题意：给你一组数，从中找出一些数，求和是m的倍数。

最简单的方法就是dp，定义一个布尔值数组dp[n][m]，

dp[i][j]=1表示前i个物品存在子序列模m后值为j。

每次对第i+1个物品，考虑放或者不放，然后状态转移即可。

最后看dp[n][0]是不是1，是的话就可以

但是n的范围是1e6，dp会爆呀

根据鸽巢原理，n>m时一定有解，所以只要算n<m的部分就行了，这样dp数组范围就缩到1e3了

#include <bits/stdc++.h>

#define ll long long

using namespace std;

bool dp[1010][1010];

int a[1000010];

int main()

{

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

int n,m;

scanf("%d%d",&n,&m);

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

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

if(n>m)

{

puts("YES");

return 0;

}

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

dp[i+1][a[i]%m]=1;//为什么是i+1看下面

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

for(int j=0;j<m;j++)

{

dp[i+1][j]=dp[i][j]||dp[i+1][j];

dp[i+1][(j+a[i])%m]=dp[i+1][(j+a[i])%m]||dp[i][j];

}

//第i个数的贡献到dp[i+1]去看

if(dp[n][0])

puts("YES");

else puts("NO");

return 0;

}