Problem Kaguya Wants to Receive Flowers

Input data	`stdin`
Output data	`stdout`

Kaguya has never received flowers from Miyuki (*...and we need to change this ASAP!*). First, out of the deep pockets of the Shinomiya business conglomerate, Kaguya made a generous donation towards the restoration of the garden of Shuchi'in Academy, where she and Miyuki study. Then, she plans to take Miyuki to the garden under the pretence of discussing student council business. (*If he's surrounded by flowers, he will definitely get the hint and offer me a bouquet!*)

Shuchi'in Academy's garden has the shape of an N meter wide square and is divided into $N \times N$ square parcels of size 1 meter. The garden map shows that parcels are arranged neatly into rows and columns, and are denoted by (r, c) pairs, where r is the row and c is the column a parcel occupies. Some parcels, marked with 0 in the garden map, contain the ancient trees of the garden that could not be moved or cut when the garden was restored. The other parcels, marked with 1, contain flowers. We denote by F the total number of parcels that contain flowers. We also define the distance between two parcels (r, c) and (r', c') as $|r - r'| + |c - c'|$.

Kaguya defines the flowering degree of a parcel as the sum of distances from the current parcel to the closest K parcels which contain flowers. She wants to know the flowering degree of each parcel. (*If there are too many flowers around him, it will be obvious to Miyuki what I want! But if there are too few flowers, he won't get the hint...*).

Input Data

The first line of the input contains two space separated integers N and K, with the respective meanings from the statement above. The next N lines each contain N digits of 0 or 1, without any spaces between them. The j^{th} digit of line i will be 0 if parcel (i, j) doesn't contain flowers, or 1 if it does.

Output data

Output N lines, each containing N space separated integers: the j^{th} number of line i will be the flowering degree of parcel (i, j).

Restrictions

- $1 \leq N \leq 1\,000$.
- $1 \leq K \leq F \leq N \times N$.
- One of the closest K parcels that contain flowers to parcel (i, j) can be itself, if it is marked by 1 on the map.

#	Points	Restrictions
1	5	$N \leq 10$, $K = 1$, $F = 1$
2	16	$N \leq 50$
3	22	$N \leq 250$
4	12	$N \leq 650$, $K = 1$
5	10	$N \leq 650$, $F \leq 10$
6	17	$N \leq 650$
7	7	$N \leq 850$
8	11	No further constraints.

Examples

Input data	Output data
5 3 10111 10000 10000 01000 00010	3 4 3 2 3 2 5 5 5 6 3 4 6 7 8 4 5 6 6 8 7 6 7 7 9

Explanation

In this example, the garden has size $N = 5$ and we need to find, for each parcel, the sum of distances from the current parcel to the closest $K = 3$ parcels that contain flowers.

Let's consider parcel $(4, 2)$, in line 4, column 2. This parcel is marked by 1, and therefore contains flowers. The closest $K = 3$ parcels that contain flowers to parcel $(4, 2)$ are:

- $(4, 2)$ (the same parcel), at distance $|4 - 4| + |2 - 2| = 0 + 0 = 0$,
- $(3, 1)$, at distance $|4 - 3| + |2 - 1| = 1 + 1 = 2$, and
- $(5, 4)$, at distance $|4 - 5| + |2 - 4| = 1 + 2 = 3$.

The sum of these distances is $0 + 2 + 3 = 5$, and therefore the 2^{nd} number of line 4 that we output is 5.

Please note that parcel $(2, 1)$ also contains flowers and is located at distance 3 from parcel $(4, 2)$ (same as the distance to parcel $(5, 4)$), but as we had already found $K = 3$ parcels that were equally close or closer, we do not need to include parcel $(2, 1)$ in the distance computation.