#include "bits/stdc++.h"

using namespace std;

template<class T,class U>

ostream& operator<<(ostream& os,const pair<T,U>& p){

os<<"("<<p.first<<", "<<p.second<<")";

return os;

}

template<class T>

ostream& operator <<(ostream& os,const vector<T>& v){

os<<"{";

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

if(i)os<<", ";

os<<v[i];

}

os<<"}";

return os;

}

#ifdef LOCAL

#define cerr cout

#else

#endif

#define TRACE

#ifdef TRACE

#define trace(...) \_\_f(#\_\_VA\_ARGS\_\_, \_\_VA\_ARGS\_\_)

template <typename Arg1>

void \_\_f(const char\* name, Arg1&& arg1){

cerr << name << " : " << arg1 << std::endl;

}

template <typename Arg1, typename... Args>

void \_\_f(const char\* names, Arg1&& arg1, Args&&... args){

const char\* comma = strchr(names + 1, ',');cerr.write(names, comma - names) << " : " << arg1<<" | ";\_\_f(comma+1, args...);

}

#else

#define trace(...)

#endif

void \_\_print(int x) {cerr << x;}

void \_\_print(long x) {cerr << x;}

void \_\_print(long long x) {cerr << x;}

void \_\_print(unsigned x) {cerr << x;}

void \_\_print(unsigned long x) {cerr << x;}

void \_\_print(unsigned long long x) {cerr << x;}

void \_\_print(float x) {cerr << x;}

void \_\_print(double x) {cerr << x;}

void \_\_print(long double x) {cerr << x;}

void \_\_print(char x) {cerr << '\'' << x << '\'';}

void \_\_print(const char \*x) {cerr << '\"' << x << '\"';}

void \_\_print(const string &x) {cerr << '\"' << x << '\"';}

void \_\_print(bool x) {cerr << (x ? "true" : "false");}

template<typename T, typename V>

void \_\_print(const pair<T, V> &x) {cerr << '{'; \_\_print(x.first); cerr << ','; \_\_print(x.second); cerr << '}';}

template<typename T>

void \_\_print(const T &x) {int f = 0; cerr << '{'; for (auto &i: x) cerr << (f++ ? "," : ""), \_\_print(i); cerr << "}";}

void \_print() {cerr << "]\n";}

template <typename T, typename... V>

void \_print(T t, V... v) {\_\_print(t); if (sizeof...(v)) cerr << ", "; \_print(v...);}

#ifndef ONLINE\_JUDGE

#define debug(x...) cerr << "[" << #x << "] = ["; \_print(x)

#else

#define debug(x...)

#endif

#define rep(i, n) for(int i = 0; i < (n); ++i)

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

#define repD(i, a, n) for(int i = a; i >= (n); --i)

#define IOS ios :: sync\_with\_stdio(0), cin.tie(0), cout.tie(0), cin.exceptions(cin.failbit)

#define trav(a, x) for(auto& a : x)

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

#define rall(x) x.rbegin(), x.rend()

#define sz(x) (int)(x).size()

#define fill(a,b) memset(a, b, sizeof (a))

#define fi first

#define se second

#define lb lower\_bound

#define ub upper\_bound

#define mp make\_pair

#define pb push\_back

#define coun(x) \_\_builtin\_popcountll(x)

#define has\_bit(bit\_mask, x) ((bit\_mask) & (1LL << (x)))

#define turn\_on\_bit(bit\_mask, x) (bit\_mask |= (1LL << (x)))

#define turn\_off\_bit(bit\_mask, x) (bit\_mask &= (~(1LL << (x))))

#define smallest\_on\_bit(bit\_mask) (\_\_builtin\_ctzint((bit\_mask) & (-(bit\_mask))))

#define int long long

typedef long double ld;

typedef pair<int,int> pii;

typedef vector<int> vi;

const long long modb = 1000000007;

const long long inf = 0x3f3f3f3f3f3f3f3fLL; // Large int

const double oo = 1e15; // Large double

const double eps = 1e-8; // Small double, used for computational geometry

const double pi = acos(-1.0);

inline int gcd(int a, int b){ if (a == 0) return b; return gcd(b % a, a);}

inline int lcm(int a,int b){return (a\*b)/gcd(a, b);}

inline bool isPerfectSquare(int x){ int s = sqrt(x); return (s\*s == x);}

inline bool isFibonacci(int n){return isPerfectSquare(5\*n\*n + 4) || isPerfectSquare(5\*n\*n - 4);}

inline int add(int a, int b) {a += b; if(a >= modb) return a - modb; return a;}

inline int sub(int a, int b) {a -= b; if(a < 0) return a + modb; return a;}

inline int mul(int a, int b) {return ((((a%modb)\*(b%modb))%modb)+modb)%modb;}

inline int numofdig(int i) {return i > 0 ? (int) log10l ((ld) i) + 1 : 1;}

/\*void pre(){

int fact[N],ifact[N];

fact[0]=1,fact[1]=1;

ifact[0]=1,ifact[1]=1;

repA(i,2,N-1){

fact[i]=mul(fact[i-1],i);

ifact[i]=pwr(fact[i],modb-2);

}

}

int comb(int n,int r){

if(r>n)return 0;

return mul(fact[n],mul(ifact[r],ifact[n-r]));

}\*/

int pwr(int a,int b){

a %= modb;

int ans = 1;

while(b){

if(b & 1) ans = (ans \* a) % modb;

a = (a \* a) % modb;

b >>= 1;

}

return ans;

}

int32\_t main(){

IOS;

#ifndef ONLINE\_JUDGE

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

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

#endif // ONLINE\_JUDGE

int t;cin>>t;

rep(tc,t){

//cout<<"Case #"<<tc+1<<": ";

}

cerr << "\nTime elapsed: " << 1000 \* clock() / CLOCKS\_PER\_SEC << "ms\n";

return 0;

}