# Problem

A 3-dimensional $N \times N \times N$ array has to be rotated by K-times. One rotation is a combination of two operations, a backward rotation followed by a forward rotation. Backward rotation considers elements lying in the plane formed by joining indices $[0][0][0]$, $[N-1][0][N-1]$, $[N-1][N-1][N-1]$, and $[0][N-1][0]$. Similarly, for Forward rotation elements lying in the plane formed by joining indices $[N-1][0][0]$, $[0][0][N-1]$, $[0][N-1][N-1]$, and $[N-1][N-1][0]$ are considered. The starting points for Backward and Forward rotations are $[0][0][0]$ and $[N-1][0][0]$, respectively. Rotation by $K = 1$ in a $3 \times 3 \times 3$ array changes the content as is shown in Fig. 1.
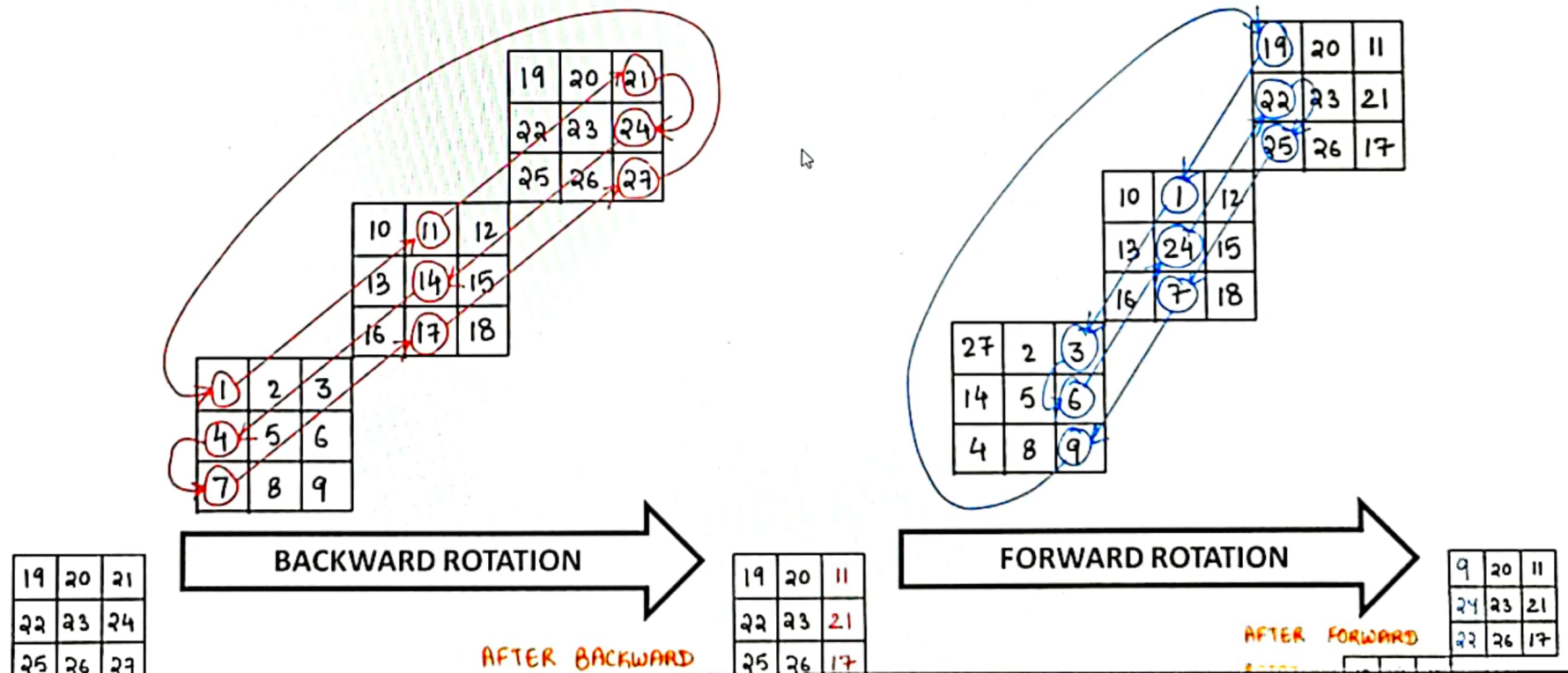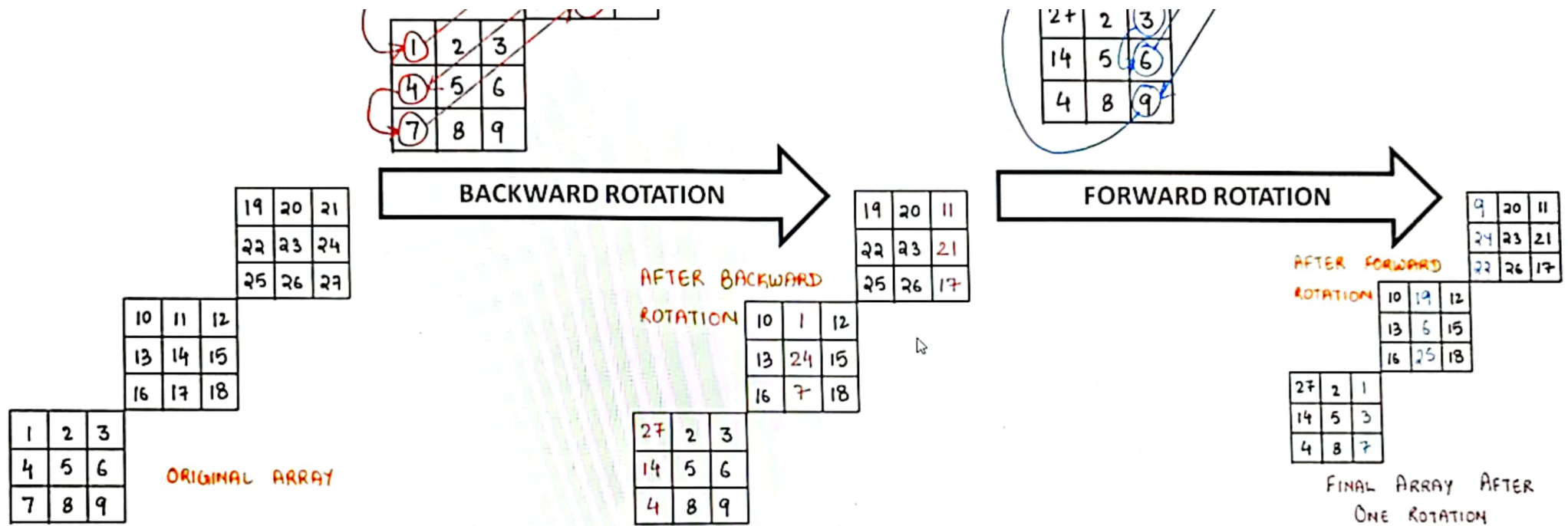
Fig. 1. One rotation (backward followed by forward) in a 3×3×3 array for k = 1.

Input Format

- Line 1 contains two space separated integers, N and K, respectively representing dimensions of a 3-dimensional array and the value of K for rotation.
- Each of the following N lines contains $N \times N$ space separated integers representing elements in each possible $N \times N$ array following a row-major order, starting with $N = 0$.

Output Format

- Line 1 is a space separated sequence of $N \times N \times N$ integers representing the final contents of a 3 dimensional array following a row major order obtained after K rotations, starting with $N = 0$

Input Format

- Line 1 contains two space separated integers, N and K, respectively representing dimensions of a 3-dimensional array and the value of K for rotation.
- Each of the following N lines contains $N \times N$ space separated integers representing elements in each possible $N \times N$ array following a row-major order, starting with $N = 0$.

Output Format

- Line 1 is a space separated sequence of $N \times N \times N$ integers representing the final contents of a 3-dimensional array following a row-major order obtained after K rotations, starting with $N = 0$.

Constraints

- All integers range in between 1 and 1000.

Sample 1:

Input

```
3 1
1 2 3 4 5 6 7 8 9
10 11 12 13 14 15 16 17 18
19 20 21 22 23 24 25 26 27
```

Output

```
27 2 1 14 5 3 4 8 7 10 19 12 13 6 15 16 25 18 9 20 11 24 23 21 22 26 17
```

Explanation:

Refer Fig. 1 for detailed explanation.

More Info