



Beyond the Basics

ARRAYS & IT'S PROBLEMS



Always remember, beginning is the hardest part.

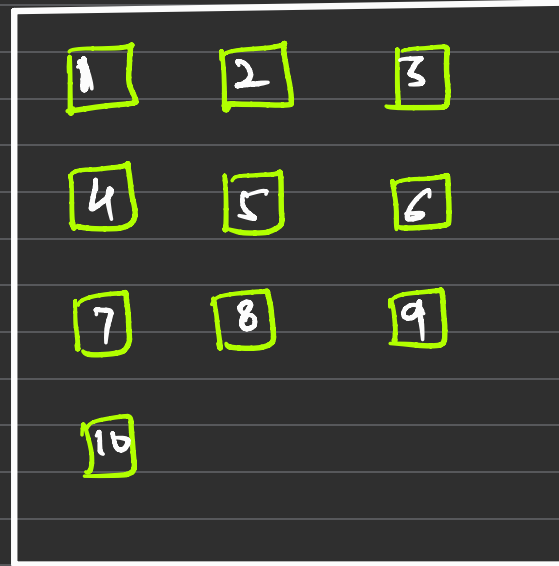
We go to the GYM because we are weak & we wanted to be strong. We go to Library, because we dont have knowledge & we wanted to gain some. Hence, process of discomfort is part of

Arrays:

- Why we need array
- What is array
- How array is Represent
- 3 / 4 problems

Why we need Array

Siddhar.



class

$$x_1 = 10$$

$$x_4 = 60$$

$$x_2 = 30$$

$$x_5 = 70$$

$$x_3 = 40$$

$$x_6 = 80$$

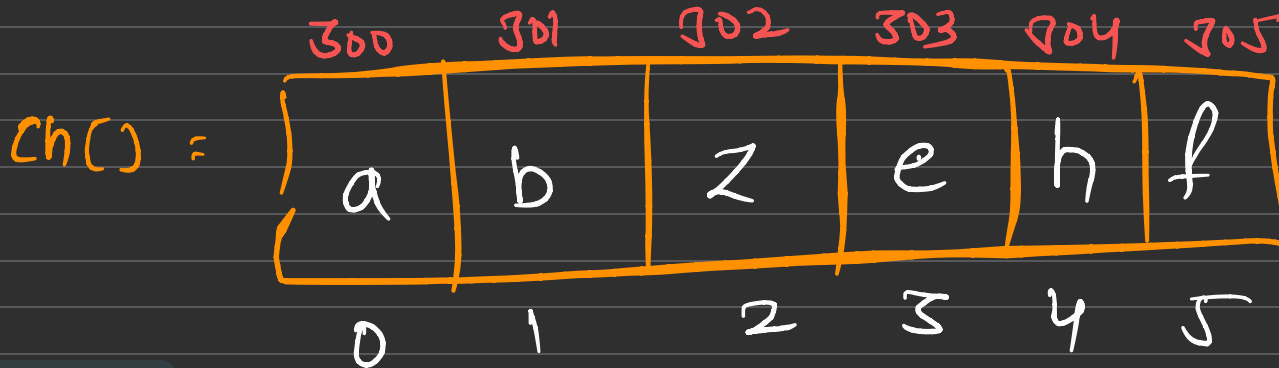
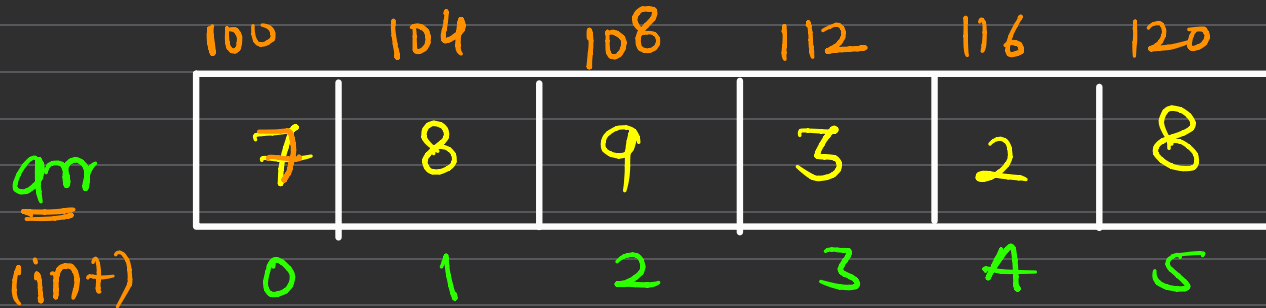
$q(3)$

$q(5)$

$$\underline{q} = \begin{matrix} & 0 & 1 & 2 & 3 & 4 & 5 \\ \left\{ \begin{matrix} x_1 & x_2 & x_3 & x_4 & x_5 & x_6 \end{matrix} \right\} \end{matrix}$$

- group of element
- similar types

Array: collection of elements of similar type
contiguous memory allocation.



find largest element from array?

10	7	1	4	9	14	16	8
0	1	2	3	4	5	6	7

Sort: $O(N \cdot \log N)$

1	4	7	8	9	10	14	16
0	1	2	3	4	5	6	7

(largest)

10	7	1	4	9	14	16	8
0	1	2	3	4	5	6	7

```
int temp = a[0]
for (i = 0 ; i < n ; i++)
{
    if (a[i] > temp)
    {
        temp = a[i]
    }
}

return temp
```


② find second largest element?

10	7	1	4	9	14	16	8
0	1	2	3	4	5	6	7

1	4	7	8	9	10	14	16
0	1	2	3	4	5	6	7

7	8	9	9	8	9
---	---	---	---	---	---

7	8	8	9	9	9
---	---	---	---	---	---

0 1 2 3 4 5

6	6	6	6	6	6
---	---	---	---	---	---

Second largest
X

(-1)

1	4	6	7	9	9	9	9
0	1	2	3	4	5	6	7

sort(arr)

for (i = n-2 ; i ≥ 0 ; i--)

{

if (arr[i] > arr[i+1])

{

return i / arr[i]

}

}

return -1

7	7	7	7
0	1	2	3

N = 4

L = ~~2~~ ~~1~~ ~~0~~ -1

* Second Smallest

10	7	1	4	9	14	16	8
0	1	2	3	4	5	6	7

1	4	7	8	9	10	14	16
0	1	2	3	4	5	6	7

1	1	1	2	3	4
0	1	2	3	4	5

sort(arr)

for (i = 1 : i < n : i++)

}

if (a[i] != a[i-1])

return a[i]

~

return -1

0	1	2	3
7	7	7	7

N = 4

k = 2, 3, 4