1. Rose is a professor of music at the University of Melody. She has organized a group singing competition for the students of her class. Rose decides the initial size of the groups. The competition is organized in such a way that all the groups will be singing on stage together at the same time. The music department has some mics to distribute among the groups. Each group will be allotted exactly one mic. In case they end up with surplus mics, Rose will split the groups in such a way as to minimize the number of students sharing a single mic. Each initial group can only be split into two. However the new groups that are formed can be split further if necessary. Rose wishes to find the number of students in the largest group that is sharing a mic after the regrouping process.

Write an algorithm to help Rose calculate the number of students in the largest group that is sharing a mic.

Input

The first line of the input consists of two space-separated integers - N and M. representing the number of initial groups and number of mics available, respectively. The next line consists of N space-separated integers-arri, arr...arry representing the initial size

of each group.

Output

Print an integer representing the number of students in the largest group sharing a mic after re

grouping Constraints

1sNs105

1s Ms 106

1sarris 106

151≤ N

Example

Input:

57

108643

Output:

6

Explanation:

The number of groups is 5 and the number of mics available is 7. Surplus available mics are 2(7-5) so two groups are split.

As the number of students sharing a mic should be minimized so the maximum sized group is split.

Step 1:

The maximum group size is 10.

This group is split into two equal groups: 5, 5.

There are 6 groups of size: [5, 5, 8, 6, 4, 31 Now, the surplus mic available is 1.

Step 2:

The maximum group size is 8, split this group into two equal groups: 4,4 Now there are 7 groups of size: 15, 5, 4, 4, 6, 4, 3]

So, the output will be 6 as now it is the maximum group size.

3. The MNC "CompSoft" maintains a hierarchy for its employees. In the hierarchy there are N employees with employee ID labeled from 1 to N. The CEO of the company is the head of the hierarchy with ID G. If

employee X is working under employee Y, then Y will be the manager of X and so on.

At the end of the year the company had made extra profit on some of its projects so they have decided to award some employees with bonuses. The company uses this hierarchy to distribute bonuses among its employees. If an employee has worked on a project then the manager of the employee will be given bonus. For example in the given hierarchy X-Y-Z where Y reports to X and Z reports to Y, if employee Z has worked on the project then bonus will be offered to the highest person in the hierarchy i.e X. IfX has already got the bonus earlier then Y will get the bonus and in case Y also has earlier got the bonus then Z will get the bonus. Similar bonus allocation pattern will be followed for all the employees in the hierarchy. The head of the hierarchy ie the CEO is not included in this process of bonus distribution.

Write an algorithm to find the IDs of the employees who won't receive a bonus.

Input

The first line of the input consists of two space-separated integers -numEmployess (N) and numProjects(P), representing the total number of employees in this hierarchy and the total number of projects worked upon respectively. The second line consists of N space-separated integers representing the managers IDs where the

position of the manager in the list depicts the employee ID working under that manager. The CEO will have the ID 0. The third line consists of P space-separated integers representing the IDs of the employees who worked on the project…

Question

Output

Print space-separated integers representing the IDs of the employees who did not receive the bonus. The IDs should be sorted in ascending order

Constraints

0 ≤ numEmployess ≤ 900

Examples

Input:

84

01122344

8465

Output:

4678

Explanation:

The hierarchy is as follows:

0

1

1

2

3

4 56

7 8

The employee with ID 8 has worked on the project therefore the bonus will be given to the manager with employee ID 1. The employee with ID 4 has worked on the project therefore the bonus will be given to the manager with employee ID 2. The employee with ID 6 has worked on the project therefore the bonus will be given to the manager with employee ID 3.

The employee with ID 5 has worked on the project but the manager with ID 2 and the manager's manager with ID 1 have both received the bonus earlier hence the bonus will be given to 5. So, the employees with IDs 4, 6, 7 and 8 can't receive the bonus

2. A fully automated Cooking Championship has been organized and N chefs are participating. Each chef has a winning score associated with them. A positive score denotes that the chef has won more rounds than they have lost, and a negative value denotes that the chef has lost more rounds than they have won. There are N chefs and the cookery management has to form two teams of chefs using an automated system. The system divides the chefs into two teams where the difference between the scores of the two teams should be as high as possible. Each team can consist of any number of chefs. It is not possible to have chefs with the same score within the same team.

Write an algorithm to find the maximum possible difference in the scores of the two teams according to the automated system.

Input

The first line of the input consists of an integer - num, representing the number of chefs participating in the championship (N).

The second line consists of N space-separated integers score score ScoreN.1, representing the scores of the N participating chefs.

Output

Print an integer representing the maximum possible difference in the scores of the two teams according

to the mentioned criteria.

Constraints

0< nums 105

0<score scores Score 10

Note

Both the teams must collectively contain the scores of all the chefs, No chef should be left after dividing into two teams.

The chefs with the same scores cannot exist in the same team. It is possible that all the chefs are in the same team in the case when their scores are unique and either all are positive, or all are negative. In such a case, the score of the other team will be zero as there will be no chef in that team.

Examples

Example 1

Input:

4

58-17

Output:

21

Explanation:

The two teams will be (5,8.7) and (-1) and the difference will be 5+8+7-20-(-1)=21.

Thus, the maximum difference between the two teams will be 21.

Example 2

Input:

-10-20-30

Output:

60

Explanation

As all the cores are Unique so can be taken into one team The two teams will be (0) and (-10-(-20)+(30) --60). The maximum difference between the scores of the two teams will be 0-1-60)-60.