

CCO '99 P6 - Fast Food

Time limit: 2.0s **Memory limit:** 64M

Canadian Computing Competition: 1999 Stage 2, Day 2, Problem 3

The fast food chain, McBurger, has recently consolidated all activities to restaurants along the Trans Canada Highway. (Following the completion of fixed links joining Newfoundland and Vancouver Island to the mainland.) They have also decided to build several warehouses along the highway, each located at a restaurant and supplying those restaurants nearby. Naturally, these warehouses should be placed so as to minimize the distance between the warehouses and the restaurants they service. Your task is to write a program that determines the optimal positions for the warehouses.

To make the problem more precise, chairman McBoss has issued the following specification: You are given the positions of n restaurants across the country as n integers $d_1 < d_2 < \dots < d_n$. These integers are the distances, in kilometres, from the company's new headquarters in St. John's, at the extreme Eastern end of the highway. You are also given the number $k \leq n$, the number of warehouses to be constructed. You are to place the warehouses at the locations of k of the n restaurants so as to minimize the maximum distance from any restaurant to its nearest warehouse.

Input Specification

The input file contains several test data sets. Each data set begins with the two integers $n \leq 200$ and k . Following this will be n integers d_1, d_2, \dots, d_n . The number 0 will follow the last data set. Each integer will be on a separate line of input.

Output Specification

For each input data set, output three lines. The first line must contain k integers giving the locations of the warehouses, in kilometres from head office, in ascending order. If several sets of warehouse locations yield the same maximal distance, any one will do. The second line must contain the maximum distance from any restaurant to the nearest warehouse (this is the quantity that your program is to minimize). The third line must be blank.

Sample Input

```
6
3
5
6
12
19
20
27
0
```

Sample Output

```
6 20 27
6
```