/*Day 89 coding Statement :

You are given N integers. In each step you can choose some K of the remaining numbers and delete them,
if the following condition holds: Let the K numbers you've chosen be a1, a2, a3, ..., aK in sorted order.
Then, for each i ≤ K - 1, ai+1 must be greater than or equal to ai * C.

You are asked to calculate the maximum number of steps you can possibly make.*/

```
import java.lang.*;
import java.io.*;
class Main{
 public static void main(String[] args){
 Scanner sc=new Scanner(System.in);
 PrintWriter pw=new PrintWriter(System.out);
 int t=sc.nextInt();
 sc.nextLine();
 while(t-->0){
 int n=sc.nextInt();
 int k=sc.nextInt();
 long c=sc.nextInt();
 long a[]=new long[n];
 for(int i=0;i<n;i++)a[i]=sc.nextLong();
 Arrays.sort(a);
 int min=0;
 int max=n/k;
 int ans=0;
 while(min<=max){
 int mid=min+(max-min)/2;
 if(check(a,mid,k,c)){
 ans=mid;
 min=mid+1;
 }
 else
 max=mid-1;
 }
 pw.println(ans);
 }

 pw.close();
 }
 static boolean check(long[] a,int x,int k,long c){
 if(k*x>a.length) return false;
 if(x==0) return true;
 long v[][]=new long[k][x];
 for(int i=0;i<x;i++) v[0][i]=a[i];
 int s=x;
 for(int i=1;i<k;i++){
```

```
 for(int j=0;j<x;j++){
 boolean flag=false;
 while(s<a.length){
 if(a[s]>=c*v[i-1][j]){
 v[i][j]=a[s];
 s++; flag=true;
 break;
 }
 s++;
 }
 if(!flag) return false;
 }
 }

 return true;
 }
}
```