Segment Processor

Dilshan is working with a new supercomputer that can run the same program multiple times simultaneously (at the same time). Dilshan has ***N*** lists of numbers, and wants to calculate the sum of each number list, using the supercomputer.

He wrote two programs to get this done. **mainProgram** calls **getSumOfSet** program ***N*** times (simultaneously).

mainProgram accepts **3** parameters.

- ***S*** - the length of a single list
- ***startingPositions*** - a list containing the starting positions of the N lists
- ***input*** - a list containing all the numbers

getSumOfSet program accepts **3** parameters.

- ***S*** - number of elements to be added from input
- ***startingPosition*** - index of the first number in the sequence
- ***Input*** - the list containing all the numbers

```
mainProgram ( int S, int startingPositions [ ],  int input [ ] ) {
        for each startingPosition in startingPositions:
                call getSumOfSet( S, startingPosition, input )
}
```

```
getSumOfSet ( int S, int startingPosition, int input[  ] ) {
        BigInt ans = 0;
        for(int i = 0; i < S; i++) {
                ans += input[ startingPosition + i];
        }
        print ans;
}
```

Notice that ***S*** is the same for all calls to the **getSumOfSet** program. Hence he needs your help to prepare the input parameters to be sent to **mainProgram**.

You are allowed to modify the original ***N*** lists of numbers in order to meet the input criteria of **mainProgram**.

Input Format

First line contains a single integer, ***N***.
Next line contains ***N*** integers, with i^{th} of them being C_i.
N lines follow, with the i^{th} line having C_i space separated integers and the j^{th} of them being $V_{i,j}$.

Constraints

- $1 \leq N \leq 10^5$
- $1 \leq C_i \leq 10^3$

- $-10^8 \leq V_{i,j} \leq 10^8$

Limits

- **Time Limit**: 1s
- **Memory Limit**: 256MB

Output Format

First line should contain a single integer, ***S***, the length of a number set.
Next line should contain ***N*** integers, the starting positions for each number set(***startingPositions***).
Third line should contain a list of integers (***N x S*** maximum length), the newly prepared list of numbers (***input***).

Note: the sequence is indexed from 0.

Sample Input 0

```
3
3 3 3
3 2 1
4 5 6
9 8 7
```

Sample Output 0

```
3
0 3 6
3 2 1 4 5 6 9 8 7
```

Sample Input 1

```
3
5 2 3
4 3 2 1 0
5 6
9 8 7
```

Sample Output 1

```
5
0 5 10
4 3 2 1 0 5 6 -5 3 2 9 8 7 -8 8
```