set

take a vertice, ek set me daaldo and saari connected in opp. set and queue me bhi, then ~~not~~ queue se nikalo, saari connected vertices opp. ~~to set~~ me, if same set me exists a vertice → then impossible to make bipartite.

else possible

Biparttite graph
↳ Can't have odd length cycles

Bit masking

Creating masks ~~to~~ via bit
{ eg. in TSP }

Date

for eg. [2, 5, 4, 3, 1] array we need to find no. of subsets with sum K.

00000
00001
00010
00011
⋮
11111

→ 0 to $2^5 - 1$ possible subset

each value is a mask telling if corresponding array element considered in the sum.

for mask = 0 to $1 << n - 1$
sum = 0
{ all bits which are set
↓
sum += corresponding element
if (sum == K)
{ count ++ }.

Bitmasking is gen. done with DP. To remove repetition done in recursion.

To set j^{th} bit,
mask | $(1 << j)$
to check j^{th} bit.
mask & $(1 << j)$

Scanned with CamScanner

Date _____

Tries

↳ for problemes related to XOR.

for eg. find max xor of any pair in a given array

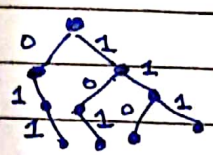
create trie –

eg. values are 6, 3, 7, 5

110 011 101 111

↳ max no. of bits reqd. = 3,

~~Trie → vohi value leaf tk reach karte jinke~~



~~Trie – values vohi end tk reach karti hai~~

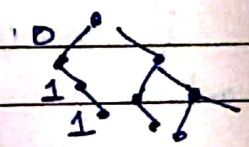
Trie – me unhi ki leaves bnti hain jo values are there.

Now we need to check, 5 ka xor max kiske saath hai

5 is 101

max xor hoga 010 k saath.

check if it is there in trie.



now third bit zero nhi hai toh us bit k liye 1 vali jgh proceed kar jaenge.

toh 011 i.e. 3 k saath max. XOR hoga, 5 ka

Scanned with CamScanner

Date _____

toh to find value of xor maximum pair, ~~take~~ initialize max_xor_value = 0

ek ek krke element lo, insert it in trie and find max xor possible for that value and update max_xor_value if its smaller otherwise keep going.

↳ complexity $N \times (\log(\max_ele))$

height of tree

→ Find max xor subarray

| 1 | 8 | 2 | 5 | 6 |

store 1 in trie

tha 1 xor 8, then 1 xor 8 xor 2,

1 xor 8 xor 2 xor 5

now 5 & k lies max xor

jisse aaega wo nikal lo

↳ Sorta

→ Find no. of subarrays whose xor is less than k

trie ~~s~~ make sim. tree as above

if k = 011011

and p = 101100 → current element from array.

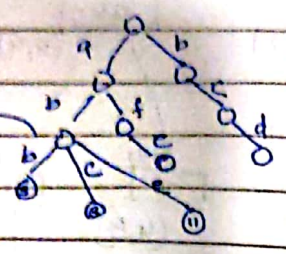
current bit of p, k

p=0, k=0	if p=1, k=1	if p=1 k=0	p=0 k=
move left	count all left element on right, move ~~right~~ left.	move right	count all left, move right

Date _____

→ N strings given each with a value
find string with max value
having some given prefix.

abc - 8 bcd - 8.
abb - 9
abfe - 10
abbe - 11



have k max value of all its leaf.
↳ do while constructing.

Fenwick Tree / Binary Index Tree

- Kaam is same as that of segment tree but easier to code and takes less space.

array → [2, 3, 4, 5, 6, 7, 11, 1]
indices → 1 2 3 4 5 6 7 8
↳ let them start from 1.

Same with fenwick tree ka array.
↳ indices from 1 to n.
size of array (n+1)

fenwick me
space complexity = O(n)
time complexity = O(log n), O(log n)
for update for query.

Scanned with CamScanner

Date

→ each level has one extra bit 1 from its parent.

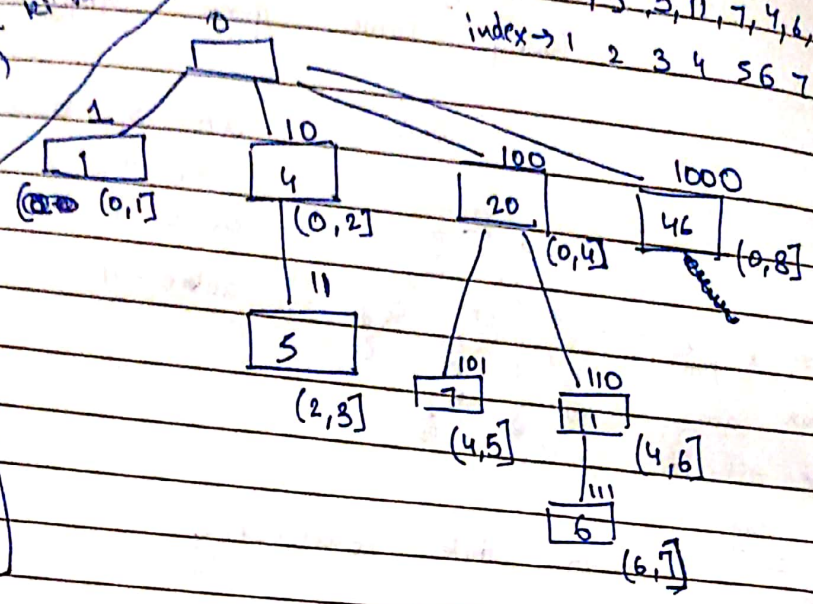
to understand it think has ek node pr ek binary no. stored hai and sum of values stored in that node is associated hai for eg. no is 1100 tak isme 1000 se (exclude) 1100 (include) tk. ki values ka sum aaya is stored.

Fenwick tree

array ↓

index → 1 2 3 4 5 6 7 8

1, 3, 5, 11, 7, 4, 6, 9



exclude ho raha hai kyunki index starts from 1

fenwick tree always give sum from [0, k]

so if ~~we~~ ~~100~~ k is eg. 110

so we will go to 6 (110) that will give sum of (100 to 110]

then we will add this to 4 (100) (by removing right most bit) then adding to 0 (000).

right most bit is removed by ↓

$x - (x \& (-x))$

(or how its make 100100 me pehli 1 bit ko same rkh kr baaki left bits ko invert) ← complement do

think that this

Date _____

for update → to add some value at specific index.

for eg. we want to update 5th index.

toh we keep adding 1 in right most bit and keep update all those places with new value

(till it goes greater than n)

for eg. 101 (5) p update krna hai,

toh we need to do it in 110, 1000. — think 101 kis kis range me include hoga

```
void update(int index, int val, int * BIT, int n){
    for(; index < n; index += index & (-index)){
        BIT[index] += value; }
}

void query(int index, int *BIT){
    int sum=0;
    for(; index=0; index -= index & (-index)){
        sum += BIT[index]; }.
```

Problem →

Coders ki list hai, each coder with two vals x, y. We need to c1 baaki kitne coders se bada hai. c1 > c2 if

c1.x > c2.x

c1.y > c2.y.

Scanned with CamScanner

Date

We sorted C ka array on basis of x.

then had a BIT. each index has ki usse choti values kitni hain

tree will be of size max val. of y.

eg. array after sorting is

x	y
1	2
2	9
3	5
4	8
5	1

so, phle dekha [0,2] tk kitni values hain fenwick me and then update.
then for [0,9] then update 9.
then [0,5], update 5 so on--

→ Problem - distinct query.
We are given an array

2	3	1	3	2	5	1

We need to check for index i to j no. of distinct values.

queries jo aa rahi hain unke end index k hisaab se sort kr do.
then phle save the result of the queries jinki values phle khtm to rahi hain then aage values ki coz

2 | 3 1 2 | 5 hai

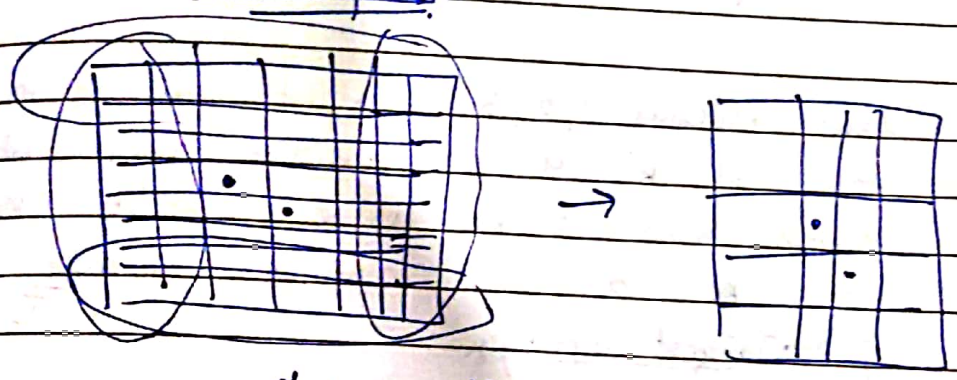
idhr ke distinct nikalne hain toh before reaching idhr we need to do ↗

Date _____

coz number 2 phle aa chuka tha and aage vo vapis se aa raha hai toh aage vale k lie unique +1 purane vale k lie -1 then aage vali query will work (tho pecche vali wrong ho but hum unka res nikaal chuke hain)

↳ o.soche.

Coordinate compression.



If • rakhe pos^n hi kaam kei hai then can reduce it to

OR.

values are like

15, 10, 20, 30, 2

3, 2, 4, 5, 1

] - Store this info via map

do stuff on this

we are not concerned for the values but for just ~~these~~ kei kon kitno se ~~bhaaa~~ bada hai

then replace

Date

Strings

Pattern matching.

KMP algorithm (knuth moris prath algorithm)

~~S: abc z abc g abz abcga~~

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23

S: abczabcgabz abcgabcgabcz

Pl: abcgabcz
1 2 3 4 5 6 7 8

start matching 1, 2, 3 then Pl ka 4 not matching toh Sl k 4 se continue then abcgab match now S ko 11 se. continue karenge but Sl ko 3 se continue karenge.

~~Largest epsilon~~

longest prefix suffix nikalenge
kaun ek ka.

i j

a b a b a d a d

0	0	1					

if str[i] ≠ str[j] and j = 0
then ar[i] = 0.

if str[i] = str[j]
ar[i] = ar[j]+1, i++, j++.

if str[i] ≠ str[j], j > 0
j--;
checking till

Date

Z Algorithm.

to find optimally this dekh lo

Z-function, Z(k) = length of longest substring starting at k which is also the prefix of the string.

a a c b y a a k a a c
0 1 0 0 0 2 1 0 3 1 0.

for eg. idhr se string aa hai and start of whole string is also aa that is common

String - yabck abdabc.

pattern - abc

make a string - string $ pattern.

i.e. abc $ y a b c k a b d a b c
0 0 0 0 0 3 0 0 0 2 0 0 3 0 0

this are two pos^n where Z-value = 3 = length of pattern, hence, this is answer.

Scanned with CamScanner

Number Theory

~~To find primes~~
Sieve of Eratosthenes
↳ To check primes b/w 1 to n.

Prime →

for each i ≤ n ~~i ≤ n~~ ~~sabke~~ ~~i ≤ n~~ ~~sabke~~ i ≤ n multiple ~~main~~ ~~i ≤ n~~ ~~sabke~~ lie prime false kr do

Euclid Algorithm.

$$GCD(a, b) = GCD(b, a \% b).$$

$$GCD(n, 0) = n.$$

Diophantine equations

↳ polynomial equations
↓
Integral ~~sol~~ sol^n exist

for linear diophantine eq^n,

$$ax + by = c$$

integral sol^n exists if GCD(a, b) divides c

x, y found by extended euclidean algorithm

Date _____

Extended euclidean algo.

if eq^n is

$$ax + by = GCD(a, b)$$
$$= GCD(b, a\%b)$$

$\exists\, x_1, y_1$ $\quad bx_1 + b\cdot(a\%b)\, y_1 = GCD(b, a\%b)$

$$ax_1 + \left(a - b\left[a/b\right]\right) y_1 = gcd(b, a\%b)$$
$$= gcd(a, b)$$

$$bx_1 + \left(a - b\left[a/b\right]\right) y_1 = ax + by.$$

comparing coeff

$$x = y_1$$
$$y = x_1 - \left[\frac{a}{b}\right] y_1$$

recursive

f^n bana ~~[illegible]~~.

Multiplicate modula inverse

$$(A \cdot B) \bmod m = 1$$

$$\Rightarrow (AB - 1) \bmod m = 0$$
$$\Rightarrow AB - 1 = mq.$$
$$AB - mq = 1$$
$$AB + mQ = 1$$

if B - integral.

$\Rightarrow$ GCD (A, m) divides 1

$\Rightarrow$ $A, m \Rightarrow$ coprime.

now $\quad AB + mQ = gcd(A, B)$

↳ now extended euclidian algo find B.

Scanned with CamScanner

Advanced gcd

gcd(a, b)

$0 \le a < 40000$

$0 \le b < 2^{450}$

Store it in string.

Now need to calc. GCD.

$\therefore (a \times b) \% m = ((a \% m) \times (b \% m)) \% m$

24 56 78 921 → for eg. ye no. hai we need %40 what we do.

$(0 \times 10 + 2) \% 40 = 2$

$(2 \times 10 + 4) \% 40 = 24$

$(24 \times 10 + 5) \% 40 = 5$

$(5 \times 10 + 6) \% 40 = 16$

$(16 \times 10 + 1) \% 40 = \ldots$ so on.

Find all divisors of n.

$n = p1^a \times p2^b \times p3^b \times \ldots$

p1, p2, p3 – all primes below n.

$a = \left[\frac{n}{p1}\right] + \left[\frac{n}{p1^2}\right] + \left[\frac{n}{p2^3}\right] + \ldots$

[.] = GIF.

divisors $= (a+1)(b+1)(c+1)\ldots$

{for n! we need to find primes till n.}

Scanned with CamScanner

Game theory

→ Without playing need to tell who will win.

Combinatorial game theory (game which will end after some time)

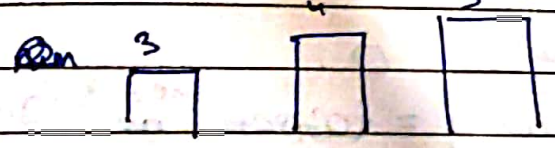
Impartial game
same moves available for each person.
eg. game of nim

Partisan game
dif moves for each player. eg. chess.
[black can move only black, white can move only white]

Game of nim.

There a n piles and each pile having k stones. a player loses if no more stone to remove.

each player can remove any no. of stones from any pile.



Pl starts first

nim sum = 3 ^ 4 ^ 5

if it = 0 → Pl loses

≠ 0 → Pl win.

if nim sum ≠ 0 then.

ek move pkka hoga making nim sum = 0. and use vo bhj k saamne vale ko bhj denge. the saamne vale k paas no choice except sending nim sum ≠ 0 and this goes on

Scanned with CamScanner

Mex (Minimum exclusant set)

↳ smallest non-negative value not present in set.

Mex {1, 2, 3} = 0

Mex {0, 1, 2, 5} = 3.

Grundy numbers.

Grundy(n) = Mex {Grundy of all the possible posⁿ from nᵗʰ posⁿ of game}.

Grundy(0) = 0.

↳ means end of game

for eg n stones we can take 1, 2, 3 stones out of it at a time.

Grundy(0) = 0.

Grundy(1) = mex {Grundy(0)} = 1

Grundy(2) = mex {Grundy(0), Grundy(1)} = 2.

Grundy(3) = mex {Gr(0), Gr(1), Gr(2)} = 3

Grundy(4) = mex {Gr(1), Gr(2), Gr(3)} = 0

Sprague grundy theorom.

for eg. ab game of nim we can remove only 1, 2, 3 stones from piles p1, p2, p3 ..

Scanned with CamScanner

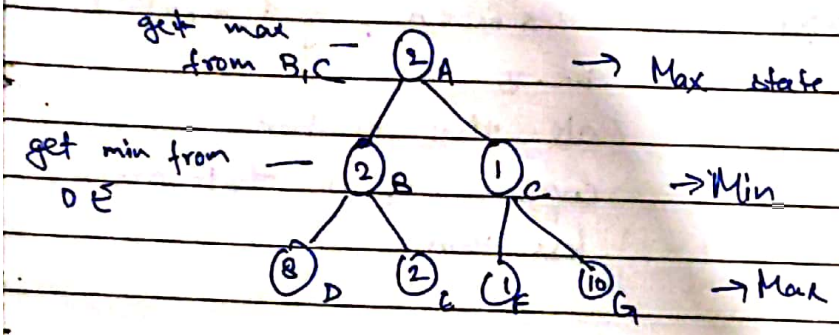
Date _____

nim_sum = Grundy (p1) ^ Grundy (p2) ^ Grundy (p3)

nim_sum $\neq 0$ → p1 win
$= 0$ → p2 win.

Min - Max Algorithm.

Maximizer - max score we can get from curr state
Minimizer - min score we can get from current state.



(We toggle max min)

A, B, C states
i.e. B, C states possible from A.
D,E from B ; F,G, from C.

leaf nodes will have a score with them.

Evaluation function.
↳ Score found on leaf.

for eg. on Tic Tac Toe
if A win f^n returns +10,
if B win f^n returns -10
if draw f^n returns 0.

Scanned with CamScanner

Date

for Othello, evaluate f^n be,

count[1] - count[0].

total # discs with 1 (black)
- total discs 0 (white)

Number theory — π

Euler totient function.

$\phi(n) \rightarrow$ no. of 'm' such that
$n \geq 1$ $1 \leq m < n$, $\gcd(m,n) = 1$
i.e.,
all the number coprime of n that are less than.

$\phi(a,b) = \phi(a) \cdot \phi(b)$ if $GCD(a,b) = 1$

$n = p_1^a \cdot p_2^b \cdot p_3^c \cdots p_k^\alpha$
— primes.

$\Rightarrow \phi(n) = \phi(p_1^a)\, \phi(p_2^b)\, \phi(p_3^c) \cdots$

now $\phi(p^a)$?

$1, 2, 3 \; p + 2p + 3p \cdots p^a$.

only these will not be coprime, baaki sab coprime hange

$\phi(p^a) = p^a -$ (all multiples of p)

Scanned with CamScanner

no. of elements / multiples,

$p + (n-1)p = p^a$.

$(n)\,p = p^a$

$\boxed{n = p^{a-1}}$.

$\phi(p^a) = p^a - p^{a-1}$

$\phi(p^a) = p^a\left(1 - \frac{1}{p}\right)$

$\Rightarrow \quad \phi(n) = \phi(p_1^a)\phi(p_2^b) \cdots \phi(p_k^x)$

$= p_1^a\left(1 - \frac{1}{p_1}\right) p_2^b\left(1 - \frac{1}{p_2}\right) p_k^x\left(1 - \frac{1}{p_k}\right)$

$$\boxed{\phi(n) = n\left(1 - \frac{1}{p_1}\right)\left(1 - \frac{1}{p_2}\right)\cdots\left(1 - \frac{1}{p_k}\right)}$$

To calc. $\phi(n)$ can use sieve of eratothes wali approach.

and

$\phi(p) = p - 1$

↳ if $p \to$ prime

1) Initializ arr with $arr[i] = i$

2) $i = 2$ to N.

if $(arr[i] = i)$

① $arr[i] = i - 1$

② All multiple → multiply by $\left(1 - \frac{1}{i}\right)$

Scanned with CamScanner

LCM

$$LCM(a,b) = \frac{a \times b}{\gcd(a,b)}$$

To find

$$S = LCM(1,n) + LCM(2,n) + \dots LCM(n,n)$$

$$S - n = LCM(1,n) + L(2,n) + \dots LCM(n-1,n)$$

$$S - n = LCM(n-1,n) + LCM(n-2,n) + \dots LCM(1,n)$$

$$LCM(a,n) = \frac{a.n}{\gcd(a,n)}$$

$$LCM(n-a,n) = \frac{(n-a)n}{\gcd(n-a,n)} = \frac{(n-a)n}{\gcd(a,n)}$$

odd

$$\boxed{2S - 2n = \sum_{i=1}^{n-1} \frac{n^2}{\gcd(i,n)}}$$

now for some $\alpha_1, \alpha_2, \alpha_3 \dots$ $\gcd(\alpha,n) = d_1$

for some $\beta_1, \beta_2, \beta_3$ $\gcd(\beta_1,n) = d_2$

& $\alpha_1\ \alpha_2\ \alpha_3\ \beta_1\ \beta_2\ \beta_3$

↳ all are $< n$.

if we get this our time will get reduced.

Date _____

$\gcd(i,n) = d$

$\gcd\left(\frac{i}{d}, \frac{n}{d}\right) = 1$

↳ to get all the i

$\Rightarrow$ $\phi\left(\frac{n}{d}\right)$ = no. of numbers for which $GCD(i,n) = d$.

$$\boxed{2S - 2n = \sum_{d \mid n,\ d \neq n} \frac{n^2}{d} \times \phi\left(\frac{n}{d}\right)}$$

replace d by n/d.

$$\boxed{2S - 2n = \sum_{d \mid n,\ d \neq 1} n \times \phi(d) \times d}$$

$\because \phi(1) = 1$

$$(2S - 2n) = n\left(\sum_{d} \phi(d) \cdot d - 1\right)$$

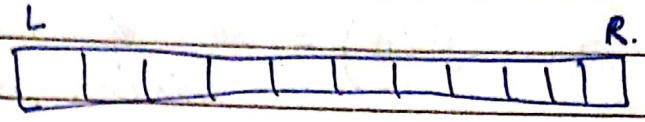
now $d = 1$ inc.

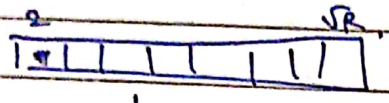
$$\Rightarrow \boxed{S = \frac{n}{2}\left(\sum \phi(d) \times d + 1\right)}$$

Scanned with CamScanner

Segmented sieve.

we need to find prime no's b/w l, r.





find primes below R

multiples within l to R
↳ false

Number Theory - III

if we want ans % c.

```
power (x,n)
        if (n % 2)=0.
                val=pow(x, n/2) % c ;
                return (val x val) % c ;
        if (n % 2) == 1
                val = pow (x, n-1) % c
                return (x x val) % c ;
```

Recursively →

if we wanted
ans = a^{45} % c

$(45)_{10} = (101101)_2$
5 4 3 2 1 0

Iteratively →

$(a^{2^5} \times a^{2^3} \times a^{2^2} \times a^{2^0}) \% c$

$[(a^{2^5} \% c) \times (a^{2^3} \% c) \times (a^{2^2} \% c) \times (a^{2^0} \% c)]$

↳ (now $b/2^b/2$ kyka multiply karte rahoo)

Scanned with CamScanner

Matrix exponentiation

↳ helps in reducing time to find recurence relation.

for eg.

$$f(n) = f(n-1) + f(n-2)$$

we need two base case/values here, hence M is of size 2x2

agar $f(n) = f(n-1) + f(n-2)$ hota toh 3x3 hoti.

$$M_{2\times2} \times \begin{bmatrix} f(n) \\ f(n-1) \end{bmatrix} = \begin{bmatrix} f(n+1) \\ f(n) \end{bmatrix}$$

any matrix jisse ek matrix se multiply karien toh next state mil jaye.

prev state

next state.

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} f(n) \\ f(n-1) \end{bmatrix} = \begin{bmatrix} f(n+1) \\ f(n) \end{bmatrix}$$

$$a\, f(n) + b\, f(n-1) = f(n+1)$$
$$c\, f(n) + d\, f(n-1) = f(n)$$

$\because$ $f(n) = f(n-1) + f(n-2)$

given rec. relⁿ.

$\Rightarrow a = 1,\ b = 1$
$c = 1,\ d = 0.$

now,

$$M \times M \times \underbrace{\begin{bmatrix} f(n) \\ f(n-1) \end{bmatrix}}_{k^{th}\text{ state}} = M \times \underbrace{\begin{bmatrix} f(n+1) \\ f(n) \end{bmatrix}}_{k+1^{th}\text{ state}} = \underbrace{\begin{bmatrix} f(n+2) \\ f(n+1) \end{bmatrix}}_{(k+2)^{th}\text{ state}}$$

Scanned with CamScanner

Date

$$M^k \times \begin{bmatrix} f(n) \\ f(n-1) \end{bmatrix} = \begin{bmatrix} f(n+k) \\ f(n+k-1) \end{bmatrix}$$

$$M^{n-1} \times \begin{bmatrix} f(1) \\ f(0) \end{bmatrix} = \begin{bmatrix} f(n) \\ f(n-1) \end{bmatrix}$$

find this by power peeche jaise nikali in logn.

Now we can find $f(n)$ in $\log n$.

agr $f(n) = f(n-1) + f(n-3)$

we need 3 base case values.

$\Rightarrow 3 \times 3$ matrix.

$$\underbrace{\begin{bmatrix} 1 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}}_{M} \begin{bmatrix} f(n) \\ f(n-1) \\ f(n-2) \end{bmatrix} = \begin{bmatrix} f(n+1) \\ f(n) \\ f(n-1) \end{bmatrix}$$

Soch lo. will do.

Scanned with CamScanner

Date _____

lets $f(n) = a \times f(n-1) + c\, f(n-3) + d\, f(n-4) + e$

$$\begin{bmatrix} a & 0 & c & d & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} f(n) \\ f(n-1) \\ f(n-2) \\ f(n-3) \\ e \end{bmatrix} = \begin{bmatrix} f(n+1) \\ f(n) \\ f(n-1) \\ f(n-2) \\ e \end{bmatrix}$$

if $f(n) \to f(n-1)$ for odd
$\to f(n-2)$ for even] make two seperate matrix.

Problem

need to find
sum of fibonacci's no. from m to n.

$S_n = S_{n-1} + F_n$.

$\boxed{S_{n-2} = F_n - 1}$ $\to$ $F_n = F_{n-1} + F_{n-2}$
$F_{n-1} = F_{n-3} + F_{n-4}$
$\vdots$
$F_1 = 1$

$\Rightarrow F_n = \underbrace{F_{n-2} + F_{n-3} + F_{n-4} + \cdots + 1}_{S_{n-2}}$

Fermat's little theorem.

p = prime no.

$\Rightarrow \boxed{(a^p) \% p = a}$

Scanned with CamScanner

Date

$\Rightarrow a^p \equiv a \pmod p$

$\Rightarrow a^{p-1} \equiv 1 \pmod p$

$(a^{p-1}) \% p = 1.$

↳ used to find multiplicate ~~inverse~~ inverse

We need to find $(a^{-1}) \bmod p$.

$(a^{p-1}) \% p = 1$

$a^{-1}(a^{p-1}) \% p = a^{-1}$

$(a^{-1}(a^{p-1}) \% p) \% p = (a^{-1}) \% p.$

$(a^{-1} \cdot a^{p-1}) \% p = (a^{-1}) \% p.$

$(a^{p-2}) \% p = (a^{-1}) \% p.$

$\Rightarrow$ multiplicative inverse of $a \% p = a^{p-2} \% p.$

$$\boxed{(a^{-1}) \bmod p = (a^{p-2}) \bmod p}$$ p-prime

WILSON'S THEOREM

$p \to$ prime

$$\boxed{(p-1)! \bmod p = -1 \text{ or } p-1}$$

$(p-1)$ factorial.

Date _____

if we need to find $n!$

ans = $(n!) \% p$. p - prime

if $n \geq p$.

ans = 0.

if $n < p$.

$\because \{[1 \times 2 \times 3 \cdots (n-1) \times n] \times [(n+1)(n+2) \times \cdots (p-1)]\} \% p$
$= -1 = p-1$

$= (n! \% p) \times [(n+1)(n+2) \cdots (p-1) \% p] = -1$

$\Rightarrow (n!) \% p = -1 \times (n+1)^{-1}_{\text{mod } p} \times (n+2)^{-1}_{\text{mod } p} \times \cdots (p-1)^{-1}_{\text{mod } p}$

$\because$ -ve

end we ans = ans + p.

$\because a^{-1} \text{ mod } p = a^{p-2} \text{ mod } p$.

$$\Rightarrow \boxed{(n!) \% p = p + [((n+1)^{p-2} \% p) \times [(n+2)^{p-2} \% p] \times \cdots [(p-1)^{p-2} \% p]] \% p}$$

Problem. -

$f(n) = f(n-1) + f(n-2) + f(n-1) \times f(n-2)$

$f(0), f(1) \rightarrow$ given.

↳ whenever recurrence relⁿ are given try to reduce them into fibonacci or somewhat simmilar thing

Date

$$f(n) = (1+f(n-1))(1+f(n-2)) - 1$$

$$\Rightarrow \quad 1+f(n) = [1+f(n-1)]\,[1+f(n-2)]$$

let $G(n) = 1+f(n)$

$$\Rightarrow \quad G(n) = G(n-1) \times G(n-2)$$

$G(0) = a$

$G(1) = b.$

$G(2) = a^1 b^1$

$G(3) = a^1 b^2$

$G(4) = a^3 b^5$

$\vdots$

$$\boxed{G(n) = a^{fib(n-1)}\, b^{f(n)}}$$

if we need to give $\%p$

$\because \; a^{p-1} \bmod p = 1.$

p is prime

$$G(n) = \left[a^{f(n-1)} \cdot b^{f(n)}\right] \%p$$

$$= \left[a^{f(n-1)} \%p\right] \left[a^{f(n)} \%p\right]$$

$$\boxed{f(n-1) = k \times (p-1) + f(n-1) \%(p-1)}$$

one k will always be there.

$$a^{f(n-1)} \%p = a^{k(p-1) + \underbrace{(f(n-1) \%\, p-1)}_{\to x}} \%p.$$

$$= \left(a^{k(p-1)+x}\right) \%p.$$

Scanned with CamScanner

Date _____

$$a^{f(n-1)} \% p = \left(a^{k(m-1)} \% m\right) \times \left(a^{x} \% m\right)$$

$$\downarrow$$

$$a^{f(n-1)} \% p = 1 \times a^{x} \% m$$

$$\boxed{a^{f(n-1)} \% p = \left[a^{f(n-1) \% (p-1)}\right] \% p}$$

use this

for this

Computational geometry.

Vectors

→ dot product

→ cross product

Distance of pt from line

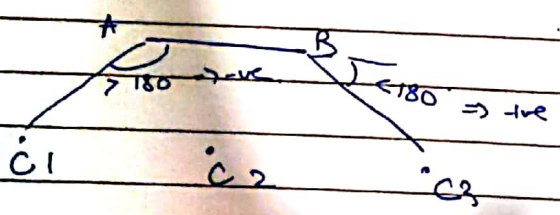
$$d = \frac{(\vec{b} - \vec{a}) \times (\vec{c} - \vec{a})}{|\vec{b} - \vec{a})|}$$

c

Distance of pt from line segment.

3 case.

angle > 180



if $AB \cdot AC = -ve$

dist = AC1

if $AB \cdot BC = +ve$

dist = AC2

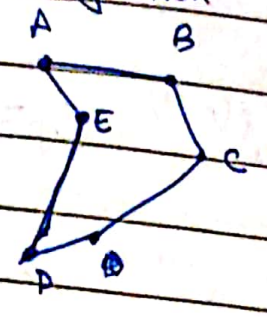
else → $dis = \frac{(\vec{b} - \vec{a}) \times (\vec{c} - \vec{a})}{|\vec{b} - \vec{a}|}$

Scanned with CamScanner

Area of a polygon. (triangulation method)

$$Ar = |Ar(ABC) + Ar(ACD) + Ar(ADE)|$$

apne aap +ve -ve hoga.



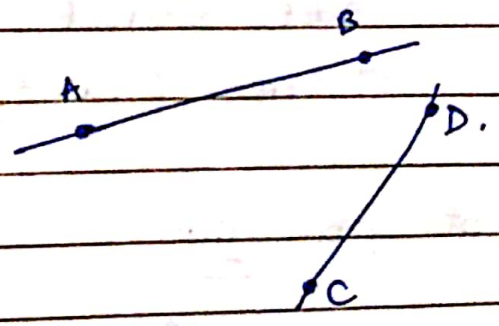
$$Ar(ABC) = \frac{\vec{AB} \times \vec{AC}}{2}$$

cross product of (x_1, y_1), (x_2, y_2)

$= x_1 y_2 - y_1 x_2$.

determinant

Intersection of two lines.



normal maths se nikal lo.

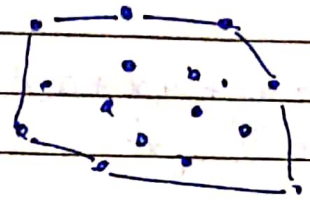
if line segment hoti

then pt of intersection must be b/w A and B and C and D

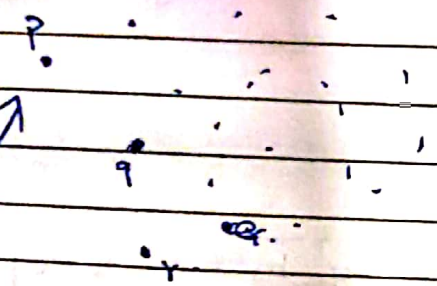
Date _____

Jarvis Algorithm.

Convex hull. → we need to find boundary of points.



→ to find this boundary.



sbse phle left most pt. lo. and push in a vector

now take a point q. (~~random~~ iteratively.)

now for all r.

↓

if

$\vec{pq} \times \vec{qr} > 0$.

aisi cond^n ke hoga

$q = r$.

now iterate

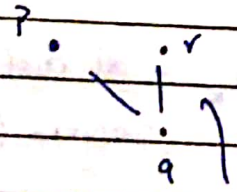
push q in vector.

$p = q$.

do this till

$q = p$

means polygon is now closed.



for this $pq \times qr < 0$.

Scanned with CamScanner

Date

Fast fourier transform. (FFT)

→ application :- Multiply two polynomials.

$A(x) = a_0 + a_1 x + a_2 x^2 + \dots a_{n-1} x^{n-1}$

$A(x) = \sum_{i=0}^{n-1} a_i x^i$

↳ Degree $n-1$

$C(x) = A(x) . B(x)$.

↓

$$\boxed{c_i = a_i b_0 + a_{i-1} b_1 + a_{i-2} b_2 + \dots a_0 b_i}$$

↳ n^2 approach.

want to reach by $n \log n$

Point value form

need n point of eqn of degree n.

$A(x) = \{ (x_0, y_0), (x_1, y_1), (x_2, y_2) \dots (x_{n-1}, y_{n-1}) \}$

$$\boxed{y_k = A(x_k)}$$

x_k - all distinct integers (random)

for quadratic eqn

$y_0 = a_0 + a_1 x_1 + a_2 x_1^2$ | $A(x_1)$

$y_1 = a_0 + a_1 x_2 + a_2 x_2^2$ | $A(x_2)$

$y_2 = a_0 + a x_3 + a_3 x_3^2$ | $A(x_3)$

from these we can find a_0, a_1, a_2.

Date

$$\begin{pmatrix} x_0 & x_0^2 & \cdots & x_0^{n-1} \\ x_1 & x_1 & \cdots & x_1^{n-1} \\ \vdots & & & \\ x_{n-1} & x_{n-1}^2 & \cdots & x_{n-1}^{n-1} \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_{n-1} \end{pmatrix} = \begin{pmatrix} y_0 \\ y_1 \\ \vdots \\ y_{n-1} \end{pmatrix}$$

$$X A = Y$$

$A = X^{-1} Y$ → will also find it in $O(n \log n)$

1) $A(x)$, $B(x)$ - need $(2n-1)$ pairs for $C(x)$

$A(x) = [(x_0, \alpha_0), (x_1, \alpha_1) \cdots (x_{2n-1}, \alpha_{2n-1})]$

$B(x) = [(x_0, \beta_0), (x_1, \beta_1) \cdots (x_{2n-1}, \beta_{2n-1})]$

↳ will calc. this in $O(n \log n)$.

2) $C(x) = [(x_0, \alpha_0\beta_0), (x_1, \alpha_1\beta_1) \cdots (x_{2n-1}, \alpha_{2n-1}\beta_{2n-1})]$

↳ will calc. in $O(n)$.

this is known as convolution.

#

$$\omega_n^k \to e^{i2\pi k/n}$$

1) for $n \geq 0$, $k \geq 0$, $d \geq 0$

$$\omega_{dn}^{dk} = \omega_n^k.$$

2) for $n > 0$, n - even

$$\omega_n^{n/2} = \omega_2^1 = -1.$$

Scanned with CamScanner

3) $n > 0$ n - even, square of n complex n^{th} root of unity is $(n/2)^{th}$ root of unity

$$\omega_n^k = e^{i2\pi k/n}$$

$$(\omega_n^k)^2 = \omega_{n/2}^k$$

each $n/2^{th}$ root of unity repeated exactly twice.

4) $$\left[\omega_n^{(k+n/2)}\right]^2 = \omega_{n/2}^k = (\omega_n^k)^2$$

5) $n \geq 0$ $k \geq 0$,

$$\omega_n^{k+n/2} = -\omega_n^k.$$

Converting coeff value form $\rightarrow$ point value form.

$$y_k = A(x_k)$$

$$A(x) = \sum_{j=0}^{n-1} a_j x^j$$

need n distict points,

will take n complex root of unity.

Scanned with CamScanner

$$A(\omega_n^k) = A(\omega_n^k) = \sum_{j=0}^{n-1} a_j \omega_n^{kj}$$

$A(x)$ distribute

$A_{even}(x)$
$= a_0 + a_2 x + a_4 x^2 + \dots a_{n-2} x^{n-2}$

$A_{odd}(x)$
$a_1 + a_3 x + a_5 x^2 + \dots + a_{n-1} x^{n-1}$

$$A(x) = A_{even}(x^2) + x A_{odd}(x^2)$$

need n dist. point

(galt hai in video too)

$A_{even}(x^2)$ and $A_{odd}(x^2)$

calc by $n/2$ dist. point.

$$\begin{bmatrix} (\omega_n^0)^2 & (\omega_n^1)^2 & \dots \\ 1 & & \\ (\omega_n^{n/2-1})^2 & & \end{bmatrix}$$

$n/2$ complex $n/2$ roots of unity

evaluated on these

$\because \omega_n^{2k} = \omega_{n/2}^k$.

$$\left(\omega_{n/2}^0, \omega_{n/2}^1, \omega_{n/2}^{n/2-1}\right)$$

want point val. on these for $A(x)$

will get point value by this of odd and even

$A(x)$ → $A_{even}(x)$

$A(x)$ → $A_{odd}(x)$

need n dist. point

need $n/2$ dist. point by each. (recursion)

Date

$$A(x) = A_{even}(x^2) + x \cdot A_{odd}(x^2)$$

$$A(\omega_n^k) = A_{even}(\omega_n^k)^2 + \omega_n^k \times A_{odd}(\omega_n^k)^2$$

$$k = 0 \ldots n-1$$

$$A(\omega_n^k) = A_{even}(\omega_{n/2}^k) + \omega_n^k A_{odd}(\omega_{n/2}^k)$$

$$k = 0 \ldots n/2 - 1$$

for $A(\omega_n^k) = A_{even}(\omega_{n/2}^k) + \omega_n^k (A_{odd}(\omega_{n/2}^k))$

and $A_{\omega_n^{k+n/2}} = A_{even}(\omega_n^{k+n/2})^2 + \omega_n^{k+n/2} (A_{odd}(\omega_n^{k+n/2})^2)$

can solve these ||ly

$$A_{\omega_n^{(k+n/2)^2}} \rightarrow A_{\omega_{n/2}^k} = A_{\omega_n^{2k}}.$$

$$\omega_n^{k+n/2} = -\omega_n^k$$

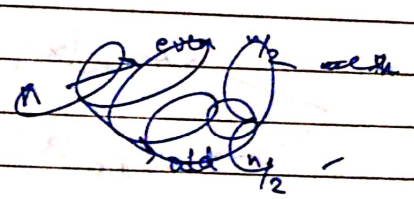
as this is

$$A_{\omega_n^{k+n/2}} = A_{even}(\omega_{n/2}^k) - \omega_n^k (A_{\omega_{n/2}^k})$$

toh $(n \log n)$

↳ me ho jaega via recursion

as



to merge sol^n will take n.

n k. liye n/2 things to be evaluated and two sol^n will be given. ~~be~~ ~~[illegible]~~ by + and − recursion for n/2.

$\log n$

Scanned with CamScanner

Complex is a inbuilt data type

Date _____

Scanned with CamScanner