Assume you are given an array **A** of N integers, array **ID** of $N+1$ integers from the interval $[1, N-1]$ and an integer R.

We are doing a Warshall-Turing-Fourier transformation[1] on array **A** in the following way:

```
sum = 0

for i = 1 to N
    index = min{ ID[i], ID[i+1] }
    sum = sum + A[index]
    rotate array A to the right by R places

change the signs of all elements in A

for i = 1 to N
    index = max{ ID[i], ID[i+1] }
    index = index + 1
    sum = sum + A[index]
    rotate array A to the right by R places
```

You are given the array **A** and constant R, but you are not familiar with the array **ID**. What is the largest possible value of variable `sum` after execution of the above algorithm?

INPUT

The first line of input contains the integers N and R ($2 \leqslant N \leqslant 3\,000$, $1 \leqslant R < N$) from the task.

The second line of input contains the elements of array **A**, respectively from **A**[1] to **A**[N]. These are integers from the interval $[-10^4, 10^4]$.

OUTPUT

The first line of output must contain the maximal value of `sum`.
The second line of output must contain the array **ID** of $N+1$ integers from the interval [1, N - 1] for which the algorithm outputs the maximal sum. If there are multiple such arrays, output any.

If only the first line is correct (regardless of whether the second is printed), you will get 50% of points for the corresponding test case.

SCORING

In test cases worth 20% of total points, it will hold $N \leqslant 7$.
In test cases worth 60% of total points, it will hold $N \leqslant 300$.

SAMPLE TESTS

input	input
5 3 1 -1 1 -1 1	6 5 2 5 4 1 3 5
output	**output**
10 1 1 1 2 2 3	16 3 2 1 1 5 4 1

[1] This doesn't really exist.