Quy   arr = | 3 | 5 | 2 | 7 | 9 |

Largest no = 97532
               → solution,

arr = | 1 | 9 | 25 | 21 | (925)
            25   9   (259) } ✓
                                       largest

(31)   a b → ab]
 2   ba] ✓

arr → | 1 | 9 | 25 | 21 | 3 | ASCII

num = | "1" | "9" | "25" | "21" | "3" | (927)
                                                (279) ASCII
                        9 27
⇒ Arrays.sort (num, (a, b) → {   [+ve]
        string (ab) = a+b;   [-ve]
        stolu (ba) = b+a;   [0]
        return ab.compareTo(ba);
     })
                  → lexographically sort

(geek)
(geebster) → lay

geek

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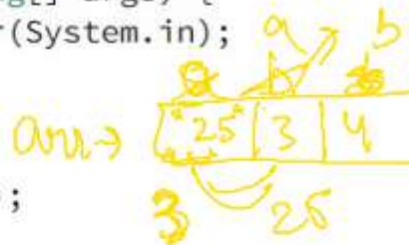
```java
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    int n = scn.nextInt();
    int[] arr= new int[n];
    for(int i=0;i<n;i++){
        arr[i]= scn.nextInt();
    }

    String[] nums = new String[n];
    for(int i=0;i<arr.length;i++){
        nums[i]= String.valueOf(arr[i]);
    }

    Arrays.sort(nums,(a,b)->{
        String s1= a+b;
        String s2 = b+a;
        return s2.compareTo(s1);
    });

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

    /* Enter your code here. Read input from STDIN. Print output to STDOUT. Your
}
```

a → b

arr → | 25 | 3 | 4 |

3   25

ab = 253
ba = 325

325

325.compareTo(253);

325 - 253 = +ve , a

+ve
- ve
0

Array ✓ = int[] arr = new int[7]   String[] str = new String[8];

ArrayList = [ArrayList <Integer> arr = new ArrayList <>();]
     ↳ classes   String

ArrayList

arr.add(90)

arr.add(100)

{10

arr[i]

.charAt(i)

.leg()

arr   5k

. arr.get(i);
         ↓
         5

      ↓
      60

         0 1 2 3 4 5 6
arr |10|22|30|80|00|60|70|
4k ↓

     |10| 22| 30|40|50|60|70|90|100|110|-|-|--
sr   0 1 2 3 4 ⑤ 6 7 8 91 - - -
                  ↓ 2.4 length

[| | | | | | | | | | | | | | | | | | | |]

arr[i] = val

arr.add(i, val);

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

    ArrayList<Integer> sol = PeaKFinder(arr);
    for(int i=0;i<sol.size();i++){
        System.out.print(sol.get(i)+" ");
    }
    /* Enter your code here. Read input from STDIN. Print output to STDOUT. You
}

public static ArrayList<Integer> PeaKFinder(int[] arr){
    ArrayList<Integer> ans = new ArrayList<>();

    for(int i=1;i<arr.length-1;i++){
        if(arr[i]>arr[i-1] && arr[i]> arr[i+1]){
            ans.add(arr[i]);
        }
    }

    return ans;
}
```

$arr[i-1] < arr[i] > arr[i+1]$ $Sol = 5, 8$

arr= | 4 | 5 | 3 | 8 | 6 | 1 |

5K

heap

5K,
arr | 4 | 5 | 3 | 8 | 6 | 1 |
      0   1   2   3   4   5

peakfinder | arr=5K,
ans=8K,

main | arr=5K, Sol=8K

8K

any | 5 | 8 |

8K

5   8

Solution

Qu.) Subarray

arr → | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
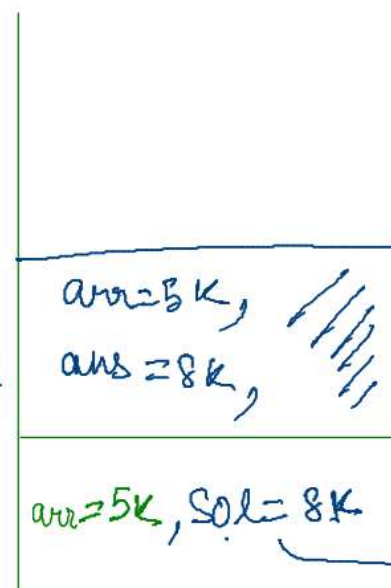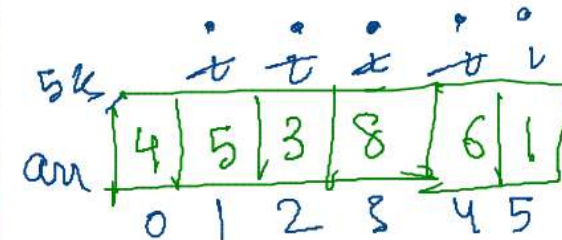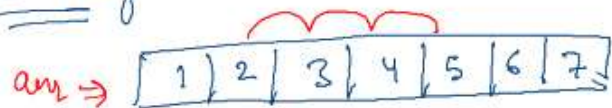
A subarray is (continious) part of an array

```
1              2              3        4
1 2            2 3            3__      4---
1 2 3          2 3 4         ⊆        ⊆
1 2 3 4        2 3 4 5       ⊆        ⊆ __
1 2 3 4 5      ⊆ —⊆—
1 2 3 4 5 6    ⊆
1 2 3 4 5 6 7  ⊆
```

```
i  j  i
| 1 | 2 | 3 | 4 | 5 | 6 |
  0   1   2   3   4   5
  I   j   r
```
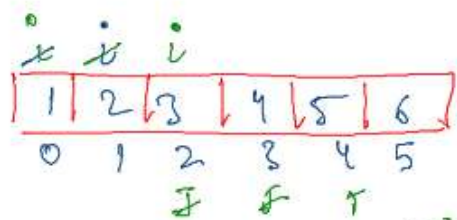
for (i =0 to length) {
 ↳ for (j= i to length) {
  ↳ for (k= i to j) {
   [Syso (arr[k]);
  }
  Syso ();
 }
}

O(n³)

O(n²)

```
1              3
1 2            3 4
1 2 3          3 4 5
1 2 3 4
1 2 3 4 5
1 2 3 4 5 6
2
2 3
2 3 4
2 3 4 5
2 3 4 5 6
```

Qu4

Subarray sum equals to zero

arr =

1	3	5	-4	1	-5	8	6
0	1	2	3	4	5	6	7

Bruteforce → $= O(n^2)$

1
1 3
1 3 5   Sum =
1 3 5 -4
1 3 5 -4 1   if (sum == 0) {
                  Sru (true)
               }

sum

0   50
50

Syn (false);

| 1 | 3 | 5 | ~4 | 1 | ~5 | 8 | 6 |

HashSet
↓
To store unique
elements

$TC = O(n)$

$SC = O(n)$

× 9   × 8

y ≥ 1

y - k = x

HS

prefsum →

| 1 | 4 | 9 | 5 | 6 | 1 | 9 | 15 |

HS (circle containing: 1, 4, 9, 5, 6)
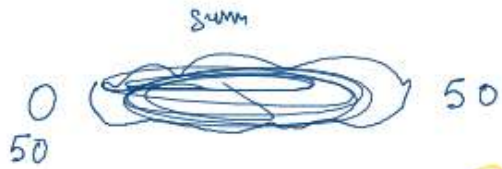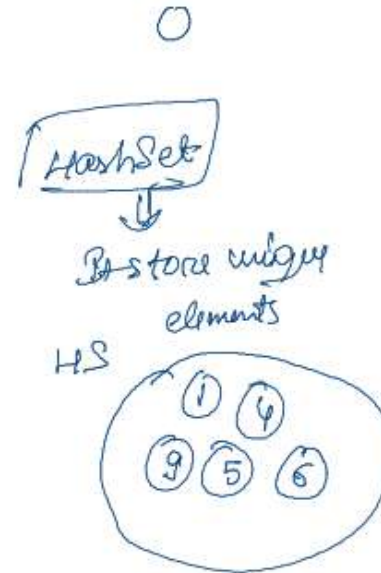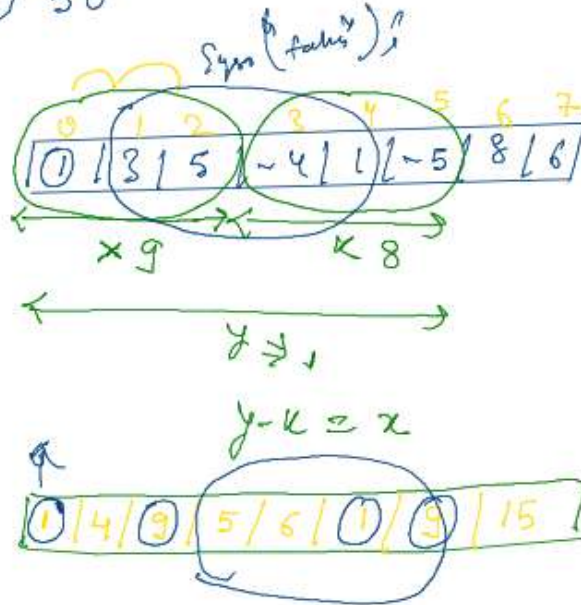
HashSet<Integer> HS = new Hashset<>();

contains (obj) = true   → tru
                            su

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

    for(int i=0;i<arr.length;i++){
        for(int j=i;j<n;j++){
            int sum=0;
            for(int k =i;k<=j;k++){
                sum +=arr[k];
            }
            if(sum ==0){
                System.out.print("true");
                return;
            }
        }
    }

    System.out.println("false");
    /* Enter your code here. Read input from STDIN. Print output to STDO
}
```