

⇒ Custom Sort (alters the properties of)
inbuilt function

arr =

| (25) | (9) | (4) | (64) | (49) |
|------|-----|-----|------|------|
| 5 | 3 | -2 | 8 | 7 |

~~different
arrangements~~

; -2 8 5 3 7

; -2 3 5 7 8

Note :- Custom sort never effect the time Complexity.

Arrays.sort(arr, new myComparator());
↳ declaration

→ implementation

```
public static class myComparator implements Comparator<Integer> {
    @Override
    public int compare(Integer a, Integer b) {
        return b-a; // decreasing order
    }
    return a-b; // increasing order
}
```

Imp Note :-

- ↳ Increasing order on the basis of value : $a - b$
- ↳ Decreasing order on the basis of value : $b - a$

```
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    int n = scn.nextInt();
    Integer[] arr = new Integer[n];
    // input
    for (int i = 0; i < n; i++) {
        arr[i] = scn.nextInt();
    }

    // in built function
    Arrays.sort(arr, new myComparator());

    // printing
    for (int i = 0; i < n; i++) {
        System.out.print(arr[i] + " ");
    }
}

public static class myComparator implements Comparator<Integer> {
    @Override
    public int compare(Integer a, Integer b) {
        return b - a;
    }
}
```

Sort the array according to their Square of each element

| | | | | | |
|------|-----|-----|------|------|-----|
| (25) | (4) | (9) | (16) | (81) | (1) |
| 5 | -2 | -3 | 4 | 9 | 1 |

arr =
 0 1 2 3 4 5

increasing :- -3 -2 1 4 5 9

increasing sq :- 1 -2 -3 4 5 9

$$a - b$$

↑ing order

$$a^2 - b^2$$

$$Q_{\text{irr}} = \begin{matrix} 4 & -1 & 0 & -5 & 6 \\ (16) & (1) & (0) & (25) & (36) \\ \underline{\underline{=}} & \underline{\underline{=}} & \underline{\underline{=}} & \underline{\underline{=}} & \underline{\underline{=}} \end{matrix}$$

$$\text{ans :- } \begin{matrix} 6 & -5 & 4 & -1 & 0 \\ \underline{\underline{=}} & \underline{\underline{=}} & \underline{\underline{=}} & \underline{\underline{=}} & \underline{\underline{=}} \end{matrix}$$

$b^2 - a^2$

| | | | | | |
|-------|------|------|-----|------|-----|
| | (-8) | (27) | (0) | (-1) | (8) |
| Ans = | -2 | 3 | 0 | -1 | 2 |

↳ decreasing order of cube :- $b^3 - a^3$

| | | | | | |
|-------|---|---|---|----|----|
| Ans = | 3 | 2 | 0 | -1 | -2 |
|-------|---|---|---|----|----|



code

```
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    int n = scn.nextInt();
    Integer[] arr = new Integer[n];
    for (int i = 0; i < n; i++) {
        arr[i] = scn.nextInt();
    }

    Arrays.sort(arr, new myComparator());

    for (int i = 0; i < n; i++) {
        System.out.print(arr[i] + " ");
    }
}

public static class myComparator implements Comparator<Integer> {
    @Override
    public int compare(Integer a, Integer b) {
        return a * a - b * b;
    }
}
```

→ Lambda function

Syntax

```
Arrays.sort(arr, (a, b) -> {  
    // return a - b;  
});
```

Code

```
public static void main(String[] args) {  
    Scanner scn = new Scanner(System.in);  
    int n = scn.nextInt();  
    Integer[] arr = new Integer[n];  
    for (int i = 0; i < n; i++) {  
        arr[i] = scn.nextInt();  
    }
```

```
    Arrays.sort(arr, (a, b) -> {  
        return a * a - b * b;  
    });  
  
    for (int i = 0; i < n; i++) {  
        System.out.print(arr[i] + " ");  
    }  
}
```

Imp

- ↳ If we return -ve value :- ascending order
(-1) [a appears first]
- ↳ If we return +ve value :- descending order
(+1) [b appears first]
- ↳ If we return $a-b$:- ascending order
based on values
- ↳ If we return $b-a$:- descending order
based on values

→ GMP

| | | | | | |
|---|---|---|----|---|----|
| 5 | 2 | 1 | -2 | 5 | -7 |
|---|---|---|----|---|----|

myself other (a, b)

$$\begin{array}{c} \xrightarrow{\quad} \\ a - b = \end{array} \begin{array}{c} +1 \\ -1 \\ 0 \end{array}$$
$$\Rightarrow \begin{array}{c} \textcircled{b-a} \\ \underline{\underline{b-a}} \end{array}$$
$$\Rightarrow \begin{array}{c} \textcircled{a-b} \\ \underline{\underline{a-b}} \end{array}$$
$$\Rightarrow \begin{array}{c} -1 \\ \underline{\underline{-1}} \end{array}$$
$$\Rightarrow \begin{array}{c} +1 \\ \underline{\underline{+1}} \end{array}$$

$$[5, -7]$$

$$\downarrow \quad \downarrow$$
$$[-7, 5]$$

$$[5, -7]$$

$$[-7, 5]$$

Sort Array By Parity

(Imp)

arr =

| | | | | | | |
|----|----|----|----|----|----|----|
| 3 | 4 | 2 | 8 | 7 | 1 | 5 |
| ↑↑ | ↑↑ | ↑↑ | ↑↑ | ↑↑ | ↑↑ | ↑↑ |

~~Que~~ ↳ all even values first & then odd values

↳ in non-decreasing order (↑ing order)

arr =

| | | | | | | |
|------|---|---|-----|---|---|---|
| 4 | 2 | 8 | 3 | 7 | 1 | 5 |
| even | | | odd | | | |

arr =

| | | | | | | |
|---|---|---|---|---|---|---|
| 2 | 4 | 8 | 1 | 3 | 5 | 7 |
|---|---|---|---|---|---|---|

Logic

$a \rightarrow \text{even}$, $b \rightarrow \text{odd}$ return -1

$a \rightarrow \text{odd}$, $b \rightarrow \text{even}$ return +1

$a \rightarrow \text{odd}$, $b \rightarrow \text{odd}$ return $a - b$

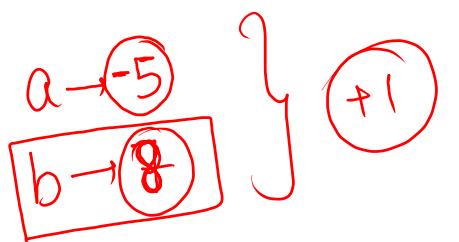
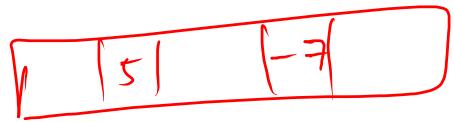
$a \rightarrow \text{even}$, $b \rightarrow \text{even}$ return $a - b$

Code

$T.C = O(N \log(N))$
 N is size array

$S.C = O(1)$

```
public static void main(String[] args) {  
    Scanner scn = new Scanner(System.in);  
    int n = scn.nextInt();  
    Integer[] arr = new Integer[n];  
    // input  
    for (int i = 0; i < n; i++) {  
        arr[i] = scn.nextInt();  
    }  
  
    // in built function  
    Arrays.sort(arr, (a, b) -> {  
        if (a % 2 == 0 && b % 2 != 0) { // a is even and b is odd  
            return -1;  
        } else if (a % 2 != 0 && b % 2 == 0) { // a is odd and b is even  
            return 1;  
        } else if (a % 2 == 0 && b % 2 == 0) { // both even  
            return a - b;  
        } else { // both odd  
            return a - b;  
        }  
    });  
  
    // printing  
    for (int i = 0; i < n; i++) {  
        System.out.print(arr[i] + " ");  
    }  
}
```



One arrange, odd values should be first
and then even values, and odd values
should be sorted based on cube and even
in Tiling values on basis of
square in Tiling order

$$a \rightarrow \text{even} \quad , \quad b \rightarrow \text{odd} \quad \longrightarrow \quad +1$$

$$a \rightarrow \text{odd} \quad , \quad b \rightarrow \text{even} \quad \longrightarrow \quad -1$$

$$a \rightarrow \text{odd} \quad , \quad b \rightarrow \text{odd} \quad \longrightarrow a*a*a - b*b*b$$

$$a \rightarrow \text{even} , \quad b \rightarrow \text{even} \quad \rightarrow \quad b*b - a*a$$