

给定 n 个正整数，将它们分组，使得每组中任意两个数互质。至少要分成多少个组？

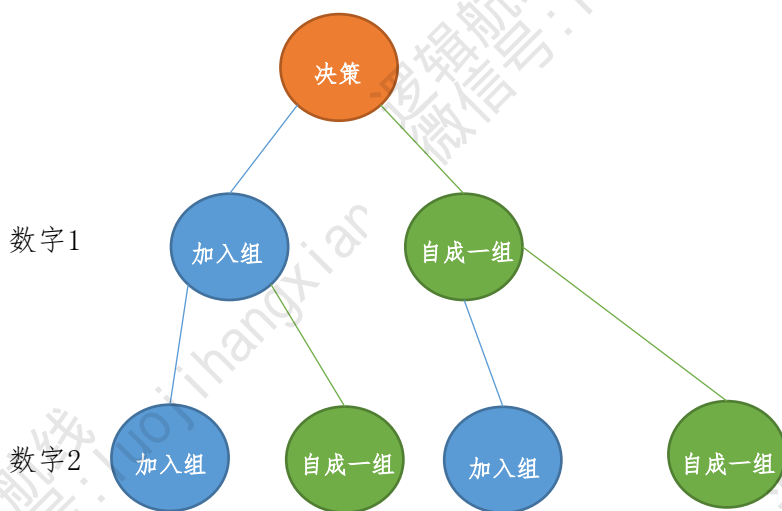
第二行是n个不大于10000的正整数。

一个正整数，即最少需要的组数。

6
14 20 33 117 143 175

3

首先来绘制决策树，很容易想到对于每一个数字来说，要么放到前面的某一个组合中，要么自成一组，因此决策树如下：



最终，我们只需要找到全部方案中，分组数目最少的那个即可。

编码

```
#include <bits/stdc++.h>
using namespace std;
long long vis[11], nums[11]; //乘法的值可能会很大, 因此使用long long
int n, cnt = 1000;

//step:第几个数字
//k: 当前拥有的组数
void dfs(int k, int step) {
    //完成遍历全部数字
    if (step == n + 1) {
        if (k < cnt) {
            cnt = k;
        }
        return;
    }

    //遍历全部分组, 尝试能否将当前数字纳入
    for (int i = 1; i <= k; i++) {
        //二者互质, 则加入
        if (__gcd(vis[i], nums[step]) == 1) {
            //用乘法的方式将所有互质数合并在一起
            vis[i] *= nums[step];
            //继续判断下一个数字
            dfs(k, step + 1);
            //取消选择
            vis[i] /= nums[step];
        }
    }

    //自成一组
    vis[k + 1] *= nums[step];
    //继续判断下一个数字
    dfs(k + 1, step + 1);
    //取消选择
    vis[k + 1] /= nums[step];
}

int main(int argc, char **argv) {
    cin >> n;
    for (int i = 1; i <= n; i++) {
        cin >> nums[i];
        vis[i] = 1;
    }
    dfs(1, 1);
    cout << cnt;
    return 0;
}
```

逻辑航线培优教育, 信息学奥赛培训专家。

扫码添加作者获取更多内容。

