

# Problem A. Interior Design

**Time Limit** 1000 ms

**Code Length Limit** 50000 B

**OS** Linux

Chef decided to redecorate his house, and now needs to decide between two different styles of interior design.

For the first style, tiling the floor will cost  $X_1$  rupees and painting the walls will cost  $Y_1$  rupees.

For the second style, tiling the floor will cost  $X_2$  rupees and painting the walls will cost  $Y_2$  rupees.

Chef will choose whichever style has the lower total cost. How much will Chef pay for his interior design?

## Input Format

- The first line of input will contain a single integer  $T$ , denoting the number of test cases.
- Each test case consists of a single line of input, containing 4 space-separated integers  $X_1, Y_1, X_2, Y_2$  as described in the statement.

## Output Format

For each test case, output on a new line the amount Chef will pay for interior design.

## Constraints

- $1 \leq T \leq 100$
- $1 \leq X_1, Y_1, X_2, Y_2 \leq 100$

## Sample 1

Input	Output
4	30
10 20 9 25	29
10 20 9 20	30
10 20 20 10	143
100 43 85 61	

\*\*Test case 1:\*\* The first style costs  $10 + 20 = 30$  rupees, and the second costs  $9 + 25 = 34$  rupees. The first is cheaper, so Chef will pay 30 rupees.

**Test case 2:** The first style costs  $10 + 20 = 30$  rupees, and the second costs  $9 + 20 = 29$  rupees. The second is cheaper, so Chef will pay 29 rupees.

**Test case 3:** The first style costs  $10 + 20 = 30$  rupees, and the second costs  $20 + 10 = 30$  rupees. Both styles cost the same, so Chef is always going to pay 30 rupees.

**Test case 4:** The first style costs  $100 + 43 = 143$  rupees, and the second costs  $85 + 61 = 146$  rupees. The first is cheaper, so Chef will pay 143 rupees.

## Problem B. Divisors

**Time Limit** 1000 ms

**Mem Limit** 524288 kB

Read an integer variable and print all of its divisors (including 1 and the number itself).

### Input

The input will contain an integer  $A$  ( $0 < A < 100$ ).

### Output

Print the divisors in increasing order, one per line.

### Sample

Input	Output
6	1 2 3 6

## Problem C. GPA Calculator

**Time Limit** 1000 ms

**Mem Limit** 524288 kB

Shibli is a very good student. He is very regular and attentive in his classes. He is very serious about his study. So he makes good result in every semester.

Recently his previous semester result has published. Since Shibli is very busy with his study he asks his good friend Swapnil who is a very good programmer to calculate his Grade Point Average (GPA).

### Input

Input starts with an integer  $T$  ( $T \leq 5$ ), denoting the number of test cases. Each test case starts with an integer  $N$  ( $N \leq 8$ ), denoting the number of subjects. Next  $N$  line contains a floating point number  $p$  ( $2 \leq p \leq 4$ ) denoting the grade point Shibli got in each subject of previous semester. And you can assume that  $p$  contains four digits after the decimal point.

### Output

For each test case, print a line **Case x:**  $y$  where  $x$  is replaced by the test case number and  $y$  is the GPA of previous semester Shibli got, rounded to three places after the decimal point.

### Sample

Input	Output
2 5 3.4252 2.8526 3.9685 2.8754 3.8568 4 2.8569 3.2154 2.2369 4.0000	Case 1: 3.396 Case 2: 3.077

## Problem D. Instagram

**Time Limit** 1000 ms

**Code Length Limit** 50000 B

**OS** Linux

Chef categorises an instagram account as *spam*, if, the *following* count of the account is more than 10 times the count of *followers*.

Given the *following* and *follower* count of an account as  $X$  and  $Y$  respectively, find whether it is a *spam* account.

### Input Format

- The first line of input will contain a single integer  $T$ , denoting the number of test cases.
- Each test case consists of two space-separated integers  $X$  and  $Y$  — the *following* and *follower* count of an account, respectively.

### Output Format

For each test case, output on a new line, YES, if the account is *spam* and NO otherwise.

You may print each character of the string in uppercase or lowercase. For example, the strings YES, yes, Yes and yES are identical.

### Constraints

- $1 \leq T \leq 100$
- $1 \leq X, Y \leq 100$

### Sample 1

Input	Output
4	NO
1 10	NO
10 1	YES
11 1	YES
97 7	

**\*\*Test case 1:\*\*** The following count is 1 while the follower count is 10. Since the following count is not more than 10 times the follower count, the account is not spam.

**Test case 2:** The following count is 10 while the follower count is 1. Since the following count is not **more** than 10 times the follower count, the account is not spam.

**Test case 3:** The following count is 11 while the follower count is 1. Since the following count is more than 10 times the follower count, the account is spam.

**Test case 4:** The following count is 97 while the follower count is 7. Since the following count is more than 10 times the follower count, the account is spam.

## Problem E. Students Marks Sum

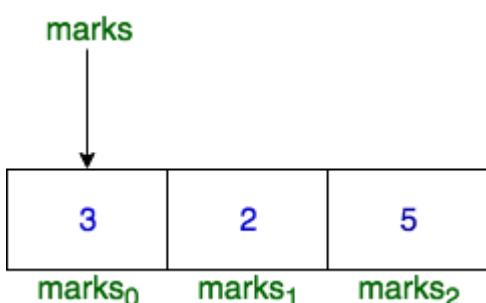
OS Linux

You are given an array of integers,  $\text{marks}$ , denoting the marks scored by students in a class.

- The alternating elements  $\text{marks}_0$ ,  $\text{marks}_2$ ,  $\text{marks}_4$  and so on denote the marks of boys.
- Similarly,  $\text{marks}_1$ ,  $\text{marks}_3$ ,  $\text{marks}_5$  and so on denote the marks of girls.

The array name,  $\text{marks}$ , works as a pointer which stores the base address of that array. In other words,  $\text{marks}$  contains the address where  $\text{marks}_0$  is stored in the memory.

For example, let  $\text{marks} = [3, 2, 5]$  and  $\text{marks}$  stores `0x7fff9575c05f`. Then, `0x7fff9575c05f` is the memory address of  $\text{marks}_0$ .



### Function Description

Complete the function, `marks_summation` in the editor below.

`marks_summation` has the following parameters:

- `int marks[number_of_students]`: the marks for each student
- `int number_of_students`: the size of `marks[]`
- `char gender`: either 'g' or 'b'

### Returns

- `int`: the sum of marks for boys if  $\text{gender} = b$ , or of marks of girls if  $\text{gender} = g$

### Input Format

- The first line contains  $\text{number\_of\_students}$ , denoting the number of students in the class, hence the number of elements in  $\text{marks}$ .
- Each of the  $\text{number\_of\_students}$  subsequent lines contains  $\text{marks}_i$ .
- The next line contains  $\text{gender}$ .

## Constraints

- $1 \leq \text{number\_of\_students} \leq 10^3$
- $1 \leq \text{marks}_i \leq 10^3$  (where  $0 \leq i < \text{number\_of\_students}$ )
- $\text{gender} = g$  or  $b$

Input	Output
3 3 2 5 b	8

## Explanation 0

$\text{marks} = [3, 2, 5]$  and  $\text{gender} = b$ .

So,  $\text{marks}_0 + \text{marks}_2 = 3 + 5 = 8$ .

Input	Output
5 1 2 3 4 5 9	6

## Explanation 1

$\text{marks} = [1, 2, 3, 4, 5]$  and  $\text{gender} = g$

So,  $\text{sum} = \text{marks}_1 + \text{marks}_3 = 2 + 4 = 6$ .

Input	Output
1 5 g	0

## Explanation 2

$\text{marks} = [5]$  and  $\text{gender} = g$

Here,  $\text{marks}_1$  does not exist. So,  $\text{sum} = 0$ .

## Problem F. Get Subscription

**Time Limit** 1000 ms

**Code Length Limit** 50000 B

**OS** Linux

Chef wants to conduct a lecture for which he needs to set up an online meeting of exactly  $X$  minutes.

The meeting platform supports a meeting of maximum 30 minutes without subscription and a meeting of unlimited duration with subscription.

Determine whether Chef needs to take a subscription or not for setting up the meet.

### Input Format

- First line will contain  $T$ , the number of test cases. Then the test cases follow.
- Each test case contains a single integer  $X$  - denoting the duration of the lecture.

### Output Format

For each test case, print in a single line, YES if Chef needs to take the subscription, otherwise print NO.

You may print each character of the string in uppercase or lowercase (for example, the strings YES, yEs, yes, and yeS will all be treated as identical).

### Constraints

- $1 \leq T \leq 100$
- $1 \leq X \leq 100$

### Sample 1

Input	Output
4	YES
50	NO
3	NO
30	YES
80	

\*\*Test Case 1:\*\* Without subscription, the platform allows only 30 minutes of duration.

Since Chef needs to conduct a lecture of 50 minutes, he needs to buy the subscription.

**Test Case 2:** Without subscription, the platform allows 30 minutes of duration. Since Chef needs to conduct a lecture of 3 minutes only, he does not need to buy the subscription.

**Test Case 3:** Without subscription, the platform allows 30 minutes of duration. Since Chef needs to conduct a lecture of 30 minutes only, he does not need to buy the subscription.

**Test Case 4:** Without subscription, the platform allows only 30 minutes of duration. Since Chef needs to conduct a lecture of 80 minutes, he needs to buy the subscription.

## Problem G. Bidding

**Time Limit** 500 ms

**Code Length Limit** 50000 B

**OS** Linux

Alice, Bob and Charlie are bidding for an artifact at an auction.

Alice bids  $A$  rupees, Bob bids  $B$  rupees, and Charlie bids  $C$  rupees (where  $A$ ,  $B$ , and  $C$  are **distinct**).

According to the rules of the auction, the person who bids the **highest** amount will win the auction.

Determine who will win the auction.

### Input Format

- The first line contains a single integer  $T$  — the number of test cases. Then the test cases follow.
- The first and only line of each test case contains three integers  $A$ ,  $B$ , and  $C$ , — the amount bid by Alice, Bob, and Charlie respectively.

### Output Format

For each test case, output who (out of **Alice**, **Bob**, and **Charlie**) will win the auction.

You may print each character of **Alice**, **Bob**, and **Charlie** in uppercase or lowercase (for example, **ALICE**, **aliCe**, **aLICe** will be considered identical).

### Constraints

- $1 \leq T \leq 1000$
- $1 \leq A, B, C \leq 1000$
- $A$ ,  $B$ , and  $C$  are **distinct**.

### Sample 1

Input	Output
4 200 100 400 155 1000 566 736 234 470 124 67 2	Charlie Bob Alice Alice

\*\*Test Case 1:\*\* Charlie wins the auction since he bid the highest amount.

**Test Case 2:** Bob wins the auction since he bid the highest amount.

**Test Case 3:** Alice wins the auction since she bid the highest amount.

**Test Case 4:** Alice wins the auction since she bid the highest amount.

## Problem H. Netflix

**Time Limit** 1000 ms

**Code Length Limit** 50000 B

**OS** Linux

Alice, Bob, and Charlie are contributing to buy a Netflix subscription. However, Netflix allows only two users to share a subscription.

Given that Alice, Bob, and Charlie have  $A$ ,  $B$ , and  $C$  rupees respectively and a Netflix subscription costs  $X$  rupees, find whether any two of them can contribute to buy a subscription.

### Input Format

- The first line of input will contain a single integer  $T$ , denoting the number of test cases.
- Each test case contains four space-separated integers  $A$ ,  $B$ ,  $C$ , and  $X$  — the amount that Alice, Bob, and Charlie have, and the cost of a Netflix subscription respectively.

### Output Format

For each test case, output **YES**, if any two of Alice, Bob, and Charlie can contribute to buy a Netflix subscription or **NO** otherwise.

You may print each character in uppercase or lowercase. For example, **NO**, **no**, **No**, and **n0** are all considered identical.

### Constraints

- $1 \leq T \leq 1000$
- $1 \leq A, B, C, X \leq 100$

### Sample 1

Input	Output
4	NO
1 1 1 3	YES
2 3 1 5	YES
4 2 3 4	NO
2 1 4 7	

\*\*Test case 1:\*\* No two people can contribute to collect enough money to buy a Netflix subscription.

**Test case 2:** Alice and Bob can contribute and collect a total of 5 rupees which is enough to buy a Netflix subscription of 5 rupees.

**Test case 3:** One possible way is, Bob and Charlie can contribute and collect a total of 5 rupees which is enough to buy a Netflix subscription of 4 rupees.

Note that there are other possible ways as well.

**Test case 4:** No two people can contribute to collect enough money to buy a Netflix subscription.

# Problem I. The Gift

**Time Limit** 1000 ms

**Code Length Limit** 50000 B

**OS** Linux

Om has  $X$  rupees. He wants to gift a laptop worth  $N$  rupees to his girlfriend.

We know that Om is the technical secretary of IIIT-A and has access to the Gymkhana funds of IIIT-A. Currently there are  $M$  rupees in the fund and Om can use the fund as much as he wants.

Find whether Om can gift his girlfriend a new laptop.

## Input Format

- The first and only line of input contains three space-separated integers  $X$ ,  $N$ , and  $M$  — the amount Om has, the price of the laptop, and the amount present in the Gymkhana fund respectively.

## Output Format

For each input, output **YES** if Om can buy the laptop and **NO** otherwise.

You may print each character in uppercase or lowercase. For example **YES**, **Yes**, **yes**, and **yES** are all considered the same.

## Constraints

- $1 \leq X, N, M \leq 10^3$

## Sample 1

Input	Output
5 10 15	YES

Om uses 5 rupees from Gymkhana fund. So, the amount he has is  $5 + 5 = 10$  rupees. He can buy the laptop with cost 10 rupees.

### Sample 2

Input	Output
4 50 7	NO

Even if Om uses the whole Gymkhana fund, he won't be able to buy the laptop.

# Problem J. The Cheaper Cab

**Time Limit** 500 ms

**Code Length Limit** 50000 B

**OS** Linux

Chef has to travel to another place. For this, he can avail any one of two cab services.

- The first cab service charges  $X$  rupees.
- The second cab service charges  $Y$  rupees.

Chef wants to spend the **minimum** amount of money. Which cab service should Chef take?

## Input Format

- The first line will contain  $T$  – the number of test cases. Then the test cases follow.
- The first and only line of each test case contains two integers  $X$  and  $Y$  – the prices of first and second cab services respectively.

## Output Format

For each test case, output **FIRST** if the first cab service is cheaper, output **SECOND** if the second cab service is cheaper, output **ANY** if both cab services have the same price.

You may print each character of **FIRST**, **SECOND** and **ANY** in uppercase or lowercase (for example, **any**, **aNy**, **Any** will be considered identical).

## Constraints

- $1 \leq T \leq 100$
- $1 \leq X, Y \leq 100$

## Sample 1

Input	Output
3 30 65 42 42 90 50	FIRST ANY SECOND

\*\*Test case 1:\*\* The first cab service is cheaper than the second cab service.

**Test case 2:** Both the cab services have the same price.

**Test case 3:** The second cab service is cheaper than the first cab service.

# Problem K. C\*++ Calculations

**Time Limit** 2000 ms

**Mem Limit** 65536 kB

**Input File** stdin

**Output File** stdout

C\*++ language is quite similar to C++. The similarity manifests itself in the fact that the programs written in C\*++ sometimes behave unpredictably and lead to absolutely unexpected effects. For example, let's imagine an arithmetic expression in C\*++ that looks like this (*expression* is the main term):

- *expression ::= summand | expression + summand | expression - summand*
- *summand ::= increment | coefficient\*increment*
- *increment ::= a++ | ++a*
- *coefficient ::= 0|1|2|...|1000*

For example, "5\*a++ - 3\*++a+a++" is a valid expression in C\*++.

Thus, we have a sum consisting of several summands divided by signs "+" or "-". Every summand is an expression "a++" or "++a" multiplied by some integer coefficient. If the coefficient is omitted, it is suggested being equal to 1.

The calculation of such sum in C\*++ goes the following way. First all the summands are calculated one after another, then they are summed by the usual arithmetic rules. If the summand contains "a++", then during the calculation first the value of the "a" variable is multiplied by the coefficient, then value of "a" is increased by 1. If the summand contains "++a", then the actions on it are performed in the reverse order: first "a" is increased by 1, then — multiplied by the coefficient.

The summands may be calculated in any order, that's why sometimes the result of the calculation is completely unpredictable! Your task is to find its largest possible value.

## Input

The first input line contains an integer  $a$  ( $-1000 \leq a \leq 1000$ ) — the initial value of the variable "a". The next line contains an expression in C\*++ language of the described

type. The number of the summands in the expression does not exceed 1000. It is guaranteed that the line describing the expression contains no spaces and tabulation.

## Output

Output a single number — the maximal possible value of the expression.

## Examples

Input	Output
1 5*a++ - 3*++a+a++	11

Input	Output
3 a+++++a	8

## Note

Consider the second example. Initially  $a = 3$ . Suppose that at first the first summand is calculated, and then the second one is. The first summand gets equal to 3, and the value of  $a$  is increased by 1. At the calculation of the second summand  $a$  is increased once more (gets equal to 5). The value of the second summand is 5, and together they give 8. If we calculate the second summand first and the first summand later, then the both summands equals to 4, and the result is 8, too.

## Problem L. Elevator

**Time Limit** 1000 ms

**Mem Limit** 262144 kB

**Input File** input.txt

**Output File** output.txt

A sky scraper with 1000 floors has been built in the city of N. It has modern superfast elevators to help to travel from one floor to another. Each elevator has two doors, the front one and the back one. If one goes in through the front door, he goes out through the back one and vice versa. The elevator has two rails numbered with numbers 1 and 2. Rail 1 is located to the left of the entrance to the front door (or correspondingly, to the right of the entrance to the back door). Rail 2 is located opposite it, to the right of the entrance to the front door and to the left of the entrance to the back door. We know that each person in the city of N holds at a rail with the strongest hand.

One day a VIP person visited the city and of course, he took a look at the skyscraper and took a ride in the elevator. We know the door through which he entered and the rail he was holding at. Now we need to determine as soon as possible whether he is left-handed or right-handed.

### Input

The first line indicates the door through which the very important person entered the elevator. It contains "front" if the person enters the elevator through the front door and "back" if he entered the elevator through the back door. The second line contains integer  $a$  ( $1 \leq a \leq 2$ ) which denotes the number of the rail at which the person was holding.

### Output

Print character "R" if the VIP is right-handed or "L" if he is left-handed.

### Examples

Input	Output
front 1	L

## Problem M. Cinema

**Time Limit** 1500 ms

**Mem Limit** 262144 kB

**Input File** stdin

**Output File** stdout

The capital of Berland has the only movie theater in the country. Besides, it consists of only one room. The room is divided into  $n$  rows, each row consists of  $m$  seats.

There are  $k$  people lined up to the box office, each person wants to buy exactly one ticket for his own entertainment. Before the box office started selling tickets, each person found the seat that seemed best for him and remembered it as a pair of coordinates  $(x_i, y_i)$ , where  $x_i$  is the row number, and  $y_i$  is the seat number in this row.

It is possible that some people have chosen the same place, then when some people see their favorite seat taken in the plan of empty seats in the theater, they choose and buy a ticket to another place. Each of them has the following logic: let's assume that he originally wanted to buy a ticket to seat  $(x_1, y_1)$ , then when he comes to the box office, he chooses such empty seat  $(x_2, y_2)$ , which satisfies the following conditions:

- the value of  $|x_1 - x_2| + |y_1 - y_2|$  is minimum
- if the choice is not unique, then among the seats that satisfy the first condition, this person selects the one for which the value of  $x_2$  is minimum
- if the choice is still not unique, among the seats that satisfy the first and second conditions, this person selects the one for which the value of  $y_2$  is minimum

Your task is to find the coordinates of a seat for each person.

### Input

The first input line contains three integers  $n, m, k$  ( $1 \leq n, m \leq 2000$ ,  $1 \leq k \leq \min(n \cdot m, 10^5)$ ) — the number of rows in the room, the number of seats in each row and the number of people in the line, correspondingly. Each of the next  $k$  lines contains two integers  $x_i, y_i$  ( $1 \leq x_i \leq n, 1 \leq y_i \leq m$ ) — the coordinates of the seat each person has chosen. Numbers on the same line are separated by a space. The pairs of coordinates are located in the order, in which people stand in the line, starting from the

head (the first person in the line who stands in front of the box office) to the tail (the last person in the line).

## Output

Print  $k$  lines, each containing a pair of integers. Print on the  $i$ -th line  $x_i, y_i$  — the coordinates of the seat, for which the person who stands  $i$ -th in the line will buy the ticket.

## Examples

Input	Output
3 4 6 1 1 1 1 1 1 1 2 1 3 1 3	1 1 1 2 2 1 1 3 1 4 2 3

Input	Output
4 3 12 2 2 2 2	2 2 1 2 2 1 2 3 3 2 1 1 1 3 3 1 3 3 4 2 4 1 4 3