Question 1

**Problem Description:**

You have trouble remembering which type of triangle is which. You write a program to help. Your program reads in three angles (in degrees).

• If all three angles are 60, output **Equilateral**.

• If the three angles add up to 180 and exactly two of the angles are the same, output **Isosceles**.

• If the three angles add up to 180 and no two angles are the same, output **Scalene**.

• If the three angles do not add up to 180, output **Error**.

Input Specification The input consists of three integers, each on a separate line. Each integer will be greater than 0 and less than 180.

Output Specification Exactly one of Equilateral, Isosceles, Scalene or Error will be printed on one line.

**Sample Input1**:

60

70

50

**Output for Sample Input1:**

Scalene

**Sample Input**:

60

75

55

**Output for Sample Input**

Error

Question 2

**Problem Description:**

A vote is held after singer A, B and singer C compete in the final round of a singing competition. Your job is to count the votes and determine the outcome.

**Input Specification**

The input will be two lines. The first line will contain V (1 ≤ V ≤ 15), the total number of votes. The second line of input will be a sequence of V characters, each of which will be A, B or C, representing the votes for a particular singer.

**Output Specification**

The output will be the sequence of A, B, C according to their votes. If the vote is same between two singer, any of their sequence is acceptable.

**Sample Input**

12

AACBCBACACBC

**Output for Sample Input**

CAB

Question 3

**Problem Description:**

You are hosting a party and do not have room to invite all of your friends. You use the following unemotional mathematical method to determine which friends to invite. Number your friends 1, 2, . . . , K and place them in a list in this order. Then perform m rounds. In each round, use a number to determine which friends to remove from the ordered list. The rounds will use numbers r1, r2, . . . , rm. In round i remove all the remaining people in positions that are multiples of ri (that is, ri , 2ri , 3ri , . . .) The beginning of the list is position 1. Output the numbers of the friends that remain after this removal process.

**Input Specification**

The first line of input contains the integer K (1 ≤ K ≤ 100).

The second line of input contains the integer m (1 ≤ m ≤ 10), which is the number of rounds of removal.

The next m lines each contain one integer. The ith of these lines (1 ≤ i ≤ m) contains ri ( 2 ≤ ri ≤ 100) indicating that every person at a position which is multiple of ri should be removed. **Output Specification**

The output is the integers assigned to friends who were not removed. One integer is printed per line in increasing sorted order.

**Sample Input**

10

2

2

3

**Output for Sample Input**

1

3

7

9

**Explanation of Output for Sample Input Initially**

our list of invitees is 1, 2, 3, 4, 5, 6, 7, 8, 9, 10. There will be two rounds of removals. After the first round of removals, we remove the even positions (i.e., every second position), which causes our list of invitees to be 1, 3, 5, 7, 9. After the second round of removals, we remove every 3rd remaining invitee: thus, we keep 1 and 3, remove 5 and keep 7 and 9, which leaves us with an invitee list of 1, 3, 7, 9.

Question 4

**Problem Description:**

The CEMC is organizing a workshop with an activity involving pairs of students. They decided to assign partners ahead of time. You need to determine if they did this consistently. That is, whenever A is a partner of B, then B is also a partner of A, and no one is a partner of themselves.

**Input Specification**

The input consists of three lines. The first line consists of an integer N (1 < N ≤ 30), which is the number of students in the class. The second line contains the first names of the N students separated by single spaces. (Names contain only uppercase or lowercase letters, and no two students have the same first name). The third line contains the same N names in some order, separated by single spaces. The positions of the names in the last two lines indicate the assignment of partners: the ith name on the second line is the assigned partner of the ith name on the third line.

**Output Specification**

The output will be **good** if the two lists of names are arranged consistently, and **bad** if the arrangement of partners is not consistent.

**Sample Input 1**

4

Ada Alan Grace John

John Grace Alan Ada

**Output for Sample Input 1**

good

**Explanation for Output for Sample Input 1**

Ada and John are partners, and Alan and Grace are partners. This arrangement is consistent.

**Sample Input 2**

7

Rich Graeme Michelle Sandy Vlado Ron Jacob

Ron Vlado Sandy Michelle Rich Graeme Jacob

**Output for Sample Input 2**

bad