https://atcoder.jp/contests/abc190/tasks/abc190_d

Q.) How many arithmetic progressions consisting of integers with a common difference of 1 have a sum of n?

Approach:-

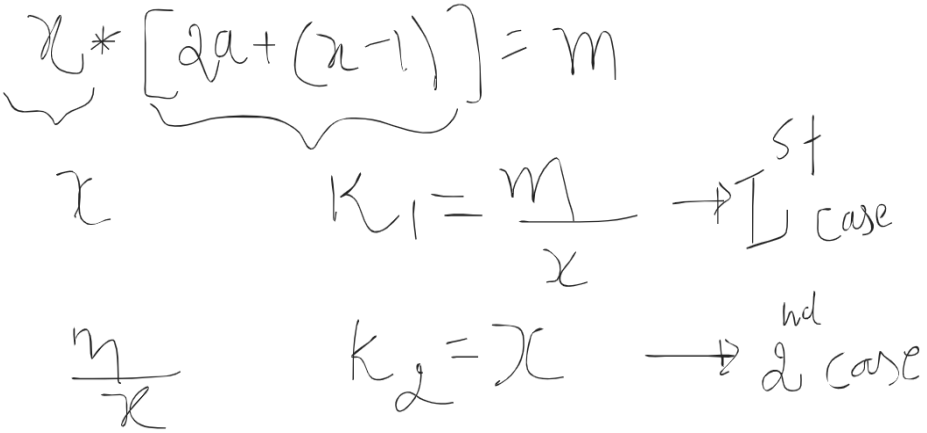
Series :- a,a+1,a+2,....,a+(x-1) [a-> first term, x-> length]

Sum= x*(2*a+(x-1))/2=n

x*(2*a+x-1)=2*n

x*(2*a+x-1)=m

Solution:-




```
#define ll long long int
ll n;
cin>>n;
ll m=2*n, ans=0;
for(ll x=1;x*x<=m;x++){
    if(m%x==0){
        ll k1=(m/x),k2=x;
        if((k1+1-x)%2==0) ans++;
        if(k2!=k1 && (k2+1-(m/x))%2==0) ans++;
        //If true then you got a series whose first term(a) and length=x
    }
}
cout<<ans<<endl;
```

Q:
Universe is defined by 2 elements -> s and t.
N universe are given .
Degree of connectivity is defined as
D(Ui,Uj) = Si+Sj+|ti-tj| , if |tui-tuj|<=m
0 Else

```
3 3 -> n and m
0 0 -> 1st universe
3 0 -> 2nd universe
9 2 -> 3rd universe
```

D(1,2) = 3
2<=n<=10^5
0<=m<=10^16
0<=si,ti<=10^16

Si+Sj+|ti-tj| -> Si+Sj+ti-tj . if ti>tj
Si+Sj+tj-ti, if tj>ti
Si+Sj+ti-tj = (si+ti) + (sj-tj)

Deque -> possible universes for that i that satisfy ti-t<=m

F…….b
B -> push i s-t =9
4,5,6

https://codeforces.com/problemset/problem/582/A

```
int main()
{
    int n;
    cin>>n;
    map<int,int,greater<int> > freq;
     // store frequency of all elements
     for(int i=0 ; i<n*n; i++)
    {
    int num;
    cin>>num;
    freq[num]++;
    }
vector<int> ans;
for(auto it=freq.begin(); it!=freq.end(); )
{
       if(it->second>0)
        {
             freq[(it->first)]--;
         for(int i=0; i<ans.size(); i++)
            {
                freq[__gcd(ans[i],(it->first))]-=2;
            }
             ans.push_back(it->first);
        }
    else
      {
        it++;
```

```
    }
}
for(int i=0; i<n; i++)
  {
    cout<<ans[i]<<" ";
  }
}
```

Time complexity: $O(n^2 \log(n))$