Project 3: Hashing

Time Limit: 1 second

Given a hash table of size *N*, we can define a hash function $H(x) = x\%N$. Suppose that the linear probing is used to solve collisions, we can easily obtain the status of the hash table with a given sequence of input numbers.

However, now you are asked to solve the reversed problem: reconstruct the input sequence from the given status of the hash table. Whenever there are multiple choices, the smallest number is always taken.

Input Specification:

Your program must read test cases from the standard input.

Input consists of several test cases. For each test case, the first line contains a positive integer *N* (≤1000), which is the size of the hash table. The next line contains *N* integers, separated by a space. A negative integer represents an empty cell in the hash table. It is guaranteed that all the non-negative integers are distinct in the table.

The input ends with *N* being 0. That case must NOT be processed.

Output Specification:

For each test case, output to the standard output. Print a line that contains the input sequence, with the numbers separated by a space. Notice that there must be no extra space at the end of each line.

Sample Input:

```
11
33 1 13 12 34 38 27 22 32 -1 21
3
3 1 2
0
```

Sample Output:

```
1 13 12 21 33 34 38 27 22 32
1 2 3
```

Grading Policy:

This assignment is due Wednesday, November 4th, 2009 at 10:00pm.

- **Programmer:** Write the program **(50 pts.) with sufficient comments**.
- **Tester:** Provide a set of test cases to fill in a test report **(20 pts.)**. Note that the tester is responsible, as well as the programmer is, for any bug later found by Judge. Write analysis and comments **(10 pts.)**.
- **Report Writer:** Write Chapter 1 **(6 pts.)**, Chapter 2 **(12 pts.)**, and finally a complete report **(2 pts. for overall style of documentation)**.