HELP RC

****
Author: Ramcharan Jehtu
****

Explaination:
We can find all pairs of an array using two nested loops(N<=1000).
In every repetation we wiil compute the greatest common divisor of pair and increase it's frequency by 1.(for storing frequency we can use either map or simple array of size 10000 beacause we know that
GCD of two elements can't be greater then max of these element).
There is no issue of time limit bacause n<=1000 and for computing GCD it will take O(log(n)) time so overall conmplexity will be O(n^2Log(n)) and for n<=1000 it will always take <1 second.

Code:

```
int gcd(int a, int b)                //function to calculate Greatest Common Divisor
{
   if (b == 0)
      return a;
   return gcd(b, a % b);

}

#include<bits/stdc++.h>
using namespace std;
int main()
{
   int T,i,j,n;
   cin>>T;
   while(T--)
   {
      cin>>n;
      int arr[n];

      for(i=0;i<n;i++)
      {
         cin>>arr[i];
      }
      int freq[10000]={0};
      //map<int,int> m;
      for(i=0;i<n;i++)
      {
         for(j=i+1;j<n;j++)
         {
            freq[gcd(arr[i],arr[j])]++;
         }
      }
      for(i=0;i<10000;i++)
      {
```

```
            if(freq[i]>0)
            {
                cout<<i<<" "<<freq[i]<<endl;
            }
        }
        /*for(auto it=m.begin();it!=m.end();it++)
        {
            cout<<it->first<<" "<<it->second<<endl;
        }*/
    }
}
```

Time Complexity: O(n^2)