数字游戏

题目描述:

小明玩一个游戏。系统发 1+n 张牌,每张牌上有一个整数。第一张给小明,后 n 张按照发牌顺序排成连续的一行。需要小明判断,后 n 张牌中,是否存在连续的若干张牌,其和可以整除小明手中牌上的数字。

输入描述:

输入数据有多组,每组输入数据有两行,输入到文件结尾结束。

第一行有两个整数 n 和 m, 空格隔开。m 代表发给小明牌上的数字。

第二行有 n 个数, 代表后续发的 n 张牌上的数字, 以空格隔开。

输出描述:

对每组输入,如果存在满足条件的连续若干张牌,则输出1;否则,输出0

补充说明:

1 <= n <= 1000

1 <= 牌上的整数 <= 400000

输入的组数,不多于1000

用例确保输入都正确,不需要考虑非法情况。

示例 1

输入:

6 7

2 12 6 3 5 5

10 11

1 1 1 1 1 1 1 1 1 1

输出:

1

 \cap

说明:

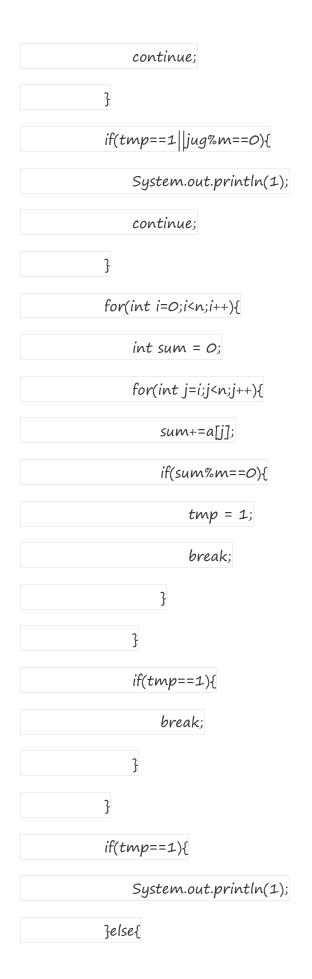
两组输入。第一组小明牌的数字为 7, 再发了 6 张牌。第 1、2 两张牌数字和为 14, 可以

整除 7,输出 1。第二组小明牌的数字为 11,再发了 10 张牌,这 10 张牌数字和为 10,

无法整除 **11**,输出 **0**。

import java.util.Scanner;

```
// 注意类名必须为 Main, 不要有任何 package xxx 信息
public class Main {
   public static void main(String[] args) {
       Scanner in = new Scanner(System.in);
       // 注意 hasNext 和 hasNextLine 的区别
       while (in.hasNextInt()) { // 注意 while 处理多个 case
           int n = in.nextInt();
           int m = in.nextInt();
           int[] a = new int[1005];
           //int[] mp = new int[400005];
           int tmp = 0;
           int jug = 0;
           for(int i=0;i<n;i++){
               a[i] = in.nextInt();
              jug+=a[i];
               if(a[i]%m==0){
                  tmp = 1;
               }
           }
           if(jug<m){
               System.out.println(0);
```



			System.out.println(0);
		}	
	}		
}			
}			