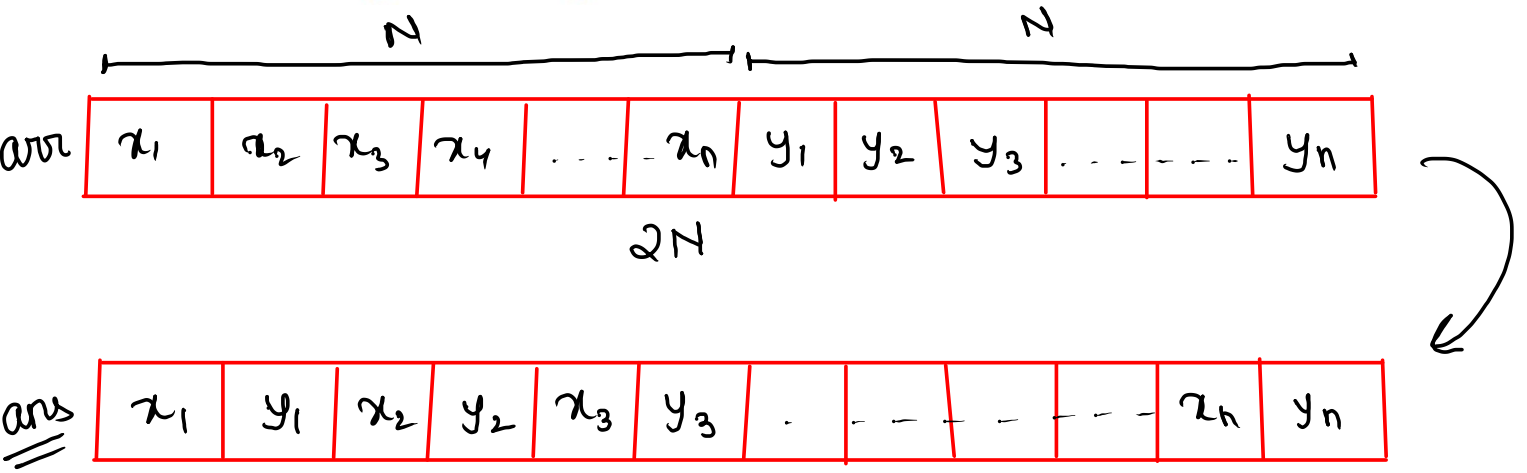
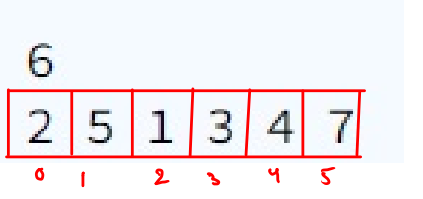
Interleaving x and y Elements

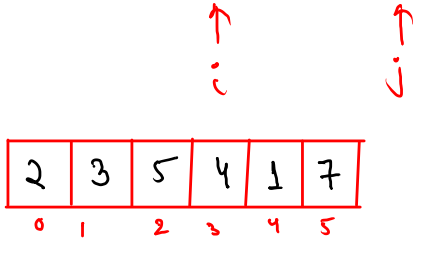


Ex



$2N = 6$

$N = 3$



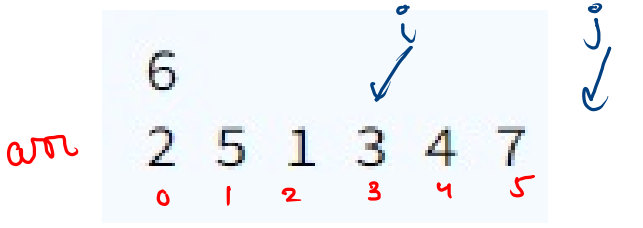
```java
public static void main(String[] args) {
    /* Enter your code here. Read input fi
    Scanner scn = new Scanner(System.in);
    int n = scn.nextInt();
    int[] arr = new int[n];
    for (int i = 0; i < n; i++) {
        arr[i] = scn.nextInt();
    }

    interleaving(arr, n);
}
```

```java
public static void interleaving(int[] arr, int n2) {
    int n = n2 / 2;
    int[] ans = new int[n2];
    int i = 0;
    int j = n;
    int k = 0;
    while ( k < ans.length ) {

        ans[k] = arr[i];
        k++;
        i++;

        ans[k] = arr[j];
        k++;
        j++;
    }

    for (int c = 0; c < ans.length; c++) {
        System.out.print(ans[c] + " ");
    }
}
```

leetcode

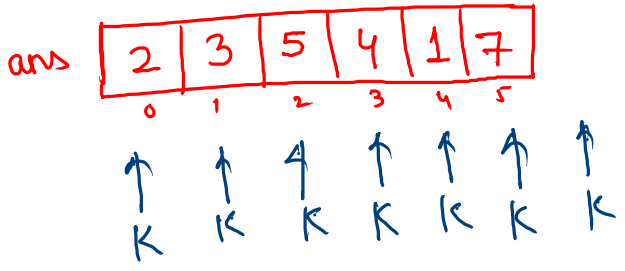
```java
public int[] shuffle(int[] arr, int n) {
    int[] ans = new int[2 * n];
    int i = 0;
    int j = n;
    int k = 0;
    while ( k < ans.length ) {
        ans[k++] = arr[i++];
        ans[k++] = arr[j++];
    }
    return ans;
}
```

Dry Run

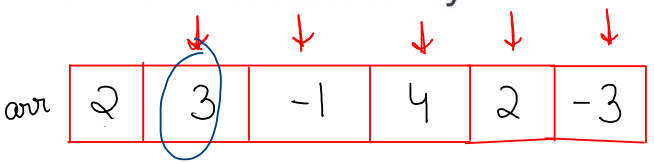


n2 = 6

n = 3



Maximum Product Subarray 2

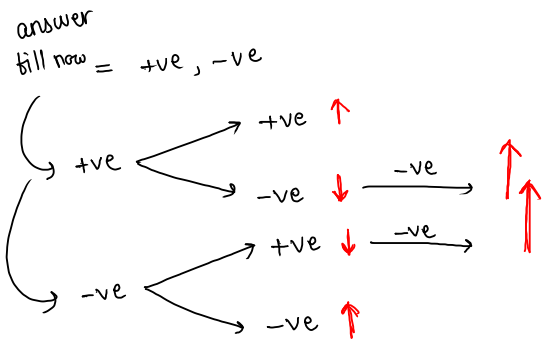


maxi = 8 x -3 = -24

mini = -48 x -3 = 144

overall Max = ~~2 6 8~~ 144

(3, 6, 6)
(-1, -6, -3)
(4, -4, -24)
(2, 8, -48)
(-3, -24, 144)

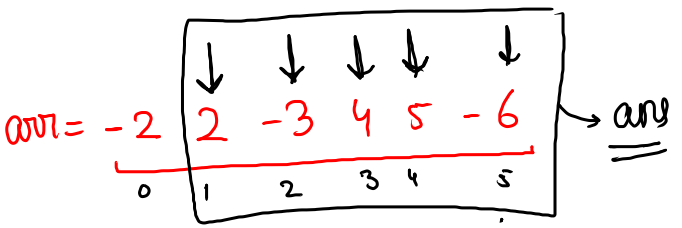


Note:- we need to store maximum value at all time

we need to store minimum value at all time

```
public static void product(int[] arr, int n) {
    int mini = arr[0];
    int maxi = arr[0];
    int overallMaxi = arr[0];

    for (int i = 1; i < n; i++) {
        int temp = maxi;
        maxi = Math.max( arr[i], Math.max( arr[i] * mini, arr[i] * maxi ) );
        mini = Math.min( arr[i], Math.min( arr[i] * mini, arr[i] * temp ) );
        overallMaxi = Math.max( overallMaxi, maxi );
    }

    System.out.println(overallMaxi);
}
```

(arr[i], arr[i] x maxi, arr[i] x mini)



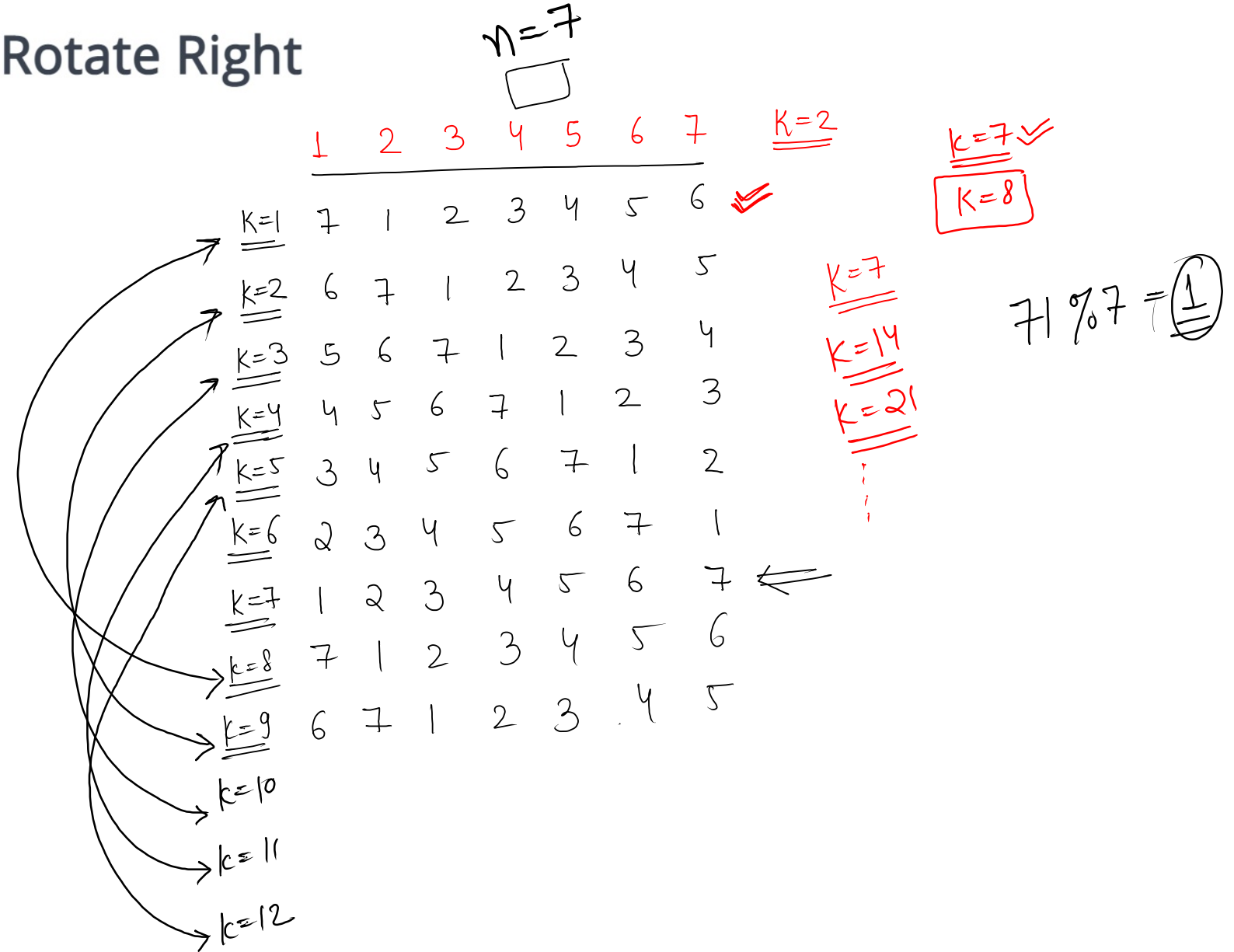
max = 720

min = -1440

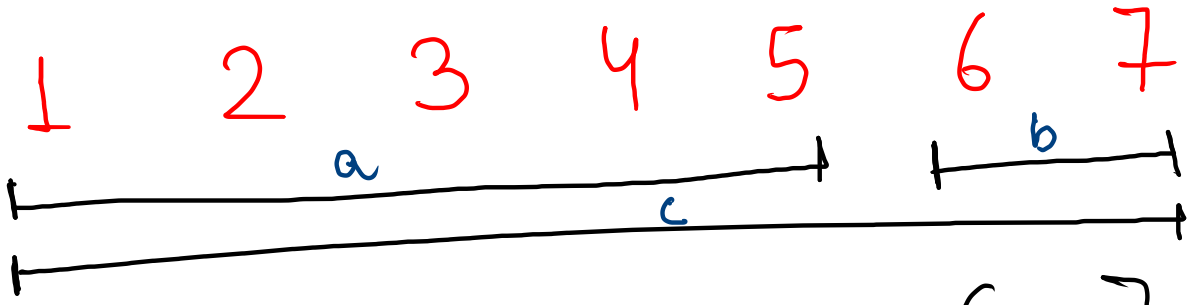
omax = ~~-2 2 12 48~~ 240 720

(2, -4, -4)
(-3, -6, 12)
(4, 48, -24)
(5, 240, -120)
(-6, -1440, 720)

Rotate Right



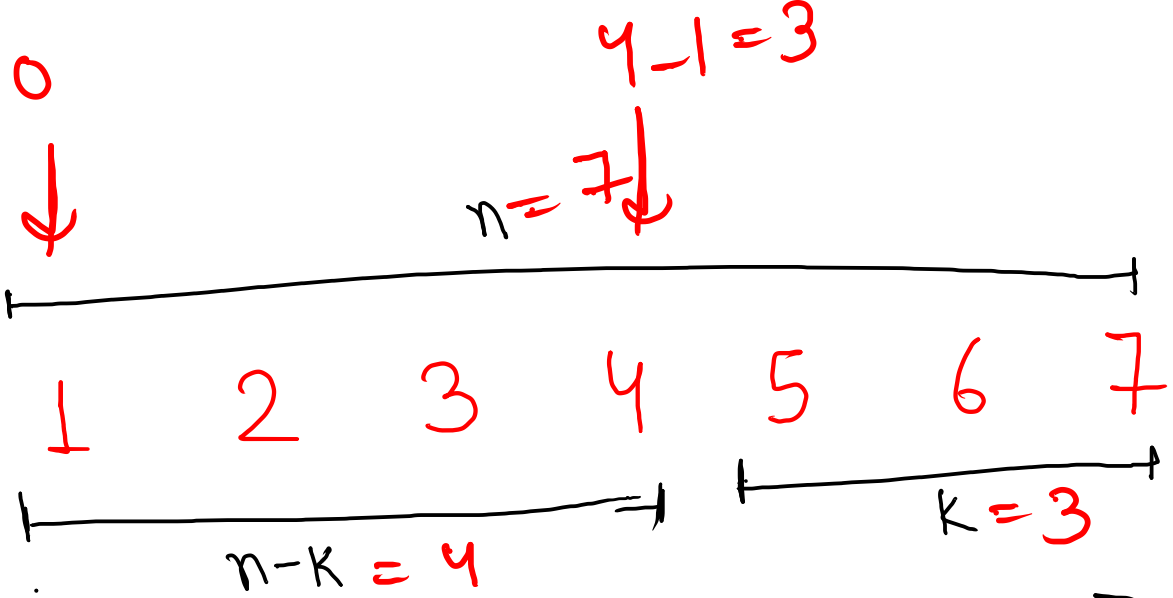
$K = 2$



step 1 5 4 3 2 1 6 7 (reverse a)

step 2 5 4 3 2 1 7 6 (reverse b)

step 3 6 7 1 2 3 4 5 (reverse c)



$k = 3$

step 1 5 4 3 2 1 6 7 (reverse a)

step 2 5 4 3 2 1 7 6 (reverse b)

step 3 6 7 1 2 3 4 5 (reverse c)