Around the World

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	64 megabytes

Bobo lived in a world consisting of n cities conveniently labeled by $1, 2, \ldots, n$. Cities were connected by bidirectional roads.

Bobo would like to find a plan $(v_1, v_2, ..., v_n)$ where $v_1, v_2, \ldots, v_n$ were n different cities. He could start at city v_1 in day 1 traveling to city v_i in day i $(2 \leq i \leq n)$, and return back to city v_1 in day $(n+1)$. Bobo was lazy. So he disliked plans which make him travel more than k roads between consecutive cities.

Input

The first line contains 2 integers n, k $(4 \leq n \leq 500, 3 \leq k \leq n-1)$.

The i-th of the following n lines contains n integers $g_{i,1}, g_{i,2}, \ldots, g_{i,n}$ $(0 \leq g_{i,j} \leq 1, g_{i,j} = g_{j,i})$. If there is a road between city i and city j, then $g_{i,j} = 1$. Otherwise, $g_{i,j} = 0$.

It is guaranteed that cities were reachable from any other city.

Output

n integers $v_1, v_2, \ldots, v_n$ denotes the plan. Any feasible plan is accepted.

Examples

standard input	standard output
4 3 0100 1010 0101 0010	1 3 2 4