Problems Overview

Problem A: Soroban

Problem B: Vasya and Public Transport

Problem C: Good Number

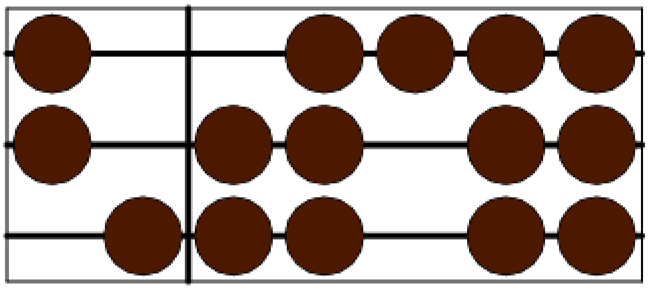
Problem D: Bear and Strings

Problem E: Vasya and Robot

Note: The input and output for all the problems are standard input and output.

Problem A: Soroban

You know that Japan is the country with almost the largest 'electronic devices per person' ratio. So you might be quite surprised to find out that the primary school in Japan teaches to count using a *Soroban* — an abacus developed in Japan. This phenomenon has its reasons, of course, but we are not going to speak about them. Let's have a look at the Soroban's construction.



Soroban consists of some number of rods, each rod contains five beads. We will assume that the rods are horizontal lines. One bead on each rod (the leftmost one) is divided from the others by a bar (the reckoning bar). This single bead is called *go-dama* and four others are *ichi-damas*. Each rod is responsible for representing a single digit from 0 to 9. We can obtain the value of a digit by following simple algorithm:

- Set the value of a digit equal to 0.
- If the go-dama is shifted to the right, add 5.
- Add the number of ichi-damas shifted to the left.

Thus, the upper rod on the picture shows digit 0, the middle one shows digit 2 and the lower one shows 7. We will consider the top rod to represent the last decimal digit of a number, so the picture shows number 720.

Write the program that prints the way Soroban shows the given number n.

Input

There are several test cases. Each test case has a line contains a single integer n ($0 \le n < 109$).

Output

For each test case: Print the description of the decimal digits of number n **from the last one to the first one (as mentioned on the picture in the statement)**, one per line. Print the beads as large English letters 'O', rod pieces as character '-' and the reckoning bar as '|'. Print as many rods, as many digits are in the decimal representation of number n without leading zeroes. We can assume that number 0 has no leading zeroes.

Sample test(s)

Input

```
2
13
720
```

Output

```
O-|OO-OO
O-|OOO-O
O-|O-OOO
O-|-OOOO
O-|OO-OO
-O|OO-OO
```

Problem B: Vasya and Public Transport

Vasya often uses public transport. The transport in the city is of two types: trolleys and buses. The city has n buses and m trolleys, the buses are numbered by integers from 1 to n, the trolleys are numbered by integers from 1 to m.

Public transport is not free. There are 4 types of tickets:

- A ticket for one ride on some bus or trolley. It costs $c1$ burles;
- A ticket for an unlimited number of rides on some bus or on some trolley. It costs $c2$ burles;
- A ticket for an unlimited number of rides on all buses or all trolleys. It costs $c3$ burles;
- A ticket for an unlimited number of rides on all buses and trolleys. It costs $c4$ burles.

Vasya knows for sure the number of rides he is going to make and the transport he is going to use. He asked you for help to find the minimum sum of burles he will have to spend on the tickets.

Input

There are several test cases. Each test case has the format:

The first line contains four integers $c1, c2, c3, c4$ $(1 \le c1, c2, c3, c4 \le 1000)$ — the costs of the tickets.

The second line contains two integers n and m $(1 \le n, m \le 1000)$ — the number of buses and trolleys Vasya is going to use.

The third line contains n integers ai $(0 \le ai \le 1000)$ — the number of times Vasya is going to use the bus number i.

The fourth line contains m integers bi $(0 \le bi \le 1000)$ — the number of times Vasya is going to use the trolley number i.

Output

For each test case print a single number — the minimum sum of burles Vasya will have to spend on the tickets.

Sample test(s)

Input

```
1 3 7 19
2 3
2 5
4 4 4
4 3 2 1
1 3
798
1 2 3
100 100 8 100
3 5
7 94 12
100 1 47 0 42
```

Output

```
12
1
16
```

Note

In the first sample the profitable strategy is to buy two tickets of the first type (for the first bus), one ticket of the second type (for the second bus) and one ticket of the third type (for all trolleys). It totals to $(2 \cdot 1) + 3 + 7 = 12$ burles.

In the second sample the profitable strategy is to buy one ticket of the fourth type.

In the third sample the profitable strategy is to buy two tickets of the third type: for all buses and for all trolleys.

Problem C: Good Number

Let's call a number k-good if it contains all digits not exceeding k (0, ..., k). You've got a number k and an array a containing $n$numbers. Find out how many k-good numbers are in a (count each number every time it occurs in array a).

Input

There are several test cases. Each test case has format: The first line contains integers n and k ($1 \le n \le 100$, $0 \le k \le 9$). The i-th of the following n lines contains integer ai without leading zeroes ($1 \le ai \le 109$).

Output

For each test case print a single integer — the number of k-good numbers in a.

Sample test(s)

Input

```
10 6
1234560
1234560
1234560
1234560
1234560
1234560
1234560
1234560
1234560
1234560
2 1
1
10
```

Output

```
10
1
```

Problem D: Bear and Strings

The bear has a string $s = s1s2... s|s|$ (record $|s|$ is the string's length), consisting of lowercase English letters. The bear wants to count the number of such pairs of indices i, j $(1 \leq i \leq j \leq |s|)$, that string $x(i, j) = sisi + 1... sj$ contains at least one string "bear" as a substring.

String $x(i, j)$ contains string "bear", if there is such index k $(i \leq k \leq j - 3)$, that $sk = b$, $sk + 1 = e$, $sk + 2 = a$, $sk + 3 = r$.

Help the bear cope with the given problem.

Input

There are several test cases. Each test case has the format: The first line contains a non-empty string s $(1 \leq |s| \leq 5000)$. It is guaranteed that the string only consists of lowercase English letters.

Output

For each test case: Print a single number — the answer to the problem.

Sample test(s)

Input

```
bearbtear
bearaabearc
```

Output

```
6
20
```

Note

In the first sample, the following pairs (i, j) match: (1, 4), (1, 5), (1, 6), (1, 7), (1, 8), (1, 9).

In the second sample, the following pairs (i, j) match: (1, 4), (1, 5), (1, 6), (1, 7), (1, 8), (1, 9), (1, 10), (1, 11), (2, 10), (2, 11), (3, 10), (3, 11), (4, 10), (4, 11), (5, 10), (5, 11), (6, 10), (6, 11), (7, 10), (7, 11).

Problem E: Vasya and Robot

Vasya has n items lying in a line. The items are consecutively numbered by numbers from 1 to n in such a way that the leftmost item has number 1, the rightmost item has number n. Each item has a weight, the i-th item weights wi kilograms.

Vasya needs to collect all these items, however he won't do it by himself. He uses his brand new robot. The robot has two different arms — the left one and the right one. The robot can consecutively perform the following actions:

- Take the leftmost item with the left hand and spend $wi \cdot l$ energy units (wi is a weight of the leftmost item, l is some parameter). If the previous action was the same (left-hand), then the robot spends extra Ql energy units;
- Take the rightmost item with the right hand and spend $wj \cdot r$ energy units (wj is a weight of the rightmost item, r is some parameter). If the previous action was the same (right-hand), then the robot spends extra Qr energy units;

Naturally, Vasya wants to program the robot in a way that the robot spends as little energy as possible. He asked you to solve this problem. Your task is to find the minimum number of energy units robot spends to collect all items.

Input

There are several test cases. Each test case has the format:

The first line contains five integers n, l, r, Ql, Qr ($1 \le n \le 105$; $1 \le l, r \le 100$; $1 \le Ql, Qr \le 104$).

The second line contains n integers $w1, w2, ..., wn$ ($1 \le wi \le 100$).

Output

For each test case: In the single line print a single number — the answer to the problem.

Sample test(s)

Input

```
3 4 4 19 1
42 3 99
4 7 2 3 9
1 2 3 4
```

Output

```
576
34
```

Note

Consider the first sample. As $l = r$, we can take an item in turns: first from the left side, then from the right one and last item from the left. In total the robot spends $4·42 + 4·99 + 4·3 = 576$ energy units.

The second sample. The optimal solution is to take one item from the right, then one item from the left and two items from the right. In total the robot spends $(2·4) + (7·1) + (2·3) + (2·2 + 9) = 34$ energy units.