Introduction to Array

Arrays

$$\text{int } \text{arr}[5] = [\overset{0}{3}, \overset{1}{2}, \overset{2}{8}, \overset{3}{9}, \overset{4}{7}]$$

int arr [N] → 1st element → 0th index

→ last elem → N-1

Quiz arr = [5, -4, 8, 9, 10]
0 1 2 3 4

1st elem → arr[0]
5th elem → arr[4]

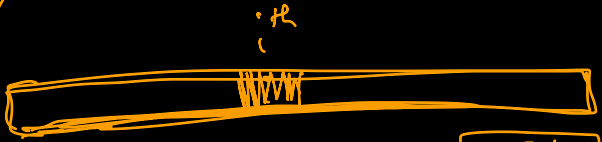
Print all the array elements.

```
for( i=0 ; i<N ; i++)
    print (arr[i])  → O(1)
```

N iterations.

T.C => O(N)

Accessing ith element ?



arr[i] => $\boxed{O(1)}$ ←

Q.1 Given array of N size, Count the number of elements which have atleast one element greater than itself. #(It's just counting all the elements other then Max element)

Example [-3, -2, 6, 8, 4, 8, 5]

N = 7 Ans = 5 .

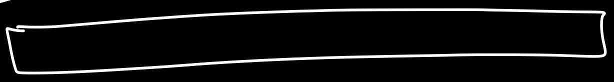
Quiz : [2, 5, 1, 4, 8, 0, 8, 1, 3, 8]

Ans = 7 .

Observation :

↳ ① Max element → this element will not be part of ans.

N elmt



Cnt max → count of max .

ans = N - Cmax   [total length - Cnt max)

Algorithm

step 1 : Find the max. no .

step 2 : Iterate & find the count of max no (Cmax)

Step 3 : return N - Cmax .

Find the max?

A : [-2, -8, -3]

```
int   max = ~~0~~ arr[0];
for ( i = 0 ; i < N ; i++ )
{
      if ( arr[i] > max )
         max = arr[i]
}
```

⇒ max = 0 → -2

INT_MIN

Find count

```
Cmax = 0;
for( i=0; i<N; i++)
{       if (A[i]== max)
             Cmax++
}
```

#

return N- Cmax;

Analysis

# of iterations

=> N+N = 2N.

T.C => O(N).

S.C => O(1)

H.W (20 mins) : Implement in a single loop. => SC => O(1)

T.C => O(N).

at end of the pdf

Q.2 Given an array of N size & a no. 'K'. Return 'true' if there exits a pair

a[i], a[j]

such that

$a[i] + a[j] = K$ & $i \neq j$

Ex $A : [\overset{0}{3}, \overset{1}{-2}, \overset{2}{1}, \overset{3}{4}, \overset{4}{3}, \overset{5}{6}, \overset{6}{8}]$

$K = 10$

$\therefore a[3] + a[5] = 4 + 6$

$= \boxed{10}$

∴ => True.

Ex 2   A : [2, 4, -3, 7] $N = 4$.

$K = 5$ $2+(-3) = -1$

=> False.

(using nested loop)

Obs : Brute force.

A : [3, -2, 1, 4, 3, 6, 8]

$K = 10$

iterations
=> N^2 $\boxed{\Rightarrow O(N^2)}$.

```
for (i=0; i<N; i++)
{    for (j=0; j<N; j++)
     {
         if (i==j) continue
         if (a[i] + a[j] == K)
             return True;
```

A : [3, 5, 2, 7] (indices 0, 1, 2, 3)

$K = 4$ 4

=> ~~A: True~~   A : False.

3

3

j

i

(0,0)	0,1	0,2	0,3
1,0	(1,1)	1,2	1,3
2,0	2,1	2,2	2,3
3,0	3,1	3,2	3,3

A : [3, 5, 2, 7] (indices 0, 1, 2, 3)

$A[1] + A[2] = 5 + 2$
$= 7$.

$A[2] + A[1] = 2 + 5$
$= 7$

just consider the upper triangle!

```
for ( i=0 ; i<N ; i++)
{
    for ( j=i+1 ; j<N ; j++)
        if ( a[i]+a[j]==K)
            return True ;

}
return False;
```

#iterations

i	j	iterations
0	1 → N-1	N-1
1	2 → N-1	N-2
2	3 → N-1	N-3
⋮	⋮	⋮
N-1	[N , N-1]	0

Total iterations

$= 0+1+2+3 \cdots N-1$

$$\boxed{\frac{(N-1)(N)}{2}}$$

$T.C \Rightarrow O(N^2)$

$1+2+3\cdots+N \Rightarrow \frac{N(N+1)}{2}$

$1+2+3+\cdots N-1 \Rightarrow \frac{(N-1)(N-1+1)}{2} \Rightarrow \frac{(N-1)(N)}{2}$

Amazon

MS

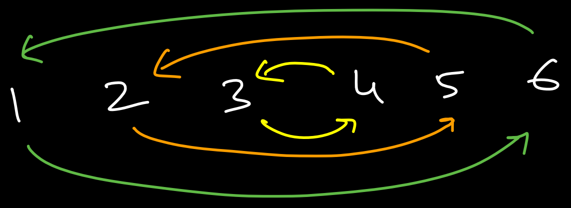
Q. Given array of size N.
Reverse the array using constant extra space.

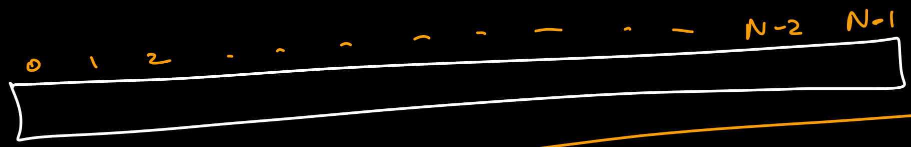
A : [-1, 4, 7, 6, 5, -2, 12, 8, 10]

A : [10, 8, 12, -2, 5, 6, 7, 4, -1]

Obs : 1st element → last element

last element → 1st element





N elements.

ith element → N-i-1 th element

0 → N-1

1 → N-2

⋮

```
void reverse (int arr[ ], int N)
{
                      → i ≤ N/2
    for (i = 0; ~~i < N~~; i++)
        swap(a[i], a[N-i-1]);

}
```

[#< N/2 or ≤ N/2] both are fine

Visualization i<N

$$\begin{array}{c|ccccc} & 0 & 1 & 2 & 3 & 4 \\ A: & 1 & 2 & 3 & 4 & 5 \end{array}$$

i = 0 ⇒ A: [5, 2, 3, 4, 1]

i = 1 ⇒ A: [5 4 3 2 1]

i = 2 ⇒ A: [5 4 3 2 1]

i = 3 ⇒ A: [5 2 3 4 1]

i = 4 ⇒ A: [1 2 3 4 5]

Method 2

```
i = 0;
j = N-1;
while (i<j)
{       swap(a[i], a[j])
        i++; j--;
}
```

iterations

$\boxed{N/2}$.

T.C ⇒ O(N)

S.C ⇒ O(1).

$$\begin{array}{cccc|ccc} & & 0 & 1 & 2 & 3 & \\ A : & [& 1 & 2 & 3 & 4 &] \end{array} \quad N = 4$$

$i < N/2$

$$\begin{array}{cccc|cccc} & & 0 & 1 & 2 & 3 & 4 & \\ A : & [& 1 & 2 & 3 & 4 & 5 &] \end{array} \quad N = 5$$

H.W what is an Anagram?

Doubts :

10:12

45min

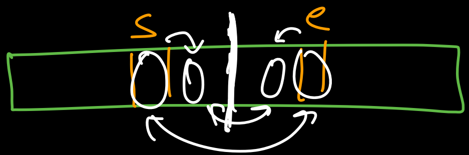
Q. Given an array of size N & 2 indices `s` & `e`. Reverse the elements from s to e. (s ≤ e)

$$\begin{array}{ccccccccccc} & & 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ A = & [& -3, & 1, & 4, & 3, & 5, & 2, & 8, & 12, & 6] \\ & & & & & s & & & e & & \end{array}$$

$s = 3, e = 6$

$A : [-3, 1, 4, 8, 2, 5, 3, 12, 6]$

```
i = s
j = e
while ( i<j)
{     swap (a[i], a[j])
      i++;
      j--;
}
```



iterations.

$\left(\frac{e-s+1}{2}\right) \Leftarrow (e,s).$

N

s, e
s_2, e_2
s_3, e_3.

$s=0, e=N-1 \Rightarrow$ T.C : O(N) worst case

S.C : O(1).

Amazon
Ola
Adobe
M.S.

Q. Given an array of size N, Rotate the array in clockwise direction by 'k' times [using constant extra space]

For now, K < N

N=5. A : [1, 2, 3, 4, 5]

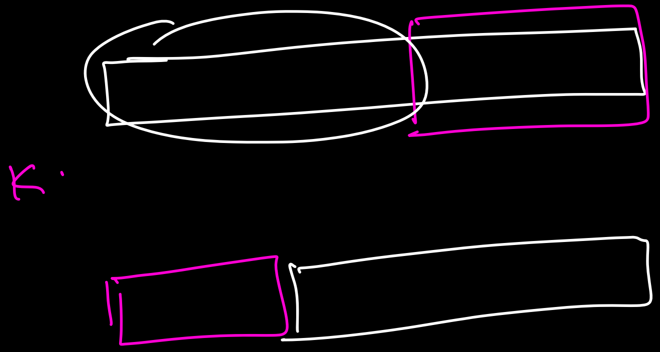
K=1 A : [5, 1, 2, 3, 4].

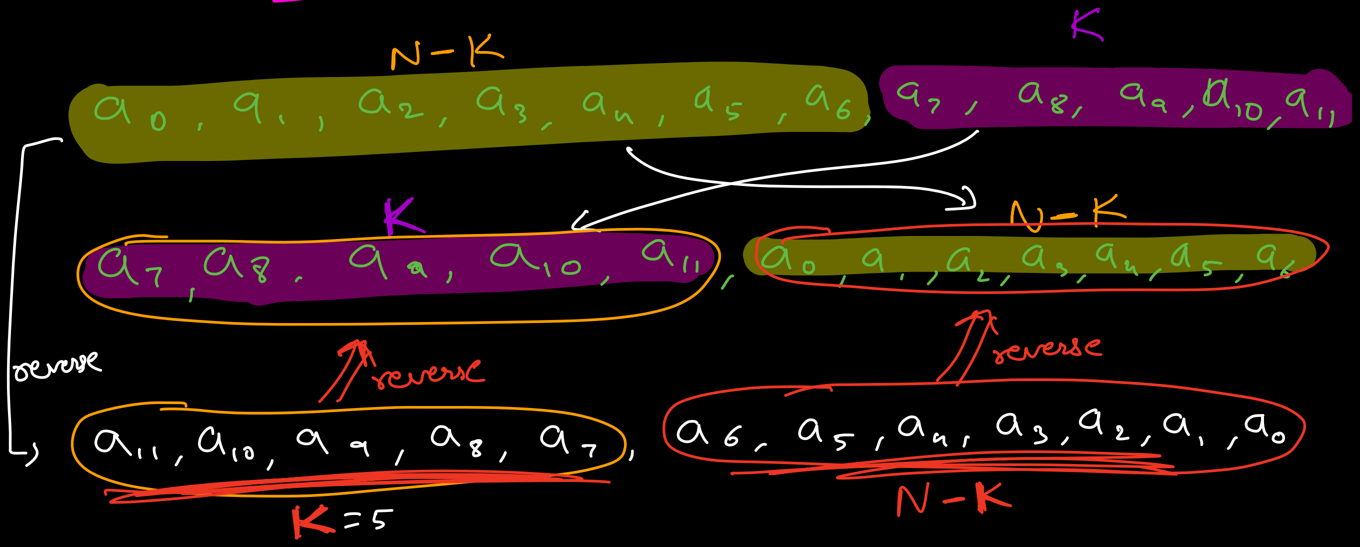
K=2 A : [4, 5, 1, 2, 3]

K=3 A : [3, 4, 5, 1, 2]

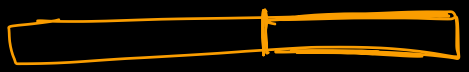
Obs:

K							
	3,	−2,	1,	4,	6,	9,	8
1	8,	3,	−2,	1,	4,	6,	9
2	9,	8,	3,	−2,	1,	4,	6
3	6,	9,	8,	3,	−2,	1,	4
4	4,	6,	9,	8,	3,	−2,	1




N−K
K
$a_0, a_1, a_2, a_3, a_4, a_5, a_6$,
$a_7, a_8, a_9, a_{10}, a_{11}$
K
N−K
$a_7, a_8, a_9, a_{10}, a_{11}$,
$a_0, a_1, a_2, a_3, a_4, a_5, a_6$
reverse
reverse
reverse
$a_{11}, a_{10}, a_9, a_8, a_7$,
$a_6, a_5, a_4, a_3, a_2, a_1, a_0$
K=5
N−K

```
Rotate (arr, K)
{ // Reverse Complete array
    Reverse (arr, 0, N-1) ==> $N/2$
  // Reverse 1st K elements.
    Reverse ( arr, 0, K-1) ==> $K/2$
  // Reverse remaining N-K elements.
    Reverse(arr, K, N-1) ==> $\frac{N-K}{2}$
}
```



iterations.

$\Rightarrow \frac{N}{2} + \frac{K}{2} + \frac{N-K}{2}$

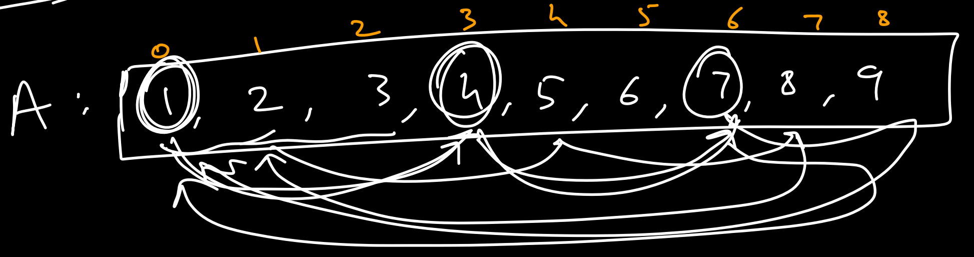
$\Rightarrow \frac{N}{2} + \frac{N}{2}$

$\Rightarrow N$

T.C => O(N)

S.C => O(1)
(Extra)

K=3



% modulo

T.C => O(N)

S.C => O(1)

Ans 7, 8, 9, 1, 2, 3, 4, 5, 6

$\frac{N}{K} * K \Rightarrow N.$

N=6 K=9

A :	a_0	a_1	a_2	a_3	a_4	a_5
K=0 :	a_0	a_1	a_2	a_3	a_4	a_5
K=1 :	a_5	a_0	a_1	a_2	a_3	a_4

K=6 12 18

7 13 19

K=2	:	a_4	a_5	a_0	a_1	a_2	a_3	8	14
K=3	:	a_3	a_4	a_5	a_0	a_1	a_2	9	
K=4	:	a_2	a_3	a_4	a_5	a_0	a_1	10	
K=5	:	a_1	a_2	a_3	a_4	a_5	a_0	11	
K=6	:	a_0	a_1	a_2	a_3	a_4	a_5		

```
if(K>N)
    K = K%N .
```

K=65

N=6

→ $\boxed{K=5}$

Just overview

int arr[10] = → 10 elemets. ←

Fixed size .

Dynamic Arrays → Array size is not fixed dynamic .

(size ↑)

C++	Python	Java	C#	JS/Ruby	C
vectors (STL)	List .	Arraylist	Arraylist	Array	No. dynamic array C++ .

List<int> a;   a.size() => 0.

a[1] → error

```
a.insert(10);        a.size() → 1.
a.insert(21)
```

0	1
10	21

a[i] => $\boxed{O(1)}$

```
for(i=0; i< a.size(); i++)
{
    . . . .
}
```

.insert() => $\boxed{O(1)}$

.size() => $\boxed{O(1)}$

Doubts

i	j	iterations
0	1→N	N
1	2→N	

sum

N

N/2

N/4

.

2N

O(N)

iteration → N

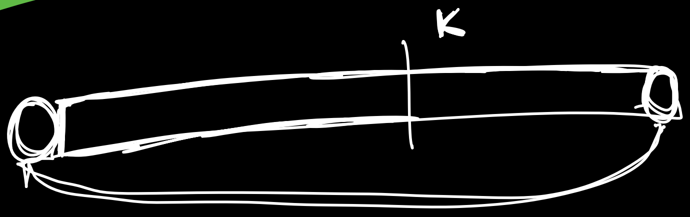
operations → ~~4~~2N+1 → N

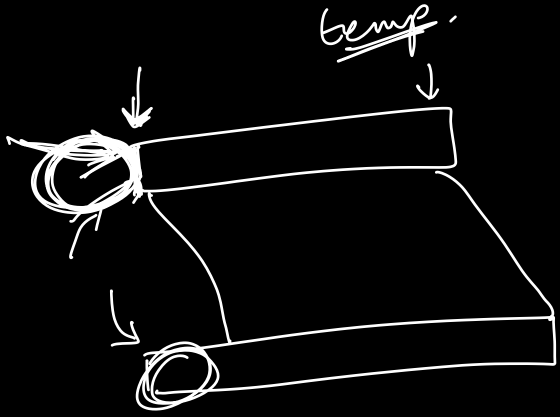
O(N).

```
      1       N      N
for(i=0; i<N; i++)
{
    print(i) → N
    j=i+1 → N
}
```

break

vector. sort()





1 shift → O(N)

K → O(N*K)

O(N).

O(N*K).

Home Work

Program to count no. of elements which have atleast one number to be greater using one loop.

Sol → Steps 1 = find Max.

Step 2 = Count occurences of Max.

0	1	2	3	4	5	6	7		
1	2	5	3	4	7	10	8	10	10

```
Cnt_Max = 1
max = a[0]

for ( i = 0; i < n ; i++)
{
        if(max < a[i])
        {
            max = a[i]
            Count = 1
        }
        else if (max == a[i])
        {
            Count += 1
        }
}
return (len(A) - Count)
```

max = ~~1~~ ~~2~~ ~~5~~ ~~7~~ 10

Count = ~~1~~ ~~1~~ ~~1~~ ~~1~~ ~~12~~ 3

① 2 > 1 ? ✓

② 5 > 2 ? ✓

③ 3 > 5 ? ✗
3 == 5 ? ✗

④ 4 > 5 ? ✗
4 == 5 ✗

⑤ 7 > 5 ? ✓

⑥ 10 > 5 ? ✓

⑦ 8 > ~~1~~0 ? ✗
8 == 10 ? ✗

⑧ 10 > 10 ? ✗
10 == 10 ? ✓

⑨ ~~1~~0 > 10 ? ✗
10 == 10 ? ✓