

L. Quantum Telemetry Alignment

Standard I/O | Time Limit: 1.0s | Memory Limit: 512MB

Deep within Sector 2026, reconnaissance operatives Alice and Bob must synchronize navigation tokens across an intermittent quantum bridge using a generalized telemetry deck. Each data token is uniquely identified by a frequency sector s ($1 \leq s \leq S$) and resonance rank r ($1 \leq r \leq R$), represented as a string $s:r$.

Your program will be executed twice within each test case to simulate the roles of both Alice and Bob via standard input (stdin) and standard output (stdout):

- Run 1 (Alice): Alice receives N distinct data tokens. She conceals exactly one chosen token in the quantum buffer and arranges the remaining $N - 1$ tokens into a specific permutation sequence to broadcast.
- Run 2 (Bob): Bob receives the ordered sequence of $N - 1$ tokens broadcast by Alice from Run 1. By analyzing the permutation arrangement, Bob must deduce and output the exact concealed token.

Input

For Run 1 (Alice):

- Line 1 contains the string "first".
- Line 2 contains three space-separated integers: N , S , and R .
- Line 3 contains N space-separated strings, representing the initial N distinct tokens given to Alice.

For Run 2 (Bob):

- Line 1 contains the string "second".
- Line 2 contains three space-separated integers: N , S , and R .
- Line 3 contains $N - 1$ space-separated strings, representing the sequence output by Alice in Run 1.

Output

For Run 1 (Alice):

- Line 1: Print the concealed token in format $s:r$.
- Line 2: Print the remaining $N - 1$ tokens in your chosen arrangement order, separated by a single space.

For Run 2 (Bob):

- Print exactly one line containing the predicted concealed token in format $s:r$.

Constraints

$4 \leq N \leq 13$, $1 \leq S \leq N - 1$, $1 \leq R \leq 2 \times (N - 2)! + 1$, $1 \leq s \leq S$ and $1 \leq r \leq R$ for all tokens. All initial N tokens are pairwise distinct.

Standard Input	Standard Output
Run 1 (Alice): first 5 4 13 1:13 2:4 3:5 1:5 2:6	Run 1 Output: 2:6 1:13 2:4 3:5 1:5
Run 2 (Bob): second 5 4 13 1:13 2:4 3:5 1:5	Run 2 Output: 2:6