// i started at 1:03 A.M. 1/2/2022 Author:: Gorssorser , nice to see you here , thnx for visiting

// somebody give me self control text me at my codeforces handle : gorssorser

// i am gorssorser <^v^>

#pragma GCC optimization ("O3")

#pragma GCC optimization ("unroll-loops")

#pragma GCC target("avx2,bmi,bmi2,lzcnt,popcnt")

#include <iostream>

#include <iomanip>

#include <stdio.h>

#include <limits.h>

#include <float.h>

#include <list>

#define ll long long int

#include <vector>

#include <map>

#include <unordered\_map>

#include <set>

#include <unordered\_set>

#include <queue>

#include <sstream>

#include <stack>

#include <deque>

#define dd cout<<"print"<<endl;

#include <algorithm>

#include<ext/pb\_ds/assoc\_container.hpp>

#include<ext/pb\_ds/tree\_policy.hpp>

#define mod1 998244353

#include <cmath>

#include <cstdlib>

#define run(a, m) for(int i = 0 ; i <m; i++ ) cin>>a[i];

#define run2(a, m) for(int i = 0 ; i <m; i++ ){ll v,u; \

cin>>u>>v; \

a[u]push\_back(v);}

#define bhag(n,a) for(int i = a ; i < n+a; i++)

#define jldi(i , n , a) for(int i = a ; i< n +a ; i++)

#define debmatrix(matrix){ cout<<"debuged matrix: "<<endl; for(auto i : matrix){for(auto j : i){cout<<j<<" "; } \

cout<<endl; }}

#define debpair(matrix){ cout<<"debuged pair: "<<endl; for(auto i : matrix){ cout<<i.first<<" "<<i.second<<endl; } \

cout<<endl; }

# define ff first

#define ss second

#define sz(a) a.size()

#define all(a) a.begin() , a.end()

# define debarr(arr){ cout<<"deb arr :"<<endl; for(auto i : arr){ cout<<i<<" "; } cout<<endl; }

# define debval(a){cout<<"var : "<<a<<endl;}

#define GNU\_optimised\_seg\_tree vector<ll>

//////////////////////////////////////////////////

using namespace std;

using namespace \_\_gnu\_pbds;

typedef tree<pair<ll,ll>, null\_type, less<pair<ll,ll>>, rb\_tree\_tag, tree\_order\_statistics\_node\_update> pbds;

using namespace std;

ll n, x, m, y ;

ll mod = 1e9+7;

ll mul(ll a,ll b, ll mod = mod)

{

return ((a%mod)\*(b%mod)+mod)%mod;

}

ll add(ll a,ll b)

{

return ((a%mod)+(b%mod)+mod)%mod;

}

ll power1(ll a, ll b, ll mod = mod)

{

ll ans = 1 ;

while(b)

{

if(b&1)

{

ans = mul(ans, a, mod) ;

}

a = mul(a, a, mod ) ;

b>>=1 ;

}

return ans ;

}

ll parent\_ (vector<ll>&parent, ll node )

{

if(parent[node] == -1 )

{

return node ;

}

else return parent[node] = parent\_(parent, parent[node]) ;

}

ll union\_(ll x, ll y, vector<ll> & parent, vector<ll> & size\_)

{

ll parx = parent\_(parent , x) ;

ll pary = parent\_ (parent , y) ;

if(size\_[parx] > size\_[pary])

{

parent[pary] = parx ;

size\_[parx] += size\_[pary] ;

}

else

{

parent[parx] = pary ;

size\_[pary] += size\_[parx] ;

}

return 0 ;

}

int solve()

{

return 0 ;

}

// u know sometime instead of trying to check for something just prebuild it in the correct order ,

// like sort or arrange the things accordinglyif

int main()

{

cin.tie(0)->sync\_with\_stdio(false);

cout.tie(NULL);

int t = 1 ;

cin>>t ;

int counter = t;

while(t--)

{

int bit = 0 ;

if(counter == 36 )bit = 1;

solve( );

// counter++;

}

return 0;

}

/\* greedy brute force dp3 binary search constructive pegion hole \*/

/\*some times negation is much preferred \*/

/\*some times u will fall but a babmoo tree1 take 4 yrs to start \*/

/\*

// a| b = a^b + a&b a^(a&b) = (a|b)^b b^(a&b) = (a|b)^a (a&b)^(a|b) = a^b

// a+b = a|b + a&b a+b = a^b + 2(a&b)

// a-b = (a^(a&b))-((a|b)^a) a-b = ((a|b)^b)-((a|b)^a) a-b = (a^(a&b))-(b^(a&b)) a-b = ((a|b)^b)-(b^(a&b))

\*/

/\*Type more ^v^\*/