Problem A. Vera and ABCDE

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	512 megabytes

When Vera was learning English, she had 5 types of toy blocks, one for each of the first 5 uppercase letters. A block can be represented as a 5×3 grid with characters `.` and `*`. The 5 types of blocks look like the following:

```
*** *** *** *** ***
*.* *.* *.. *.* *..
*** *** *.. *.* ***
*.* *.* *.. *.* *..
*.* *** *** *** ***
```

Vera has a word S with N letters and wonders what the corresponding sequence of blocks look like when arranged in a row.

Input

Line 1 contains integer N ($1 \leq N \leq 2017$).

Line 2 contains string S of length N, and consists of only letters `A`, `B`, `C`, `D`, `E`.

Output

Print 5 lines, each with $3N$ characters, the corresponding sequence of blocks.

Examples

standard input	standard output
5 ABCDE	*************** *.**.**..*.**.. *******..*.**** *.**.**..*.**.. *.************
10 ECADBECADB	****************************** *..*..*.**.**.**..*..*.**.**.* ****..****.********..****.**** *..*..*.**.**.**..*..*.**.**.* *******.*************.*******

Problem B. Vera and Banquet

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	512 megabytes

Vera knows 26 recipes, each represented by a lowercase letter from `a` to `z`. She is preparing a banquet consisting of N dishes arranged around a circular table. Since Vera is lazy, each dish is independently and uniformly randomly chosen from one of her 26 recipes. The banquet can be represented as a string S of length N where S_i is the recipe of dish i ($1 \leq i \leq N$). Dish j is clockwise to dish $j-1$ for $2 \leq j \leq N$ and dish 1 is clockwise to dish N.

A *sample* is the sequence of recipes in a consecutive section of dishes in either clockwise or counter-clockwise direction.

Vera wonders how many different samples there are. Two samples are the same if they have the same length and the same recipe at every position.

Input

Line 1 contains integer N ($2 \leq N \leq 50000$).

Line 2 contains string S of N lowercase letters. It is guaranteed that S represents a banquet created by the described process.

Output

Print one line with one integer, the number of different samples.

Examples

standard input	standard output
3 aba	8
6 ondrej	66
8 waterloo	118

Note

For the first example, the eight different samples are `a`, `b`, `aa`, `ab`, `ba`, `aba`, `aab`, `baa`.

For the second example, `rejo`, `drejon` are clockwise samples and `nojer`, `dnojer` are counter-clockwise samples.

Problem C. Vera and Canada Day

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	512 megabytes

For Canada Day, Vera is going to create a spectacular laser show. She will place N lasers one at a time onto an xy-plane. The i-th laser will be placed at (x_i, y_i) and Vera will not place two lasers at the same location. Each laser will shoot two beams of light in axis-aligned perpendicular directions (up-left, left-down, down-right or right-up). If a beam hits another laser j, the show will gain awe value v_j, and the beam will stop. Thus a beam can hit at most one laser, the closest one in the direction the beam is shot. Beams do not interfere with each other, so they can cross or two lasers can shoot at each other. A laser can be hit by multiple beams and each hit will gain awe value.

After Vera places the i-th laser ($1 \le i \le N$), she wants to know the maximum sum of awe values over all possible rotations of already placed lasers.

Input

Line 1 contains integer N ($1 \le N \le 10^5$).

N lines follow, the i-th one contains integers x_i, y_i, and v_i ($-10^9 \le x_i, y_i \le 10^9$, $(x_i, y_i) \neq (x_j, y_j)$ if $i \neq j$, $1 \le v_i \le 10^4$).

Output

Print N lines. The i-th line should contain one integer, the maximum sum of awe value after placing lasers 1 to i.

Example

standard input	standard output

`6`
`0 0 5`
`0 2 10`
`3 0 4`
`0 1 8`
`-1 0 5`
`4 4 100`
	`0`
	`15`
	`24`
	`35`
	`41`
	`41`

Problem D. Vera and Dogs

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	512 megabytes

Vera owns N doghouses numbered from 1 to N and $M = X \cdot N$ dogs numbered from 1 to M. Each doghouse should be the *primary home* of X dogs $P_{i,1}, \ldots, P_{i,X}$ and the *secondary home* of X dogs $S_{i,1}, \ldots, S_{i,X}$. Each dog should have one primary home and one secondary home different from its primary home. Every night, at most one doghouse might be unavailable due to cleaning. Each dog will sleep in its primary home if it is available, otherwise it will sleep in its secondary home. Each doghouse should contain at most $X + 1$ sleeping dogs on any night.

Help Vera find a valid assignment of doghouses to dogs, or determine that it is impossible.

Input

Line 1 contains integers N and X ($1 \leq N, X \leq 2017$, $X \cdot N \leq 50000$).

Output

If it is impossible to find a valid assignment, print one line with -1.

Otherwise print N lines. The i-th line should contain $2X$ integers $P_{i,1}, \ldots, P_{i,X}, S_{i,1}, \ldots, S_{i,X}$. If there are multiple possible assignments, print any of them.

Examples

standard input	standard output

3 2	5 1 6 4
	3 6 5 2
	4 2 1 3

2 2	-1

Note

For the first example:

Doghouse 1 is the primary home of dogs 5 and 1 and secondary home of dogs 6 and 4. If doghouse 1 is unavailable, then dog 1 will sleep in doghouse 3 and dog 5 will sleep in doghouse 2.

Doghouse 2 is the primary home of dogs 3 and 6 and secondary home of dogs 5 and 2. If doghouse 2 is unavailable, then dog 3 will sleep in doghouse 3 and dog 6 will sleep in doghouse 1.

Doghouse 3 is the primary home of dogs 4 and 2 and secondary home of dogs 1 and 3. If doghouse 3 is unavailable, then dog 4 will sleep in doghouse 1 and dog 2 will sleep in doghouse 2.

So it can be seen that no doghouse will ever contain more than 3 dogs.

Problem E. Vera and Engineering Buildings

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	512 megabytes

There are N engineering buildings at the University of Waterloo numbered from 1 to N. For $i \geq 2$, there is a two-way bridge from building i to building x_i. Two buildings are called *neighbours* if there is a bridge between them.

Each building has a unique aesthetic value which can be any integer. It is difficult to determine the exact aesthetic value of a building, so Vera would like to find a *nice* building, one that has higher aesthetic value than all of its neighbours. It takes Vera t_i seconds to inspect building i and the bridges to all of its neighbours. After the inspection, for each neighbour j of building i, Vera will know which of building i or j has higher aesthetic value.

What is the minimum total seconds such that it is guaranteed that Vera can find a nice building in that time no matter what the aesthetic values are? Vera can decide which building to inspect next based on the results of previous inspections. Ignore the time needed to travel between buildings.

Input

Line 1 contains integer N ($2 \leq N \leq 16$).

Line 2 contains N integers $t_1, \ldots, t_N$ ($1 \leq t_i \leq 10^8$).

Line 3 contains $N - 1$ integers $x_2, \ldots, x_N$ ($1 \leq x_i < i$).

Output

Print one line with one integer, the minimum total seconds that guarantees a nice building can be found.

Examples

standard input	standard output

3
10 40 20
1 2
	30

5
2 4 1 2 1
1 1 2 2
	5

Note

For the first example, inspecting buildings 1 and 3 guarantees a nice building will be found.

For the second example, one optimal strategy is to always inspect building 3 first.