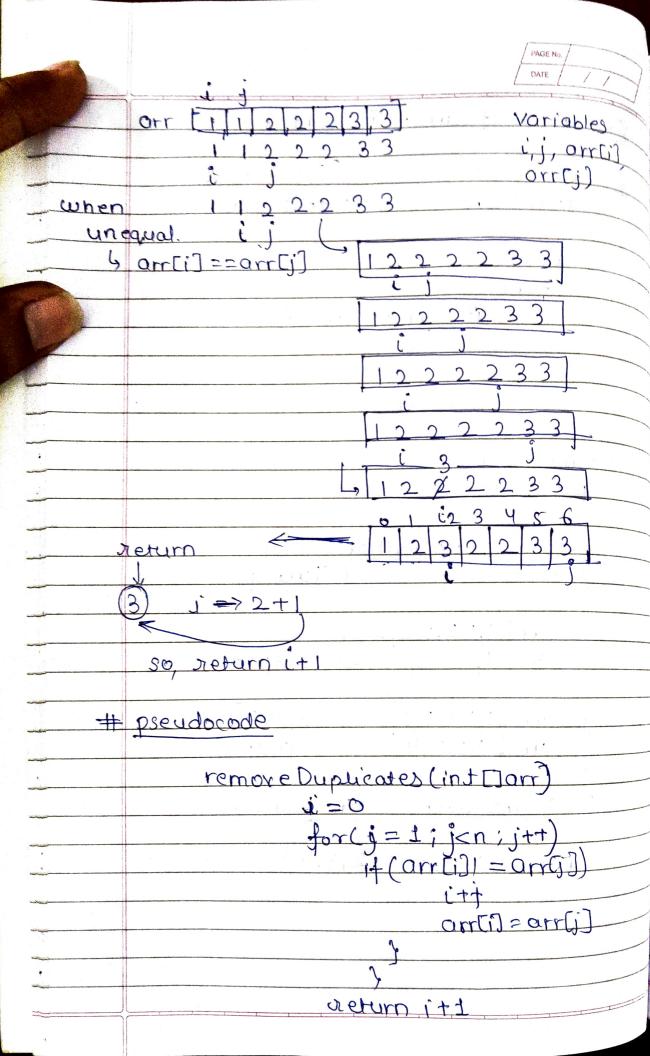
		PAGE No.
12/04/2	25	
#	= Arrays problems:	
	to the are	ay.
<u>Q</u>	3	8,10,5,7,9}
	08x[] = 22,0,40,0	= 10
	output = 5	
		Approach-1
	2 5 1 3 0	Sorting but: time: O(No
		Da . The order
50	best approach	7 /2 - 11- 4
	li Jarges	t=arr[o]
	$\begin{array}{c ccccccccccccccccccccccccccccccccccc$	7711-1
and	0 1 2 3 4	7.7 N. 10 10 10 10 10 10 10 10 10 10 10 10 10
	i Jargest a[i] is o	[i] >Jargest
		5
-4	1 2 5	5 .
	2 5 1 3 5 3	5
	4 5 5 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6	5.
	loop ends: largest = 5.	
#	Pseudoode!	
	1 - 0	
	largest = arr[0]	
	for (i=1; ( <n>; i++) if (arr[i]&gt;largest)</n>	
	Jorgest = arr [i]	
	neturn Jargest	

	PAGE No.	
	PAGE No.  CATE	
世	time: O(N)	
	space: O(1)	
Q2	find second largest and second smallest	
	eg: [1,2,4,7,7,5] eg: [1]	
	second smallest = 2 " " =-1	
	second largest = 5' " " = -1	
	P. A. a. a. a. a. b.	
-	Brute approach.	
0	find smallest, then find number just	
	greater> second smallest.	
•	find largest, then find number just	
	smaller -> Second largest.	
	0 1 2 3 4 5	
	124775 i Small sec small large sec large	
	ê arrij 1 0 1	
	2 1 1 2	
Small =r	nin (small, arrli) 4 &2 1 4	
targe= r	nax (large, orr(i) 7 3 1 7-	
SERVICE CO.	large 7 4	
(1,2)1		
(1,4)		
(1,7) 1	(4,7)7. first traversal!	
(1,7)	$\frac{1}{(7,7)7}$ small = 1	
(1,5)	(7,5)7 Jarge = 7.	
	next Step ignore this, and	
Check again		
	in next iteration: first (smallest, largest) will get removed	
	war. 34 removed	

The state of the s		
	Oute /	
+	an Pseudocode:	
A MARKETTAN AND AND AND AND AND AND AND AND AND A		
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	small larges'	
	Sec-small, sec-large	
	for(i=0; i <n; i++){<="" th=""></n;>	
	if small = math min (small, arr [i]	
The state of the s	lorge = math. max Clorge, orr[i]	
	for (i=0; i <n; i+t)<="" th=""></n;>	
AN AN PORTON AND RESTORED TO AN ARREST THE SECOND S	1) (arr[i] < sec sec small & arr[i] = small	
ocu)	Sec_small = arr[i]	
	if (arr[i] > sec_larg && arr[i] = largest sec_largest = arr[i]	
And the second s	\$ see= lorgese = arren	
Versional and a separate service of the constraint of the service		
and a special control of the second control	time: $O(2n) \approx O(N)$ Space: $O(1)$	
proportion in a state described and the contract of the contra	space.	
the grown was required to the control of the contro		
F TO A COST OF THE PROPERTY OF THE SECOND ASSESSMENT OF THE SECOND ASSE		

Q3 Check if array is sorted, [1,2,3,4,5] [5,4,6,7,8] output: false output = true # Code for (i=1; i<n; i++) if Carrli] < arrli-i] return false return true. # time : O(N) Space : O(1) O4. Remove Duplications in-place from Sorted array. (Two pointers)  $eg \ arr[1,1,2,2,2,3,3]$   $arr[1,2,3,-,-,-] \ output = 3.$ eg arr [1,1,1,2,2,3,3,3,4,4]
arr [1,2,3,4, ...] Output = 4. # Best approach: two pointers. arr = [1, 1, 2, 2, 2, 3, 3]



```
DATE / / /
   # time: O(N)
   # Space: 0(1)
  Q5. jett Rotate the Array by one
     given array = 1/2/3/4/5
                    2345
   · brute:
                     4 e nums
                     5 etest
                           test [0] = nums[]
new
                           test [1] = hums[2]
                           test [2] = nums[3
                           test[K] = nums
                           tem [i-i] = arr[i]
                           tem [n-1] = arrio]
     for(i=1;i<n;i+t)

temp[i-1] = arrti]
     temp[i-1] = arr[0]
   # time : o(n)
   # 5pace : 600(N)
```