Comparator

```
p.s.v.m(String[] args) {
Integer arr[] = {25, -2, 4, 90, 6};
```

6 15

-1 < 0 a-b

1 > 0 15 - 6 = 9

→ swap

a-b → -1

b-a = 1

6 5

a-b.

15-6 = 9.

```
Comparator<Integer> myimpl = (a,b)->b-a;
```

b-a = 1

a = 15

b = 6

```
Arrays.sort(arr, myimpl);
```

→a →b

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

↳ nlogn → TimSort

↳ MergeInsertion swap

If result is greater than zero then it will be swap, that mean a will move to b and b to a which means b will come before and if less than zero then a will come before b.

a b

1 3 2 4

1 2 3 4

1 2 4 3

2 1 4 3

2 4 1 3

a = 1
b = 3

1 - 3 = -2 < 0

Minimum Difference 7

9 4 1 7

↳ 1 4 7 9

3 3 2

min 3 min

k = 3.

1 4 7 9

n = 4.

(1)

```
int min = Integer.MAX_VALUE;
Arrays.sort(arr);
for(int i=0; i<n-1; i++){
    if(arr[i+1]-arr[i]
        <min){
    min = arr[i+1]-arr[i];
    }
}
```

0 1 2 3

1 4 7 9

1 4 7

lowest biglest

6 < min

4 7 9

lowest biglest

9-4 = 5 < min

```
int min = Integer.MAX_VALUE;
for(int i=0; i<=n-k; i++){
    if(arr[i+k-1]-arr[i]
        <min){
    min = arr[i+k-1]
        -arr[i];
    }
}
```

i = 0, 1, 2

i = 1, 2, 3

n = 5

0 1 2 3 4

1 4 7 9 12

k = 3

0 1 2

0 1 4 7

1 4 7 9

1 2 3 4

2 7 9 12

2 3 4

2 7 9 12

2+3-1

1+3-1

k = 4

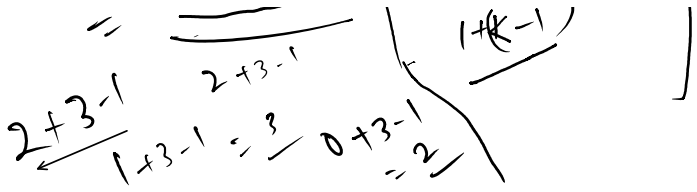
0 1 2 3

0 1 4 7 9

1 4 7 9 12

1 2 3 4

1+(k-1)


2+3-1
2+3-1
1+3-1 = 3
0+3-1
= 2
1+(k-1)

Form the largest Number

4 46 8 9 → 46 9 8 4

44689

0 1 2 3 4 →

4 6 4 8 9

9/10 = 0

98464

46

$9\times10^0 + 8\times10^1 + 4\times10^2 + 6\times10^3 + 4\times10^4$

461

461/10 = 46

46/10 = 4 →

4/10 = 0

46/10 = 4

4612/10 = 461

461/10 = 46

461/10 = 4

a → 9

b → 4

4 − 9 = −5

a = b

469/10 = 46

46/10 = 4

4/10 = 0

46

46 → int → "46" → S

S.charAt(0) → 4

Peak Elements