TASK	CESTA	PŠENICA	PRIPREME	MRAVI	SABOR	STANOVI
input	standard input					
output	standard output					
time limit	1 second	1 second	2 seconds	1 second	1 second	1.5 second
memory limit	32 MB	32 MB	32 MB	32 MB	64 MB	64 MB
score	**50**	**80**	**100**	**120**	**140**	**160**
	total 650					

One morning, completely by chance, Mirko found a positive integer N in the middle of the street. Since Mirko adores the number 30, he wants to know the maximum multiple of the number 30 that can be obtained by *shuffling* the digits of the number he found in the street.

Help our hero and write a programme that calculates that number (if it exists).

INPUT

The first and only line of input contains the integer N, consisting of at most 10^5 digits.

OUTPUT

The first and only line of output must contain the required number from the task, if it exists. If it doesn't exist, output -1.

SAMPLE TEST DATA

input	**input**	**input**
30	102	2931
output	**output**	**output**
30	210	-1

Our heroes, Mirko and Slavko, plant Christmas wheat every year on Saint Lucy's Day. It is well known that stalks of wheat grow at different speeds so, after a certain time, the wheat becomes quite messy. The guys are determined to solve this problem by playing the following game:

- When it's Mirko's turn, he chooses a stalk of wheat with the *minimal height* and prolongs its height so it's of the same height as the *first stalk longer than it*.
- When it's Slavko's turn, he chooses a stalk of wheat with the *maximal height* and cuts it to be of the same height as the *first stalk shorter than it*.
- The game lasts while there are *at least three* stalks of different heights and the loser is the player who can't make his move.

For given heights of all stalks of wheat and the assumption that Mirko is the one starting the game, determine the winner of the game and the height of the shortest and longest stalk when the game is finished.

INPUT

The first line of input contains the integer N ($1 \leqslant N \leqslant 10^5$), the number of wheat stalks.
The second line of input contains N space separated integers that denote the heights of individual wheat stalks. The height of each stalk will be less than or equal to 10^5.

OUTPUT

The first line of output must contain the word "Mirko" if Mirko is the winner of the game, or "Slavko" if Slavko is the winner of the game.
The second line of output must contain the height of the shortest and longest stalk when the game is finished.

SCORING

In test cases worth 50% of total points, it will hold $N \leqslant 500$.
In test cases worth 80% of total points, it will hold $N \leqslant 3000$.

SAMPLE TEST DATA

input	**input**	**input**
3 3 3 3	4 3 1 2 1	7 2 1 3 3 5 4 1
output	**output**	**output**
Slavko 3 3	Slavko 1 2	Slavko 2 3

Clarification of the first example: Mirko can't make his move because there are no three stalks of different heights. Therefore, Slavko is the winner.

Ante and Goran are preparing N teams of young ambitious students for a student programming competition of students from University of Zagreb. Both of them have one algorithm they *have to explain to each team*. Of course, both can't work with the same team at the same time and none of them can work with multiple teams simultaneously.

You are given the time it takes for each team to understand and implement an algorithm. Each algorithm lecture has to be done without interruption. Determine the minimal time required for Ante and Goran to do their lectures!

For additional explanation, consult the sample tests clarifications.

INPUT

The first line of input contains the integer N, the number of teams.
The following line contains N space separated integers, where the i^{th} integer denotes the time required for the i^{th} team to understand and implement an algorithm.
All numbers in the input will belong to the interval $[1, 3 \cdot 10^5]$.

OUTPUT

The first and only line of output must contain the required number from the task.

SCORING

In test cases worth 40% of total points, it will hold $N \leqslant 7$.

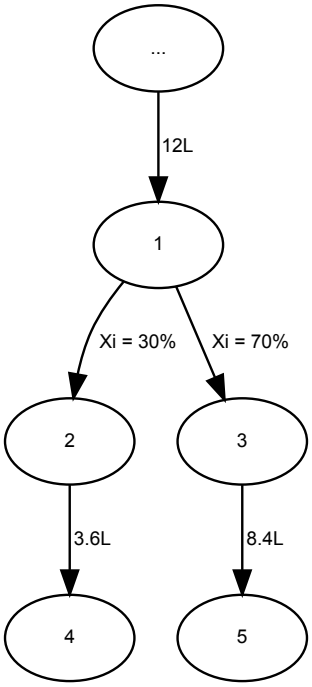
SAMPLE TEST DATA

input	**input**	**input**
3 2 2 2	3 4 1 2	4 1 3 2 1
output	**output**	**output**
6	8	7

Clarification of the first example: Each team needs 2 units of time to understand and implement an algorithm. One of the possible schedules is that Ante gives his lectures to team 1, team 2 and team 3, respectively, and Goran to team 3, team 1 and team 2, respectively.
Clarification of the second example: One of the optimal schedules is that Ante gives lectures to team 2, team 3 and team 1, respectively, but with a pause lasting 1 unit of time between team 3 and team 1. Goran will give lectures to team 1, team 3 and team 2, respectively.

Little Bobi gets up every morning and feeds his favourite pets: ants. He keeps them in a terrarium with a pipe system that can be represented as a tree with N nodes. The pipes are represented by the edges of the tree. The root of the tree is located at node denoted with 1. Inside the pipe system, the liquid flows *from a node to its children* because of gravity.

We know the flow X_i of each pipe: the percent of fluid from the parent node that flows through that pipe to the child node. Let's observe the following example:



Node 1 from the image has 12 liters of liquid in it and has two pipes after it. One has the flow of $X_i = 30$, and the other $X_i = 70$. Node 2 is going to get 3.6 liters and node 3 gets 8.4 liters. In the input data, the sum of flows of pipes going from the same node will always be equal to 100.

Some of Bobi's pipes aren't just regular pipes; they are a bit strange. They are super pipes that have the superpower to *squaretheamount* of liquid flowing through them. In the previous example, if the first pipe has the superpower, node 2 gets 12.96 liters and node 3 still gets only 8.4 liters. Notice now that a node has more liquid leaving it than the amount entering it. This is exactly the reason why these pipes are super pipes!

All super pipes can have their superpower turned on or off by Bobi.

The ants live only in the leaves of the tree (nodes that don't have any children). For each leaf we know the required amount of liquid K_i to feed all ants living in that leaf. Bobi wants to feed his ants by pouring L liters of liquid into the root of the tree. He doesn't have much money so he wants to know the minimum amount of liters of liquid he needs to buy to keep all of his ants fed.

Please note: The input data is such that the required number L will not exceed $2 \cdot 10^9$.

INPUT

The first line of input contains the integer N ($1 \leqslant N \leqslant 1000$).

Each of the following $N-1$ lines contain four integers A_i, B_i, X_i, T_i ($1 \leqslant A_i, B_i \leqslant N$, $1 \leqslant X_i \leqslant 100$, $0 \leqslant Ti \leqslant 1$) where A_i and B_i are the ends of a pipe (the labels of nodes connected by the pipe), X_i is the flow of the liquid through the pipe, and T_i denotes whether the pipe has a superpower. If T_i is 1, that pipe has a superpower, otherwise it does not.
The following line contains N integers K_i describing the amount of liquid needed for the ants in the

i^{th} node. If the i^{th} node is not a leaf, K_i will be -1, otherwise it will be an integer from the interval $[1, 10]$.

OUTPUT

The first and only line of output must contain the required number from the task.
Please note: The allowed absolute error from the correct (precise) solution is 0.001.

SAMPLE TEST DATA

input	**input**	**input**
5 1 2 50 0 1 3 50 0 2 4 25 0 2 5 75 1 -1 -1 4 1 9	3 1 2 20 1 1 3 80 1 -1 4 8	6 1 2 100 1 2 3 20 0 2 4 20 0 2 5 60 0 4 6 100 1 -1 -1 1 -1 1 2
output	**output**	**output**
8.00	10.0000	2.659

Clarification of the first example: If Bobi pours 8 liters of liquid into the root of the tree, node 3 will get 4 liters, node 4 will get 1 liter and node 5 will get 9 liters. These nodes are leaves (they have ants in them) and this is the exact minimum amount the ants need to get. Also, 8 liters is the minimum amount of liquid that satisfies the "ant" conditions.

A land far, far away has N Members of Parliament (MP). They had a turbulent and passionate debate on the law on amendments to the law on a new referendum on referendums. From Monday to Friday, all MPs joyfully came to work and argued all day.

A diligent news reporter photographed MPs at their workplace in the heat of the argument every working day of the week. What she captured on the photos are pairs of MPs fighting and scowling at each other. The five photographs have been forwarded to you for thorough analysis.

It is a fact that each MP belongs to one of the *two* political parties. Let's denote them with letters A and B. Your task is to estimate which MP belongs to which party, so that the following holds for your estimation: each MP argued with **at most two distinct members** of their own party.

INPUT

The first line of input contains an integer N ($2 \leqslant N \leqslant 200\,000$), the number of MPs. MPs are denoted with numbers from 1 to N.

The following five lines describe the photographs taken from Monday to Friday. Each of the five lines contains the list of pairs of MPs that are arguing on the photograph that day (scowling at each other). Stated first is the number of pairs P ($1 \leqslant P \leqslant N/2$), followed by P pairs in the form of "K L", where K and L are labels of MPs scowling at each other. Before each pair there is a double space.
Of course, each MP is stated at most once per line.

OUTPUT

The first and only line of output must contain an array consisting of only characters A and B, so that the K^{th} character denotes the party of K^{th} MP in a division that satisfies the given conditions.

Since the solution isn't going to be unique, output any.

SCORING

In test cases worth 30% of total points, it will hold $N \leqslant 15$.

SAMPLE TEST DATA

input

```
7
2  1 2  7 3
2  1 3  7 4
2  1 4  7 5
2  1 5  7 6
2  1 6  7 2
```

output

```
ABBBBBA
```

input

```
10
3  1 2  7 3  9 4
3  1 3  7 4  9 5
3  1 4  7 5  9 6
3  1 5  7 6  9 8
3  1 6  7 8  9 10
```

output

```
ABBBBBAAAA
```

Stanko is working as an architect in a construction company. His current task is to make a ground plan for a residential building located in Zagreb. He must determine a way to split the floor building with walls to make apartments in the shape of a rectangle. Each built wall must be parallel to the building's sides.

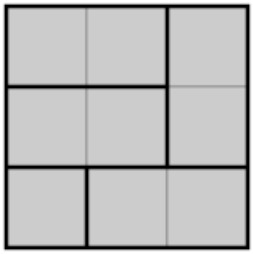
More precisely, the floor is represented in the ground plan as a large rectangle with dimensions $N \times M$, where each apartment is a smaller rectangle with dimensions $a \times b$ located inside of a larger one. The numbers a and b must be integers.

Additionally, the floor must be completely covered with apartments – each point in the floor must be located in an apartment. The apartments must not intersect, but they can touch.

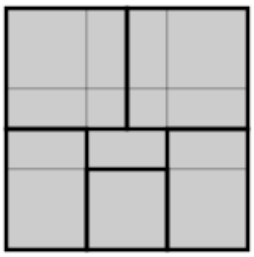
To prevent darkness indoors, the apartments must have windows. Therefore, each apartment must share its side with the edge of the rectangle representing the floor so it is possible to place a window.

Moreover, all apartments must have the approximately equal area K. The *deviation* of an area of an apartment with dimensions $a \times b$ is defined as $(a \cdot b - K)^2$. The deviation of a ground plan is the sum of all the apartment's deviations.

Stanko wants to build the best building he can, a building with minimal deviation. Help him and write a programme to determine the minimal possible deviation of a ground plan which satisfies the conditions from the task.



(a) A valid arrangement of apartments corresponding to the first example.



(b) An invalid arrangement of apartments. The length of apartments' sides are not integers and there is an apartment without windows.

INPUT

The first and only line of input contains the integers N, M, K ($1 \leqslant N, M \leqslant 300$, $1 \leqslant K \leqslant 10\,000$).

OUTPUT

The first and only line of output must contain the minimal possible deviation of the arrangement of apartments.

SAMPLE TEST DATA

input	**input**	**input**
3 3 2	2 2 2	2 3 4
output	**output**	**output**
1	0	2

Clarification of the first example: The example corresponds to the left image from the task. Notice that it is impossible to achieve a total deviation of 0.