E. Main Sequence

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

As you know, Vova has recently become a new shaman in the city of Ultima Thule. So, he has received the shaman knowledge about the correct bracket sequences. The shamans of Ultima Thule have been using lots of different types of brackets since prehistoric times. A bracket type is a positive integer. The shamans define a correct bracket sequence as follows:

- An empty sequence is a correct bracket sequence.
- If $\{a_1, a_2, ..., a_l\}$ and $\{b_1, b_2, ..., b_k\}$ are correct bracket sequences, then sequence $\{a_1, a_2, ..., a_l, b_1, b_2, ..., b_k\}$ (their concatenation) also is a correct bracket sequence.
- If $\{a_1, a_2, ..., a_l\}$ — is a correct bracket sequence, then sequence also is a correct bracket sequence, where v $(v > 0)$ is an integer.

For example, sequences $\{1, 1, -1, 2, -2, -1\}$ and $\{3, -3\}$ are correct bracket sequences, and $\{2, -3\}$ is not.

Moreover, after Vova became a shaman, he learned the *most important* correct bracket sequence $\{x_1, x_2, ..., x_n\}$, consisting of n integers. As sequence x is the most important, Vova decided to encrypt it just in case.

Encrypting consists of two sequences. The first sequence $\{p_1, p_2, ..., p_n\}$ contains types of brackets, that is, $p_i = |x_i|$ $(1 \le i \le n)$. The second sequence $\{q_1, q_2, ..., q_t\}$ contains t integers — **some positions** (possibly, not all of them), which had negative numbers in sequence $\{x_1, x_2, ..., x_n\}$.

Unfortunately, Vova forgot the main sequence. But he was lucky enough to keep the encryption: sequences $\{p_1, p_2, ..., p_n\}$ and $\{q_1, q_2, ..., q_t\}$. Help Vova restore sequence x by the encryption. If there are multiple sequences that correspond to the encryption, restore any of them. If there are no such sequences, you should tell so.

Input

The first line of the input contains integer n $(1 \le n \le 10^6)$. The second line contains n integers: $p_1, p_2, ..., p_n$ $(1 \le p_i \le 10^9)$.

The third line contains integer t $(0 \le t \le n)$, followed by t distinct integers $q_1, q_2, ..., q_t$ $(1 \le q_i \le n)$.

The numbers in each line are separated by spaces.

Output

Print a single string "NO" (without the quotes) if Vova is mistaken and a suitable sequence $\{x_1, x_2, ..., x_n\}$ doesn't exist.

Otherwise, in the first line print "YES" (without the quotes) and in the second line print n integers $x_1, x_2, ..., x_n$ $(|x_i| = p_i; x_{q_j} < 0)$. If there are multiple sequences that correspond to the encrypting, you are allowed to print any of them.

Examples

input

```
2
1 1
0
```

output

```
YES
1 -1
```

input

```
4
1 1 1 1
1 3
```

output

```
YES
1 1 -1 -1
```

input

```
3
1 1 1
0
```

output

```
NO
```

input

```
4
1 2 2 1
2 3 4
```

output

```
YES
1 2 -2 -1
```