大分部[22周赛 - Virtual Judge (vjudge.net)](https://vjudge.net/contest/548413#problem/H)

我的思路是没什么思路，只有一个两百次方的数肯定要用字符串来存

没见过此类大数取模的题，吃亏了

题解思路，字符串存起来，然后由同余定理，1234 % m等价于1000 % m + 200 % m + 3 % m + 4 % m,答案巧妙在于从左边开始枚举数位，组成一个数，来不断模m，通过数位操作+忽略后导零的同余使得这样的枚举取模合法

字符串起始位置根据正负的不同，起点也不同，for循环的i在外面定义，根据字符串正负来取不同的值作为for循环初始可以较好的解决这个问题

此外s.erase()也是不错的选择

二刷，忘记处理字符串-号了

// Each case starts with a line containing two integers a (-10200 ≤ a ≤ 10200) and b (|b| > 0, b fits into a 32 bit signed integer). Numbers will not contain any leading zeroes.

// a200次方,只能用string,

// b 32bit = 4byte,int

// 大数取模

// 从左开始枚举,不断取模,枚举的数要开long long,1e9\*10会爆int

// 要处理字符串的负号,s[0] ==  '-' 就改变for循环中的i

// 或者是 用s.erase(0,1);

#include <iostream>

#include <cstring>

#include <cstdio>

#include <algorithm>

using namespace std;

typedef long long ll;

int main(){

    int t;

    cin >> t;        int cnt = 1;

    while(t--){

        string s;

        int mod;

        cin >> s >> mod;

        ll ans = 0;

        if(s[0] == '-') s.erase(0,1);

        for(int i = 0;i < s.size();i++){

            ans = ((ans \* 10) % mod + (s[i] - '0') % mod) % mod;

        }

        if(!ans) printf("Case %lld: divisible",cnt++);

        else printf("Case %lld: not divisible",cnt++);

        puts("");

    }

    return 0;

}