**题意：**给出n个数，再给一个mod，然后现在有一种方法说是，可以把任意两个数连接起来，问你连接起来的对数取余mod等于0的有多少个

思路：乍一看没什么思路，暴力肯定不行，10^5的数n^2就炸了，我们就要想怎么去优化他，我们可以考虑预处理然后遍历一遍

**我们先给出一个例子** 给出n个数和一个mod，求多少对加起来mod等于0？

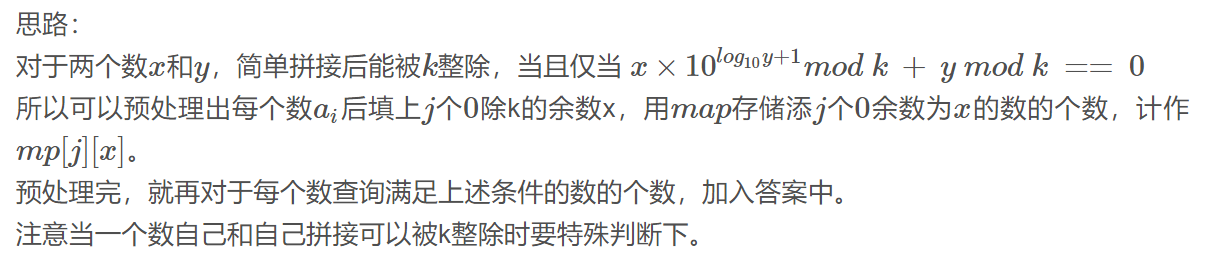
 这个n^2也不行，但是我们想一下如果我取余一个数=x 的话   我要什么情况才能让这个数加一个数%mod等于0

我们只有（x+y）%mod == 0     那在我们知道x的情况，我们只要找 mod-y的余数个数有多少即可

**同理**我们可以推理到这个题:因为是连接 12  34     我们相当于看成是  1200+34即可，就成功转移到了以上问题

因为连接不同的数的时候后面紧跟的0的个数不同，我们只要存下一个0到十个0所有的都用map存下即可

然后特别的，最后判断下自身加自身会不会也可以被mod为0，如果是的话-1



#include<iostream>

#include<cmath>

#include<bits/stdc++.h>

using namespace std;

#define ll long long

ll a[200010];

map<int,int>\_map[20];

int main()

{

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

int n;

ll k;

scanf("%d%I64d",&n,&k);

for(int i=1;i<=n;i++)

{

scanf("%I64d",&a[i]);

ll x=a[i];

for(int j=1;j<=10;j++)

{

x\*=10;

x%=k;

\_map[j][x]++;

}

}

ll ans=0;

for(int i=1;i<=n;i++)

{

int len=log10(a[i])+1;

ans+=\_map[len][(k-a[i]%k)%k];

ll x=1;

for(int j=1;j<=len;j++)

x=x\*10%k;

if(((x\*a[i])%k+a[i]%k)%k==0)

ans-=1;

}

printf("%I64d\n",ans);

return 0;

}