Solved Problem - Rotate Array

In this lesson, we'll see how to rotate an array.

We'll cover the following

Problem statement

Given an array, $A[]$ of length N. Rotate it clockwise by d or cyclic shift the element to the right by d. **Input format**

The first line consists of two space-separated integers N, d $(1 \leq N \leq 10^5)$.

The second line consists of N space-separated integers representing the array $A[]$ $(1 \leq A[i] \leq 10^5)$.

Sample

Input:

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

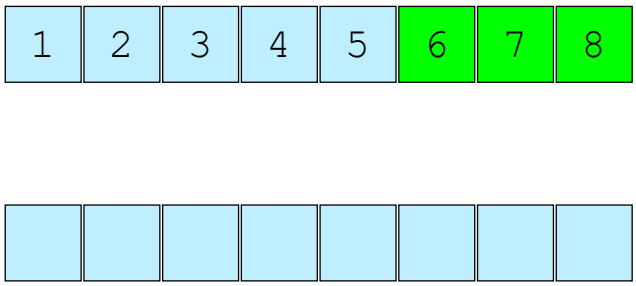
Output

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

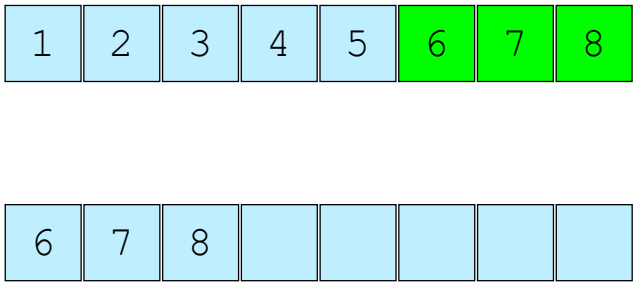
Solution

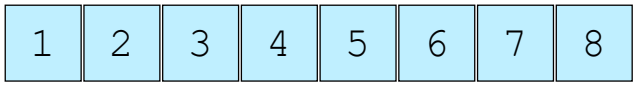
Let's see what happens when we cyclic shift by one. The last element comes to the

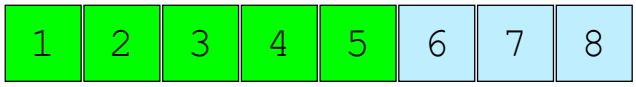
first place and each other element moves one place to the right. We can do this d times. The complexity of this solution would be $O(Nd)$.
We can optimize it further based on the observation that after d operation, the last dd elements become the first d elements and the remaining $N - d$ elements each shift to the right by dd places. We can do that in a single go.



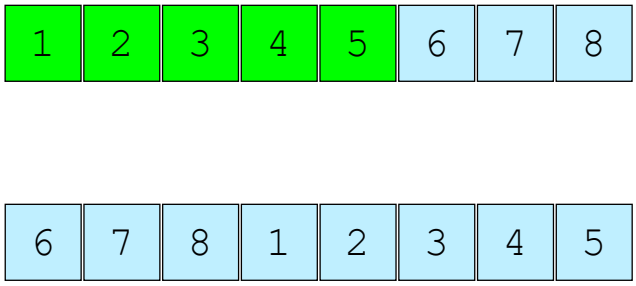
Copy last d elements







Copy remaining elements



5 of 5

main.cpp

input.txt

All code files are copied to end of the page...

Time complexity

Since we are moving each element by exactly one space, the time complexity of the solution is $O(N)$.

In the next lesson, we'll discuss how to merge two sorted arrays.

Code Files Content !!!

⤋ ⤋

```
------------------------------------------------------------------------------
|  main.cpp [1]
------------------------------------------------------------------------------



#include
#include
#include
using namespace std;

int main() {
  ifstream cin("input.txt");

  int N, d;
  cin >> N >> d;
  vector v(N);
  for (int i = 0; i < N; i++)
    cin >> v[i];

  vector res(8);

  for (int i = N - d, j = 0; i < N; i++, j++) res[j] = v[i];
  for (int i = 0, j = d; i < N - d; i++, j++) res[j] = v[i];

  for (int i = 0; i < N; i++)
    cout << res[i] << " ";

  return 0;
}


------------------------------------------------------------------------------
|  input.txt [1]
------------------------------------------------------------------------------


8 3
1 2 3 4 5 6 7 8




*******************************************************************************
```