Marks Of Pcm

a). Sort the students in Ascending order of their Physics marks.
b). Once this is done, sort the students having same marks in Physics in the descending order of their Chemistry marks.
c). Once this is also done, sort the students having same marks in Physics and Chemistry in the ascending order of their Maths marks.

	0	1	2	3	4	5
phy	50	40	20	50	90	40
chem	45	55	80	48	80	55
math	47	59	60	60	75	51

	P	C	M
2.	20	80	60
5.	40	55	51
1.	40	55	59
3.	50	48	60
0.	50	45	47
4.	90	80	75

```
public static RC {

    int phy;
    int chem;
    int math;

    public int compareTo (RC other) {

        if ( this.phy != o.phy) {
            return this.phy - o.phy;  -> ascending
        }
        else if ( this.chem != o.chem) {
            return - (this.chem - o.chem); -> descending
        }
        else {
            return this.math - o.math; -> ascending
        }

}
```

Search A 2d Matrix

matrix has the following properties:

a). Integers in each row are sorted from left to right.

b). The first integer of each row is greater than the last integer of the previous row.

(i) find potential row, using binary search. → log N

(ii) now find ele in potential row using binary search. → log M

	0	1	2	3	4
0	7	15	20	22	30
1	35	40	45	48	52
2	60	64	67	72	85
3	90	100	120	130	150

		0	1	2	3	4
lo	0	7	15	20	22	30
	1	35	40	45	48	52
m	2	60	64	67	72	85
hi	3	90	100	120	130	150

ele = 125

240. Search a 2D Matrix II

38

Write an efficient algorithm that searches for a `target` value in an `m x n` integer `matrix`. The `matrix` has the following properties:

- Integers in each row are sorted in ascending from left to right.
- Integers in each column are sorted in ascending from top to bottom.

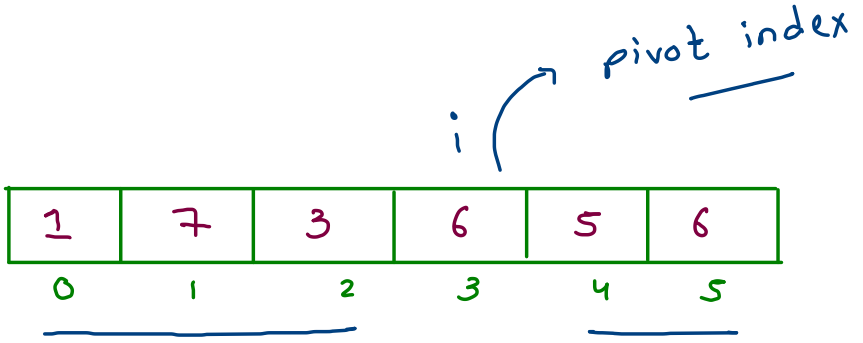
i, j

10	20	30	40	50
15	32	35	42	55
20	38	56	60	70
30	48	61	65	80

linear → m+n

Find Pivot Index

```
6
1 7 3 6 5 6
```



sum = 28

lsum 0 + 8 11

rsum 20 17 11

658. Find K Closest Elements

Medium 3730 389 Add to List Share

Given a **sorted** integer array `arr`, two integers `k` and `x`, return the `k` closest integers to `x` in the array. The result should also be sorted in ascending order.

An integer `a` is closer to `x` than an integer `b` if:

- `|a - x| < |b - x|`, or
- `|a - x| == |b - x|` and `a < b`

X = 1

a = 2 b = 3

→ a because |a-x| < |b-x|

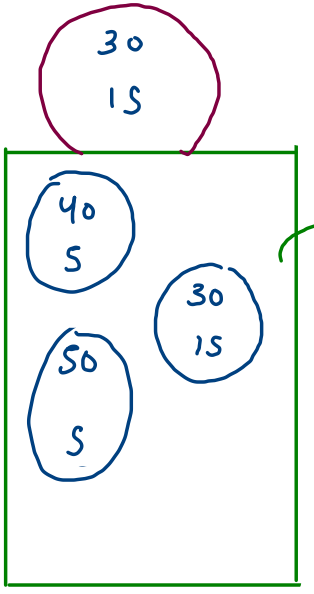
X = 2

a = 1 b = 3

→ a, |a-x| = |b-x|

6
10 20 30 40 50 60
3 = k
45 = x

10 20 30 40 50 60



max heap

30 50 40

| 30 | 40 | 50 |

pair :

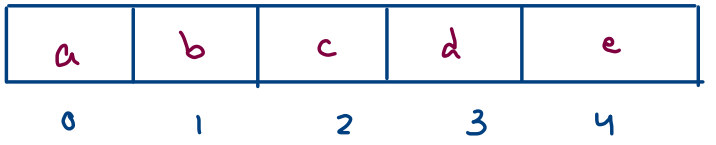
-> val

-> |val-x|

gap

Maximize Sum Of Arr[i]*i Of An Array

1. Given an array arr of N integers.
2. Your task is to write a program to find the maximum value of sum of arr[i]*i, where i = 0,1,2,...,n-1.
3. You are allowed to rearrange the elements of the array.



total possibilities : 5!

↓

Sorted array is the best possibility

$$\sum_{i=0}^{n-1} arr[i] * i$$

Max Sum In The Configuration

Medium

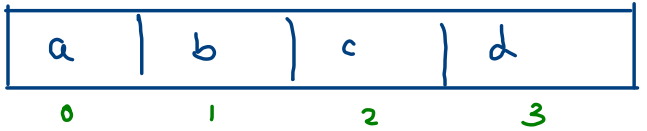
Prev

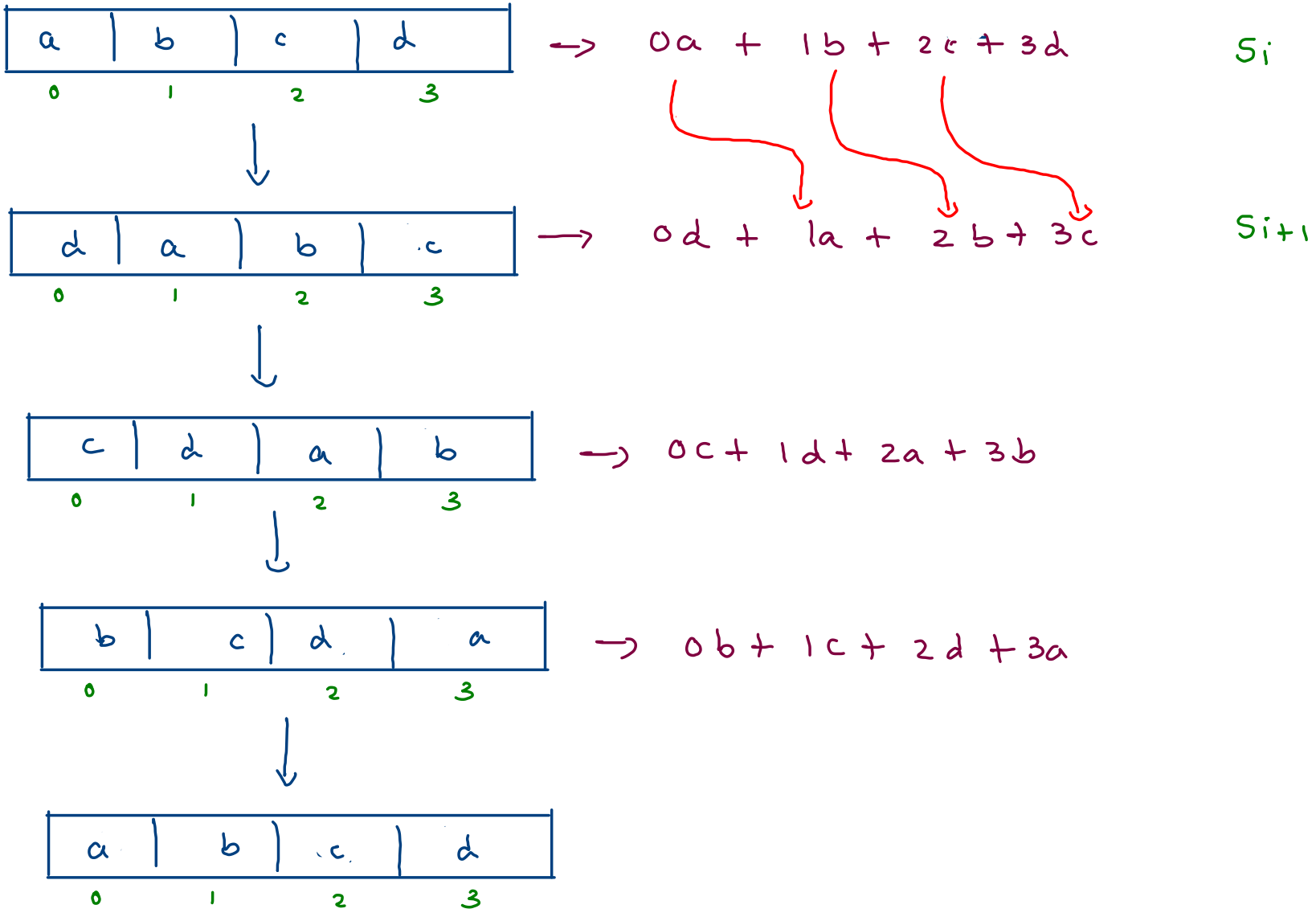
Next

1. Given an array, you have to find the max sum of i*A[i] where A[i] is the element at index i in the array.
2. The only operation allowed is to rotate(clock-wise or counter clock-wise) the array any number of times.

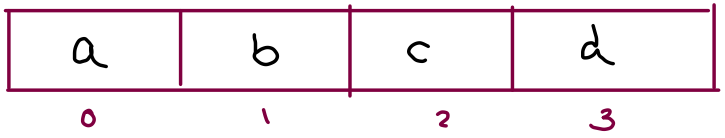
maximise $\sum_{i=0}^{n-1} arr[i] * i$

configurations → n




a b c d
0 1 2 3
→ 0a + 1b + 2c + 3d
Si
d a b c
0 1 2 3
→ 0d + 1a + 2b + 3c
Si+1
c d a b
0 1 2 3
→ 0c + 1d + 2a + 3b
b c d a
0 1 2 3
→ 0b + 1c + 2d + 3a
a b c d
0 1 2 3

$$S_{i+1} = S_i + sum - n * arr[n-i-1]$$



$$S_0 = 0a + 1b + 2c + 3d$$

$$S_1 = 0a + 1b + 2c + 3d + (a+b+c+d) - 4*d$$

$$= 0d + 1a + 2b + 3c$$

$$S_2 = 0d + 1a + 2b + 3c + (a+b+c+d) - 4*c$$

$$= 0c + 1d + 2a + 3b$$