题目的意思是给出一组数 然后找出这组数乘积中的一个最小的至少三个质因子的数 （不包括1包括自己本身）；

实质就是这堆数中最小的两个质因子 然后相乘。

1.将每个数分解后储存，分解的原理就是从2开始到n的开方遍历，中间如果有质因子大于根号n的情况时，在另一个小因子整除时就存储起来；

2.将每个数的质因子存入数组，然后排列，第一个和第二个即该数组最小的因子，相乘即结果；

3.数字为1时不要存储因子，最后结果如果有1和另一个数就直接返回-1，这里如果过多写存储1之后再判断筛选会更麻烦；

将每个数分解质因子，维护最小的两个质因子即可。

#include <cstdio>

#include <cstring>

#include <algorithm>

#include<iostream>

#include<cmath>

#define ll long long

using namespace std;

bool is\_prime(int n)

{

if(n==2||n==3)

return 1;

if(n%6!=1&&n%6!=5)

return 0;

for(int i=5;i\*i<=n;i+=6)

{

if(n%i==0||n%(i+2)==0)

return 0;

}

return 1;

}

ll a[110];

ll b[110];

ll factor[1000000];

int main ()

{

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

int T,n;

cin>>T;

while(T--)

{

cin>>n;

int cnt(0);

for(int i=0;i<n;i++)

{

cin>>a[i];

for(int j=1;j<=sqrt(a[i]);j++)

if(a[i]%j==0)

{

factor[cnt++]=j;

factor[cnt++]=a[i]/j;

}

}

sort(factor,factor+cnt);

ll output=4000000000060000000;

int ji(0);

for(int i=0;i<cnt;i++)

{

if(factor[i]==1)continue;

if(is\_prime(factor[i]))

{

if(ji==2)break;

b[ji++]=factor[i];

}

else

output=min(output,factor[i]);

}

if(ji==2)

output=min(output,b[1]\*b[0]);

else if(output==4000000000060000000)

output=-1;

cout<<output<<"\n";

}

return 0;

}