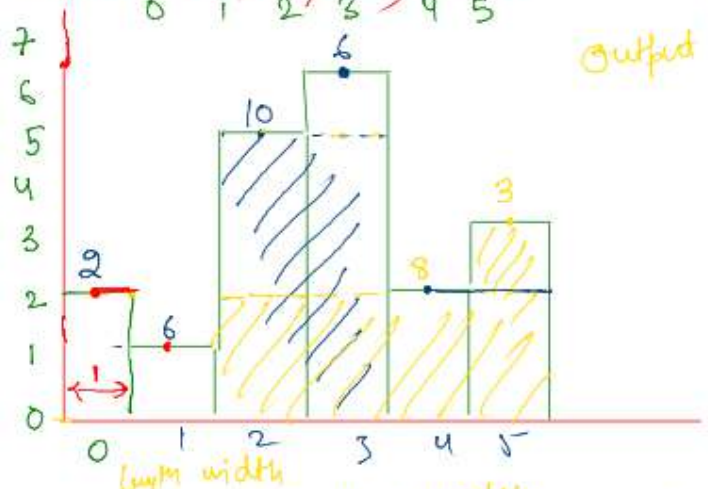
Ques Largest Area Histogram

arr = [2, 1, 5, 6, 2, 3]
0 1 2 3 4 5



output having max. Area.

index

area = 8 ?

length * width

arr[i] (calculate)

(prefix[i] + suffix[i-1])

4 * 2

(st.peek() - i)

max = 10

Leetcode Hard problem

i	0	1	2	3	4	5
arr:	2	1	5	6	2	3
left[] =	1	2	1	1	3	1
Right[] =	1	5	2	1	2	1
area =	2	6	10	6	8	3

i = 5

n = 6

n - i

stack

if (st.empty()) { Right[i] = n - i;

else Right[i] = st.peek() - i

}

width = left[i] + Right[i] - 1

max

area = arr[i] * width

max = 10

Sysо (10);

```java
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    int n = scn.nextInt();
    int[] height = new int[n];

    for(int i=0;i<n;i++){
        height[i]= scn.nextInt();
    }

    int[] left = new int[n];
    int[] right = new int[n];

    Stack<Integer> st = new Stack<>();

    for(int i=0;i<n;i++){
        while(st.size()>0 && height[st.peek()]>=height[i]){
            st.pop();
        }

        if(st.size()==0){
            left[i]= i+1;
        }else{
            left[i]= i-st.peek();
        }
        st.push(i);
    }


     st = new Stack<>();
     for(int i= n-1;i>=0;i--){
          while(st.size()>0 && height[st.peek()]>=height[i]){
              st.pop();
         }

         if(st.size()==0){
             right[i]= n-i;
         }else{
             right[i]= st.peek()-i;
         }
         st.push(i);
    }

    int maxArea=0;

    for(int i=0;i<height.length;i++){
        int width = left[i]+right[i]-1;
        int area = height[i]* width;
        maxArea = Math.max(area,maxArea);
    }

    System.out.print(maxArea);
    /* Enter your code here. Read input from STDIN. Print output to S
}
```

HashMap

Syntax = HashMap< Key , Value> hm = new HashMap<>();

- HashMap → Class name
- Key, Value → Type
- hm → object name
- new → to allocate memory
- HashMap<>() → constructor

1. keys must be unique
2. value can be repeatitive

[3 3 4 4 4 5 5 6 7]

HashMap

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

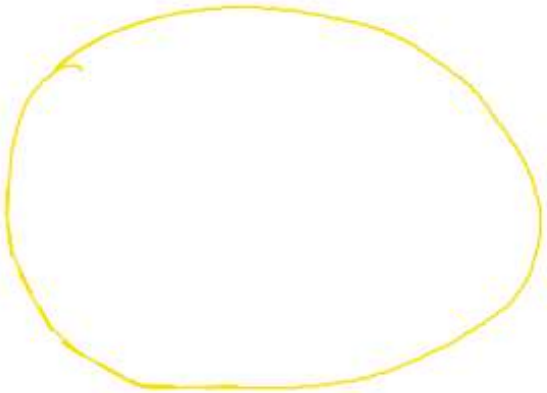
int, string, arr[]

{ Kartik = " 123 ";
Ranjeet = " ";
Rajnesh = " ";
Karthik = " "; }

You are required to create a **dictionary** consisting of word and its meaning.

Take an integer N as input and **Continue** the process untill **Case 4** is not achieved.

- If **N==1**, take **word** and **meaning** as input from user and **add** it to the dictionary.
- If **N==2**, take a **word** as input from the user and print its **meaning**, if the word is not found print **-1**.
- If **N==3**, take a **word** as input from the user and delete it from the dictionary.
- If **N==4**, **Close** the dictionary(**Exit** the program).



n=1 good
n=2 good
n=3 good
n=4 terminate.

```
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    HashMap<String, String> dictionary = new HashMap();
    while(true){
        int n = scn.nextInt();
        if(n==1){
            String word = scn.next();
            String meaning = scn.next();
            dictionary.put(word,meaning);
        }else if(n==2){
            String word = scn.next();
            if(dictionary.containsKey(word)){
                System.out.println(dictionary.get(word));
            }else{
                System.out.println("-1");
            }
        }else if(n==3){
            String word = scn.next();
            dictionary.remove(word);
        }else{
            break;
        }
    }
```

Ques Same as Frequency

```
10
4 5 -3 8 -3 4 4 -3 6 4
```

Arrays.sort(arr); → 4

→ -3 4

4 -3

if(Math.abs(arr[i]) == hm.get(arr[i]))

Sysout(key);

4

4 = 0
5 = 1
-3 = 0
8 = 1
6 = 1

4
-3

```
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    int n = scn.nextInt();
    int[] arr = new int[n];
    HashMap<Integer,Integer> hm = new HashMap<>();
    for(int i=0;i<n;i++){
        arr[i]= scn.nextInt();
        if(hm.containsKey(arr[i])){
            int val = hm.get(arr[i]);
            hm.put(arr[i],val+1);
        }else{
            hm.put(arr[i],1);
        }
    }

    Arrays.sort(arr); //O(nlogn)
    for(int i=0;i<n;i++){
        if(Math.abs(arr[i])==hm.get(arr[i])){
            System.out.println(arr[i]);
            hm.put(arr[i],0);
        }
    }
    /* Enter your code here. Read input from STDIN. Print output to S
}
```

O(n) + O(n) + O(nlogn)

2O(n) + nlogn

O(nlogn)

Find all frequency

```
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    int n = scn.nextInt();
    char[] arr= new char[n];
    HashMap<Character,Integer> hm = new HashMap();
    for(int i =0;i<n;i++){
        arr[i]= scn.next().charAt(0);
        if(hm.containsKey(arr[i])){
            hm.put(arr[i],hm.get(arr[i])+1);
        }else{
            hm.put(arr[i],1);
        }
    }

    Arrays.sort(arr);

    for(int i=0;i<arr.length;i++){
        if(hm.get(arr[i])>0){
            System.out.println(arr[i]+" "+hm.get(arr[i]));
            hm.put(arr[i],0);
        }
    }
    /* Enter your code here. Read input from STDIN. Print output to STDOUT. \
}
```