3 exercises – each 2 pts

6.1 Knapsack

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

Recall the knapsack problem: Given a set of n items, each with a weight w_i and a value v_i, determine the items to include in a collection so that the total weight is less than or equal to a given limit W and the total value is as large as possible. Each item can be taken at most once (i.e., there is a single copy of every item).

Input

The first line contains two integers n and W ($1 \le n \le 100$, $0 \le W \le 1\,000$), the number of items and maximal total weight of items in a knapsack.

Each of the next n lines contains two integers w_i and v_i ($1 \le w_i \le 1\,000$, $0 \le v_i \le 1\,000$), the weight and the value of the i-th item.

Output

The first line should contain an integer k ($0 \le k \le n$), the number of items in a collection.

The second line should contain k different integers from 1 to n, the indices of included items.

In case of several optimal solutions, output any of them.

Examples

standard input	standard output
2 2 2 2 1 4	1 2
5 5 1 1 2 3 3 2 5 4 4 5	2 1 5
3 100 52 99 51 50 49 50	2 2 3

6.2 Chain Matrix Multiplication

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

Given a sequence of n matrices $A_0, \ldots, A_{n-1}$ to be multiplied. The size of the matrix A_i is $m_i \times m_{i+1}$.

Find an order of multiplication minimizing the total cost of multiplication. Assume that the cost of multiplying $p \times q$ and $q \times r$ matrices is equal to $p \cdot q \cdot r$ (this is the number of integer multiplications of a straightforward matrix multiplication algorithm).

Input

The first line contains one integer n ($2 \leq n \leq 150$), the number of matrices to be multiplied.

The second line contains $(n + 1)$ integers $m_0, m_1, \ldots, m_n$ ($1 \leq m_i \leq 1\,000$), the size of the i-th matrix is $m_i \times m_{i+1}$.

Output

The minimum cost of multiplying the given n matrices.

Examples

standard input	standard output
2 10 20 10	2000
3 50 20 1 10	1500
10 1 2 3 4 5 6 5 4 3 2 1	138

6.4 Maximal Sum Square

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

Given an $n \times n$ matrix A of integers and an integer k.

Find a $k \times k$ submatrix of A with maximum sum of elements.

Input

The first line contains two integers n and k ($1 \leq k \leq n \leq 700$), dimensions of the input matrix and submatrix, respectively.

The i-th of next n lines contains n integers a_{ij} ($0 \leq a_{ij} \leq 1\,000$), the elements of the i-th row of A.

Output

The maximum sum of elements of a $k \times k$ submatrix.

Examples

standard input	standard output
3 2 10 2 10 1 1 1 10 1 13	16
2 1 4 1 1 2	4
5 3 1 2 3 4 5 5 4 3 2 1 1 1 1 1 1 2 1 2 1 2 6 3 4 1 5	21