~~Dijkstra's Algorithm~~

Juggling Algorithm

Simren Dubey
Simren.dubey@spit.ac.in

Array Rotation in Place

Suppose we are given an array of n integers and we have to rotate it by K positions to the left with with space Complexity O(1) i.e within the same array (in-place)

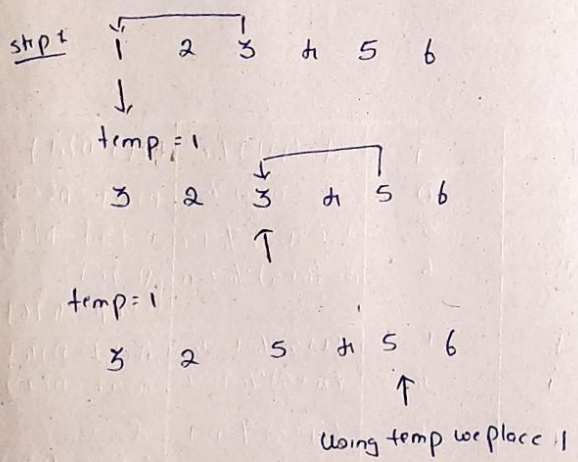
1 2 3 4 5 6 n=6

K=2

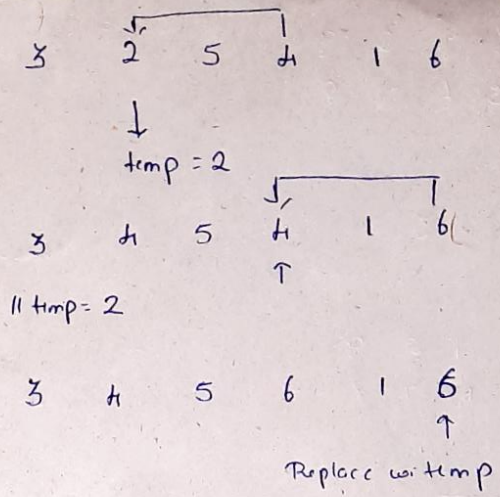
3 4 5 6 1 2

1. Divide the array into sets / cycles
2. Shift every element in a set K places to the left.
3. Then move to the next start and repeat the process

step 1



→ End of cycle 1



step 1

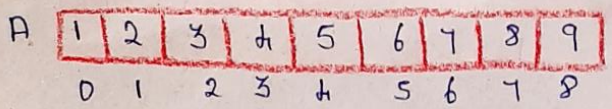
Array Rotation in-Place

Divide the array into sets & rotate each element in set K position to left

Two loops

1. Outer loop - No. of sets
2. Inner loop - rotate elements of a set K positions to the left

1. How to find the no. of sets?
   → gcd(n, k)
2. How to one by one shift the elements in a set K positions to the left?
3. When to move to the next set?



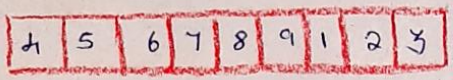
k = 3 n = 9

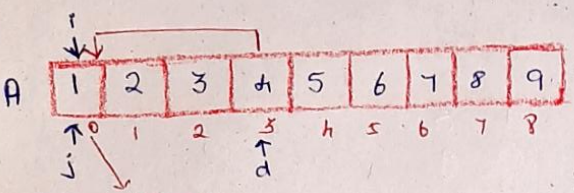
n	k	gcd
9	3	3

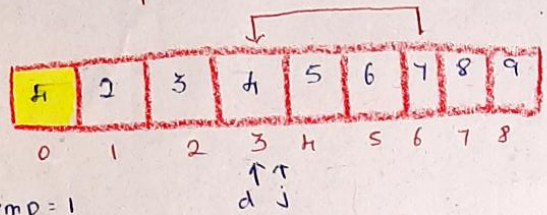
- No. of sets will depend on the value of n & k
- no. of sets = gcd(n, k)
- gcd = greatest common divisor
- Outer loop i = 0 to i < no. of sets
- A[j] = A[(j + k) % n].
  ↳ So that we move in a cyclic way.
- Inner loop ends when d == i
- Increment i to move to next step

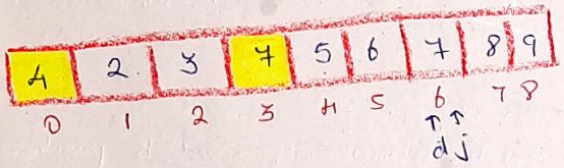
Set 1: 0, 3, 6
Set 2: 1, 4, 7
Set 3: 2, 5, 8

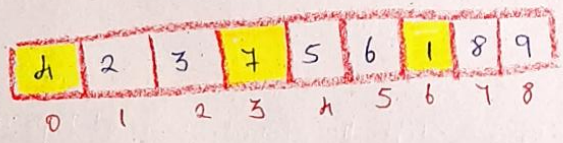
O/P - Should be











→ Like wise. Carry on.

i	j	d = (j+k) % n	A[j] = A[d]
0	0	d = (0+3) % 9 = 3	A[0] = A[3]
0	3	d = (3+3) % 9 = 6	A[3] = A[6]
0	6	d = (6+3) % 9 = 0	A[6] ↳ temp = A[6]
1	1	(1+3) % 9 = 4	A[1] = A[4]
1	4	(4+3) % 9 = 7	A[4] = A[7]
1	7	(7+3) % 9 = 1	d == i ↳ temp = A[7]
2	2	(2+3) % 9 = 5	A[2] = A[5]
2	5	(5+3) % 9 = 8	A[5] = A[8]
2	8	(8+3) % 9 = 2	d == i ↳ temp = A[8]