**PERSONAL DIARY FOR C++**

fmax(),fmin()

(fdim() **for only positive difference between 2 values**)

strcpy() (Copies character string from source to destination)

strncpy(copies n cherecter)

strncpy(dest,src,10) (src string copies to dest)

strcspn() ( searching string 1st and 2nd there any value here or not)if here which position?? its return positions)

*if*(s.find(ss) *!=* std::string::npos)

(for check one string another sub striing or not)

isalpha() (its return zero when expected cher is number)

char d=toupper(‘a’) (convert to upper cher)

**\_\_gcd(a,b) (algorithm and iostream header dewa laghbe)**

**lcm(a,b) (its only work in 20)**

**Header file:** #include<vector>

**Declaration:**vector variable\_name

### Functions

* **push\_back():** Pushes an element at the back of array in a serial manner.
* **pop\_back():** It pops the last element from a vector.
* **front():**Returns the first element of a vector.
* **back():**Returns the last element of a vector.
* **size():** Returns the size of the vector.
* **clear():** It deletes all elements in the vector.
* **empty()**: It returns a Boolean value after checking whether the vector is empty or not.

int n = sizeof(arr)/ sizeof(arr[0]);( array size)

**Header file:** #include<set>

**Declaration:** stack variable\_name

### Functions

* **insert():** This function is used to insert a new element in the Set.
* **begin():**This function returns an iterator to the first element in the set.
* **end():** It returns an iterator to the theoretical element that follows the last element in the set.
* **size():** Returns the total size of the set.
* **find():**It returns an iterator to thesearched element if present. If not, it gives an iterator to the end.
* **count():**Returns the count of occurrences in a set. 1 if present, else 0.

**Map**

**Count[string]++;**

**Setprecision(0) >>>>> its control number after decimal..(3.11416)>>>(3)**

**Decimal to binery**

1. #include<bits/stdc++.h>
2. using namespace std;
3. int main() {
4. unsigned long n = 24;
5. cout << "Binary Number: "<<bitset<8>(n).to\_string();
6. }

bool sortbyCond(const pair<int, int>& a, const pair<int, int>& b)

{

    if (a.first != b.first)

        return (a.first < b.first);

    else

       return (a.second > b.second);

}

sort(v.begin(), v.end(), comp);

vector <pair<int, int> > v;

**Its using if I want print(1,2 || 1,3 )->>>>>>>(1,3 || 1,2)**

**Decending sorting—pair**

**vector<pair<int,int>>v;**

**sort(v.begin(),v.end(),[&](pair<int,int>x , pair<int,int>y) {**

**if(x.first>y.first) {**

**return true;**

**}**

**return false;**

**//return(x.first>y.first)?true:false;**

**//return (x.first>y.first);**

**});**

**For sorting**

**Sort(v.begin(),v.end());**

**Sort(arr,arr+size)**

**Set(store unique 1 element and it’s also store sorting order)**

**Multiset(store unique more then 1 element and its store sorting order)**

long long GCD(long long n1, long long n2) {

    if (n2 != 0)

       return GCD(n2, n1 % n2);

    else

       return n1;

}

**Power mod**

int big\_mod(int a,int b,int m){

    if(b==0) return 1%m;

    int x=big\_mod(a,b/2,m);

    x=(x\*x)%m;

    if(b%2==1) x=(x\*a)%m;

    return x;

}

Sieve: (prime number store)

void sieve(vector<int>& primes) {

    bool isPrime[MAX];

    memset(isPrime, true, sizeof(isPrime));

    for (int i = 2; i \* i < MAX; i ++) {

        if (isPrime[i]) {

            for (int j = i \* i; j < MAX; j += i) {

                isPrime[j] = false;

            }

        }

    }

    for (int i = 2; i < MAX; i ++ ) {

        if (isPrime[i]) {

        primes.push\_back(i);

        }

    }

}

Prime devisor count:

int prime(int n) {

 int k=0,count=0;

    while(n%2==0){

        n/=2;

    if(k==0){

        count++;

        k++;

    }

    }

    k=0;

    for(int i=3;i\*i<=n;i+=2) {

        while(n%i==0){

             n/=i;

        if(k==0){

            count++;

         k++;

        }

        }

    k=0;

    }

    if(n>1){

       count++;

    }

    return count;

}

Prime\*prime==or 3 ta devisor.

2 ta value er multiply control big value:

long long dong(long long a,long long b){

   long long ans=0;

   while(b){

    if(b&1){

        ans=((ans)+(a))%mod;

    }

    a=((a)+(a))%mod;

    b>>=1;

   }

      return ans;

}

BINERY EXPO find nCr :

**const** int mod = (int) 1000000007;

**const** int N=1e6;

ll BE(ll x,ll y){

ll res=1;

**while**(y){

**if**(y&1){

res=(res\*x)%mod;

}

y>>=1;

x=(x\*x)%mod;

}

**return** res;

}

ll F[N+2];

void PreCalFacts(){

ll res=1;

F[0]=1;

**for**(ll i=1;i<=N;i++){

res\*=i;

res%=mod;

F[i]=res;

}

}

ll nCr(ll n,ll r){

**if**(r>n)**return** 0LL;

**if**(r==n)**return** 1LL;

ll x=F[n];

ll y=(F[r]%mod\*F[n-r]%mod)%mod;

y = BE(y,mod-2);

ll ans=(x%mod \* y%mod)%mod;

**return** ans;

}

(a/b)%mod = (a%Q)/b

Where Q=mod\*b;

When b is little

String to int

stringstream ob;

        ob <<ss;

        ob >>data;

std::string s;

std::getline(std::cin >> std::ws, s);

input output fast

ios::sync\_with\_stdio(0);

cin.tie(0);

string devide by int or not

bool check(string &ss, long long k) {

    long long rem = 0;

    for (auto i : ss)

        rem = ((rem \* 10) + i - '0') % k;

    return rem == 0;

}

String char to int array

string s;

    getline(cin,s);

    int size *=* s.size();

    int arr[size]*=*{0};

    int k*=*0;

*for* (int i *=* 0; i*<*size; i*++*) {

*if* (s*[*i*]* *==* ' '){

            k*++*;

        }

*else* {

            arr[k] *=* arr[k] *\** 10 *+* (s*[*i*]* *-* 48);

        }

    }

1 2 3 4 5

You want find closest value… use in vector by

Auto it= upper\_bound(v.begin(),v.end(),data);

Upperbound a jamela nai..sob lower bound a somossa…lowerbound ak kothai kaj kore na..value thakle oitai return kore or or saite boro ta return kore..kinty upperbound ee sob boro element iterate kore…

Auto nt= lower\_bound(v.begin(),v.end(),data);

Cout<<\*it<<endl;

\*\*\*\*int ct=count(v.begin(),v.end(),element);

Element ta koto var ase..oita print korbe…

**String…**

Reverse(s.begin(),s.end());

Its give you string reverse…..O(N)

**Check 2 value sum is equal another value??(array)**

*while*(l*<=*h){

*if*(arr[l]*+*arr[h]*==*a){

          cout*<<*arr[l]*<<*"+"*<<*arr[h]*<<*endl;

          lk*++*;

*break*;

        }

*else* *if*(arr[l]*+*arr[h]*>*a){

          h*--*;

        }

*else* {

          l*++*;

        }

       }

*if*(lk*==*0){

        cout*<<*"NO WAY!"*<<*endl;

       }

      }

BIT MUSKING(DONG OF BIT)

1<<n == 2^n

SET BIT:

Print binery number:

void printbinery(int a){

*for*(int i*=*10;i*>=*0;i*--*) {

        cout*<<*((a*>>*i)*&*1);

    }

}

Find set bit:

t=9; (4th position)

*if*((t*&*(1*<<*3))*==*0){

    cout*<<*"NOT SETBIT"*<<*endl;

 }

*else* {

    cout*<<*"YES SETBIT"*<<*endl;

 }

DO unset:

A&(~(1<<i)))

 printbinery(t*&*0);

DO SET:

A|(1<<i)

int k*=*(1*<<*4)*-*1;

   printbinery(t*|*k);

count setbit number:

int count*=*0;

*for*(int i*=*31;i*>=*0;i*--*) {

*if*((t*&*(1*<<*i))*!=*0){

        count*++*;

    }

}

cout*<<*count*<<*endl;

buildin in function for count set bit:

for long value;

long long d*=*9;

cout*<<*\_\_builtin\_popcountll(d)*<<*endl;

for int value:

cout*<<*\_\_builtin\_popcount(t)*<<*endl;

uppercase to lower case in BIT:

char A=’A’;

char a=A|(1<<5);

cout<<a<<endl;

lower case to uppercase:

char B=b&(~(1<<5));

MULTIPLICATION an DIVITION:

5<<1==(2 diye gun)

5>>1==(2 diye vag)

LEFT AND RIGHT:

Left shift 0 add kore sese ;

Right shift delete kore last bit ke.

**XOR OPERETION:**

a^b^c==b^c^a==b^a^c

0^a==a

a^a==0

if I said… find odd value count in given even count all and one element is odd value count..(count number basically) ( ak ta number kotobar ase think oibabe) so how u find>?? In o(n) and o(1) space complixicity???

Ans:

2 2 3 3 4

Ans: 4 asbe;;;

Xor this all value;;;;;

For checking if a number is a power of 2 in O(1): if -> n^(n-1) = (2n-1), then it's a power of 2

SUBSET GENERETION:

*#include* <bits/stdc++.h>

*using* *namespace* std;

vector*<*vector*<*int*>>*subsetfi(vector*<*int*>&*nums) {

    int n*=*nums.size();

    int subcount*=*(1*<<*n);

    vector*<*vector*<*int*>>*subsetfi;

*for*(int i*=*0;i*<*subcount;i*++*) {

        vector*<*int*>*subset;

*for*(int j*=*0;j*<*n;j*++*) {

*if*((i*&*(1*<<*j))*!=*0){

                subset.push\_back(nums*[*j*]*);

            }

        }

        subsetfi.push\_back(subset);

    }

*return* subsetfi;

}

int main() {

ios::sync\_with\_stdio(0);

cin.tie(0);

int a;

cin*>>*a;

vector*<*int*>*v(a);

*for*(int i*=*0;i*<*a;i*++*) {

    cin*>>*v*[*i*]*;

}

auto allsub*=*subsetfi(v);

*for*(auto subset: allsub){

*for*(int ele: subset){

        cout*<<*ele*<<*" ";

    }

    cout*<<*endl;

}

}

Subset sum divisiable by K In O(N+K)

long long subCount(long long arr[], int N, long long K)

{

long long mod[K];

memset(mod, 0, sizeof(mod));

long long cumSum = 0;

for (long long i=0; i<N; i++)

{

cumSum += arr[i];

mod[((cumSum%K)+K)%K]++;

}

long long result = 0;

for (long long i=0; i<K ; i++)

if (mod[i] > 1)

result += (mod[i]\*(mod[i]-1))/2;

result += mod[0];

return result;

}

Find subarray sum in O(n)

int best = 0, sum = 0;

for (int k = 0; k < n; k++) {

sum = max(array[k],sum+array[k]);

best = max(best,sum);

}

cout << best << "\n";

Find max subarray sum with modulo each element:

long long maxSubarray(long long arr[], long long n, long long m)

{

    long long prefix *=* 0, maxim *=* 0;

    set*<*long long*>* S;

    S.insert(0);

*for* (int i *=* 0; i *<* n; i*++*)

    {

        prefix *=* (prefix *+* arr[i])*%*m;

        maxim *=* max(maxim, prefix);

        auto it *=* S.lower\_bound(prefix*+*1);

*if* (it *!=* S.end())

            maxim *=* max(maxim, prefix *-* (*\**it) *+* m );

        S.insert(prefix);

    }

*return* maxim;

}

Combinatrics:

N length er string er substring hoite pare🡪> (n(n+1))/2;

Substring== aksathe onnerk character thakbe

Sub sequence== akta string er jekono character

N length er string er sub sequence hoite pare🡪> 2n-1;

Combinations= 10 ta theke 2 ta niye koi vabe sajano jai.

Permutations= 10 ta theeke 2 ta exact value niye oder modde nijera odol bodol kore koi vabe sombob.

nC2==(n(n-1))/2;

if r==3 then…. ==n(n-1)(n-2)/(3\*2\*1);