

Arrays - I





Topics in Today's class

Array Basics

Find the second largest element in an Array

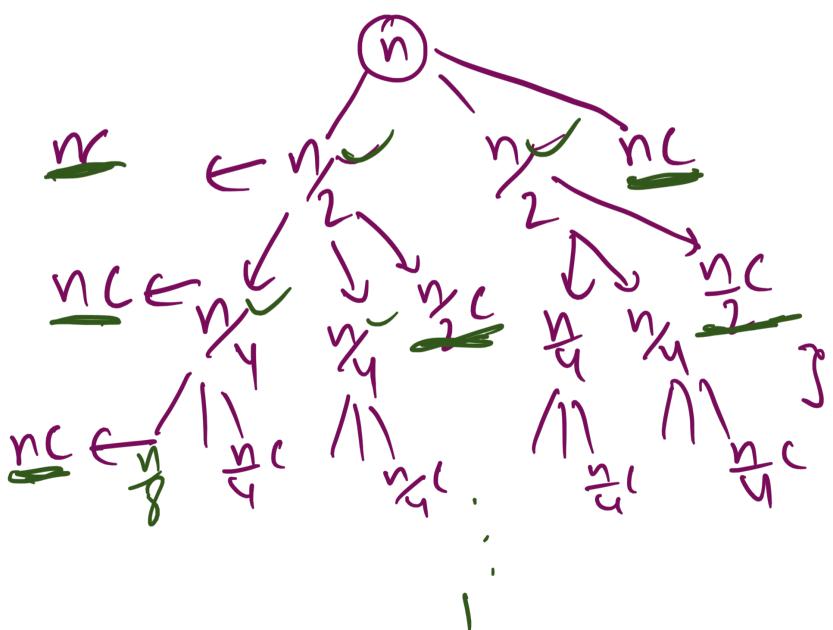
Search in an Array

Remove duplicates from a sorted Array

Delete an Element from an Array

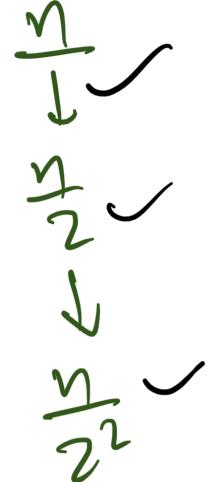
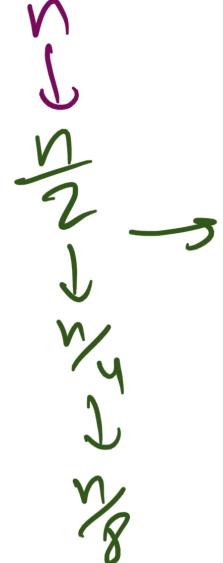
Find the largest element in an Array

Array Basics



$$T(n) = 2T(n/2) + \underline{nc}$$

$f(n)$ &
for ($i=0; i \leq n; i++$)
{
 $\rightarrow f(n/2);$
 $\times f(n/2);$
}
K
 \rightarrow knc .



$$k : \frac{n}{2^k} = 1$$

$$n = 2^k$$

$$\log_2 n = \log_2 2^k$$

$$= k \cancel{\log_2 2} = k$$

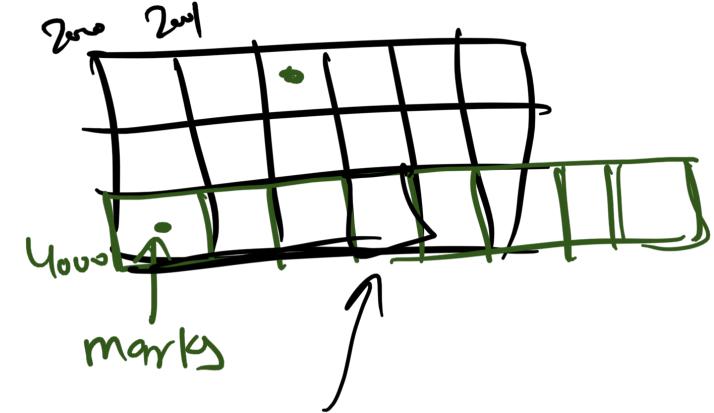
$$\boxed{k = \log n}$$

$\log n \cdot n \not\propto$

$\boxed{O(n \log n)}$

Array Basics

- = Data Structure
- = Contiguous fashion data store
- = Store similar type of data.



`int[] marks = new int[10];`

Specify the size of the array while declaring

$y \rightarrow$ size of int

Diagram illustrating a 1D array 'marks' with 10 slots. The first slot is labeled '10' and the last '41'. Indices 0, 1, 2, 3, 4 are shown above the array. Below the array, a box labeled '4020' has arrows pointing to index 4 and the value '41'. Index 5 is also indicated below the array.

`marks[5] = 41`

$$x = 4000$$

$$y = 40$$

$$\underline{x + 5y} = 4000 + 20 = 4020$$

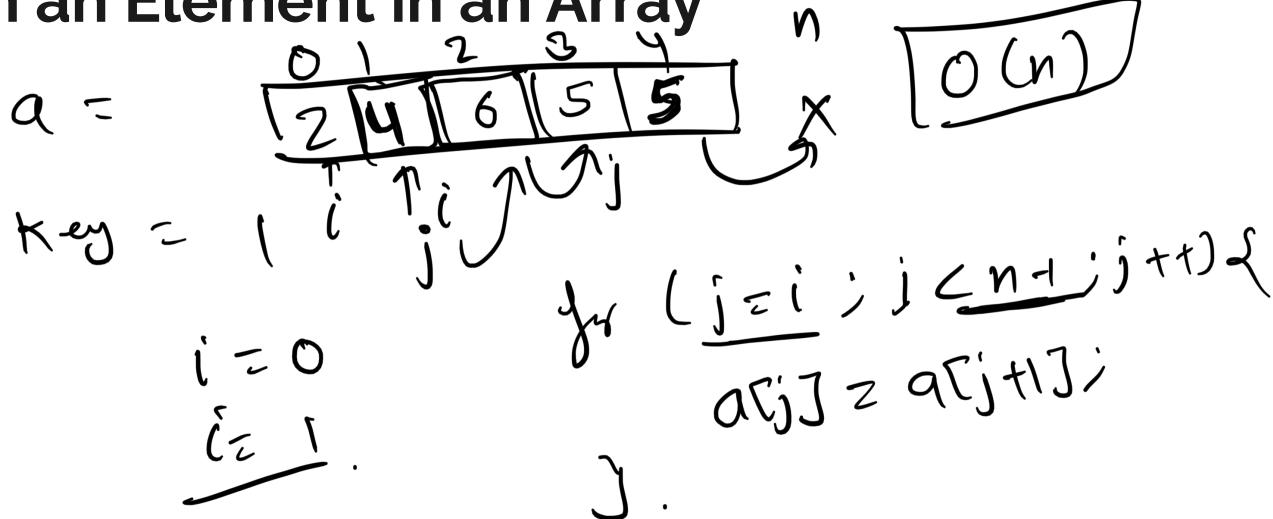
Random access to any index. $\rightarrow O(1)$

Java Array initialise with 0.

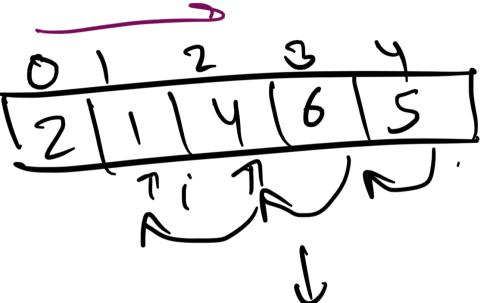
Zero based indexing.

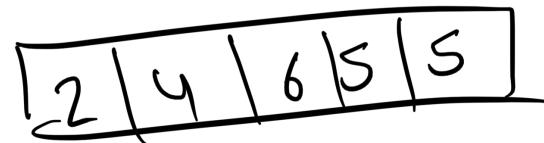
60

Search an Element in an Array

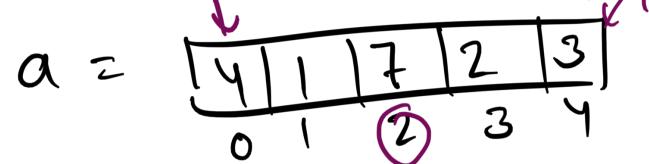


Delete an Element from an Array

$a =$ 
 $i = 1$ $j = 4$



Largest Element in an Array



⑦

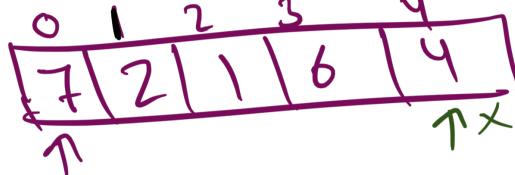
$$\boxed{\text{max} = 2}$$

$\Theta(n)$ ↗
for (int i=1; i < n; i++) {
 if (a[i] > a[max]) {
 max = i;
 }
}

1 4
7 > 4
2 > 7 x
3 7 x

Second Largest Element in an Array

int arr =



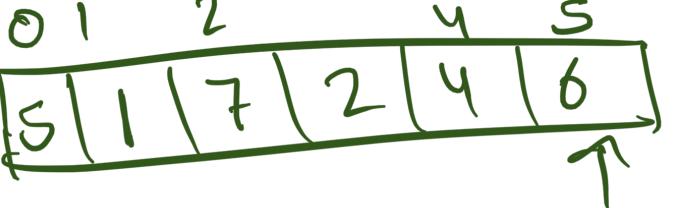
7 ↓ → 0 X
0 ←
6 ↓ → 0 X
0 ← X
↓

largest = 0

second = 3

$\Theta(n)$



int a = 

$$\frac{\text{index}}{d = 2}$$

$$de = 5$$

value ⑦

$$7 = de \quad \text{---}$$

$$5 = de \quad \text{---}$$

Remove Duplicates from a Sorted Array

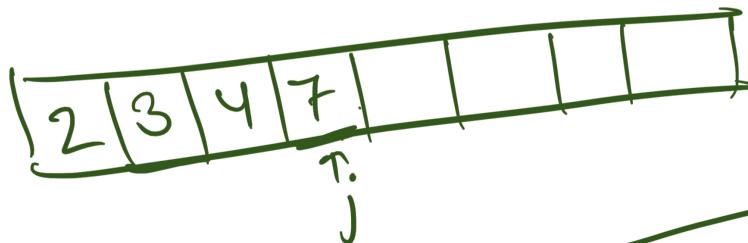
$a =$



$O(n)$

$O(n)$

$O(1)$?



\rightarrow

```
if (a[i] != a[j]) {  
    a[j+1] = a[i];  
    j++;  
}
```

Practice Problems

1. Find the smallest element in the given Array.
2. Find the Third Largest Element in an Array.
3. Check if the Array is Sorted.
4. Reverse the given Array
5. Write a program to replace every element with the greatest element on its right side.

Expected Output :

The given array is : 7 5 8 9 6 8 5 7 4 6

After replace the modified array is: 9 9 9 8 8 7 7 6 6 0

function wala
time complexity for next class.