**Admirable Professor**

Time limit:

Memory limit: 512MB

Oo now becomes a famous professor. He has a lot of students with the goal of becoming UCPC Champions. UCPC is a challenging programming competition; it is similar to ICPC but each team has students instead of 3. This year, Oo is so proud to have the strongest rosters ever, which are divided into teams.

Today is Oo’s birthday and all team members are gathering at Oo’s house. Each of them brings Oo a special gift: a balloon with a lot of candies inside. Oo is so thrilled, so he does not want to eat all those cute candies alone! Therefore, he decides to distribute all balloons back to the students. Each student receives a balloon randomly. To make it more exciting, all members in a team will gather and share with each other their balloons and candies.

However, a problem appears! Sometimes, candies cannot be divided evenly to team members. This is when the admirable professor comes to help. Oo loves his students so much (at least today), so in each team, he will take only the remainder of total candies dividing by number of members. For example, if Oo has 2 teams, each team has 3 members, the first team has 10 candies and the second has 8, Oo will take 1 candy from the first team and 2 candies from second one. Overall, he has 3 candies.

Again, Oo loves his students so much and he is thinking about a wonderful scenario, where he will take as few candies as possible. It should be an easy problem for Oo, just not today… Please help Oo find the minimum number of candies he would take and design a scenario to achieve it!

**Input**

The first line of the input contains 2 positive integers – the number of students in a team and number of teams

The next line contains positive integers , the number of candies in each balloon.

**Output**

The first line contains one integer, the minimum number of candies Oo could take.

The -th line of next lines contains integers representing for number of candies in balloons the -th team gets. If there are more than one scenario to achieve the result, print any of them.

**Sample 1**

|  |  |
| --- | --- |
| 2 2  1 2 3 4 | 0  1 3  2 4 |

**Sample 2:**

|  |  |
| --- | --- |
| 2 3  2 3 5 7 11 13 | 1  2 5  3 13  11 7 |

**Explanation:**

In the first sample, team 1 has 4 candies, so each member takes 2 and Oo has 0! Team 2 has 6 candies and Oo also takes 0! Overall, Oo is happy!

In the second sample, there is no scenario that Oo would get 0 candy. However, 1 is not bad… ((2 + 5) mod 2 + (3 + 13) mod 2 + (11 + 7) mod 2 = 1)