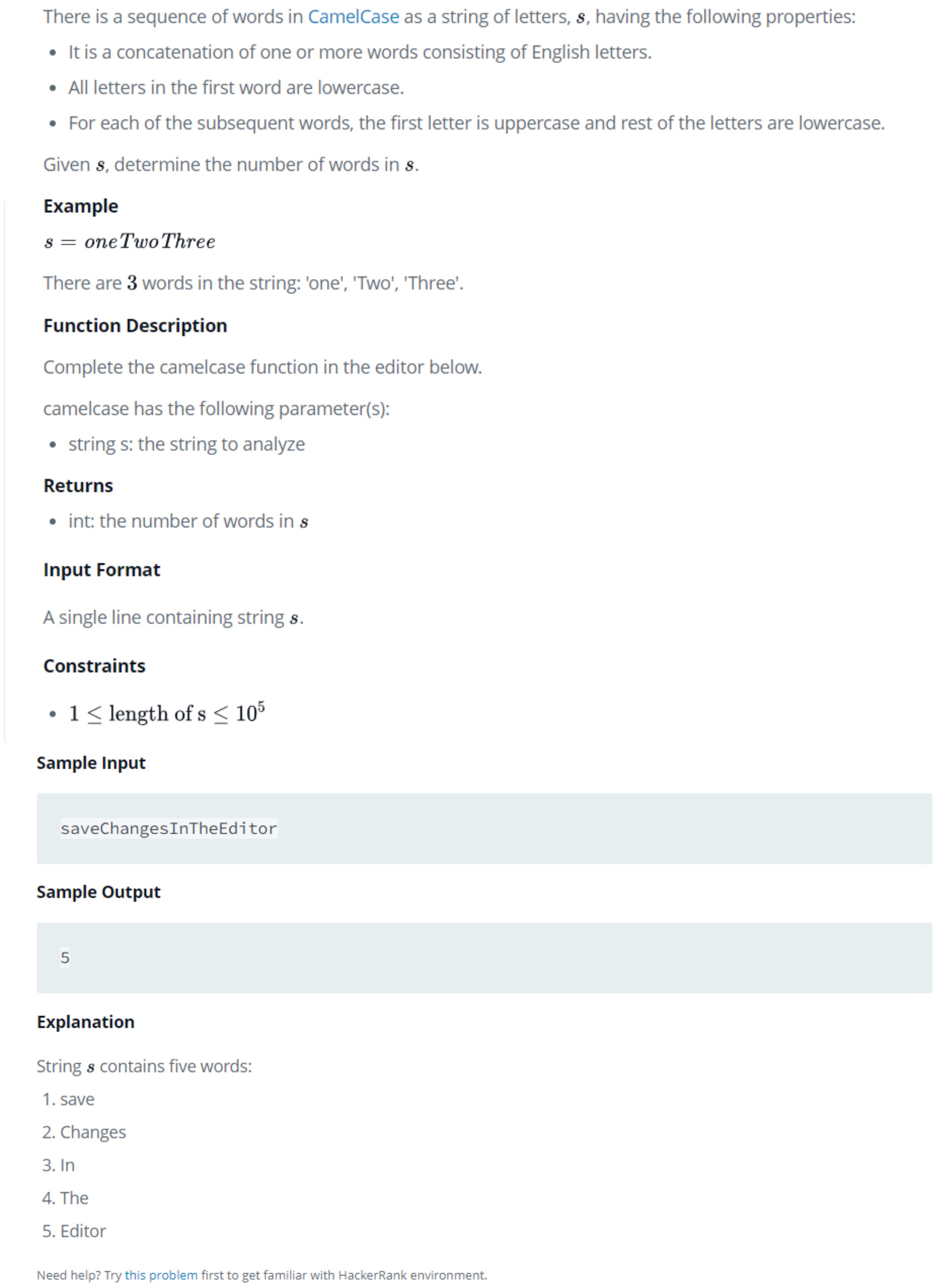
Nama : Ujang Herlan
Nim : 20220040028
Kelas : TI22F
Keterangan : Perbaikan Nilai
Matkul : Algoritma Struktur Data

1. Algorithms
   Soal pada HackerRank yaitu CamelCase pada java8 soal nya sebagai berikut:



There is a sequence of words in CamelCase as a string of letters, s, having the following properties:

- It is a concatenation of one or more words consisting of English letters.
- All letters in the first word are lowercase.
- For each of the subsequent words, the first letter is uppercase and rest of the letters are lowercase.

Given s, determine the number of words in s.

Example

$s = oneTwoThree$

There are 3 words in the string: 'one', 'Two', 'Three'.

Function Description

Complete the camelcase function in the editor below.

camelcase has the following parameter(s):

- string s: the string to analyze

Returns

- int: the number of words in s

Input Format

A single line containing string s.

Constraints

- $1 \leq \text{length of s} \leq 10^5$

Sample Input

```
saveChangesInTheEditor
```

Sample Output

```
5
```

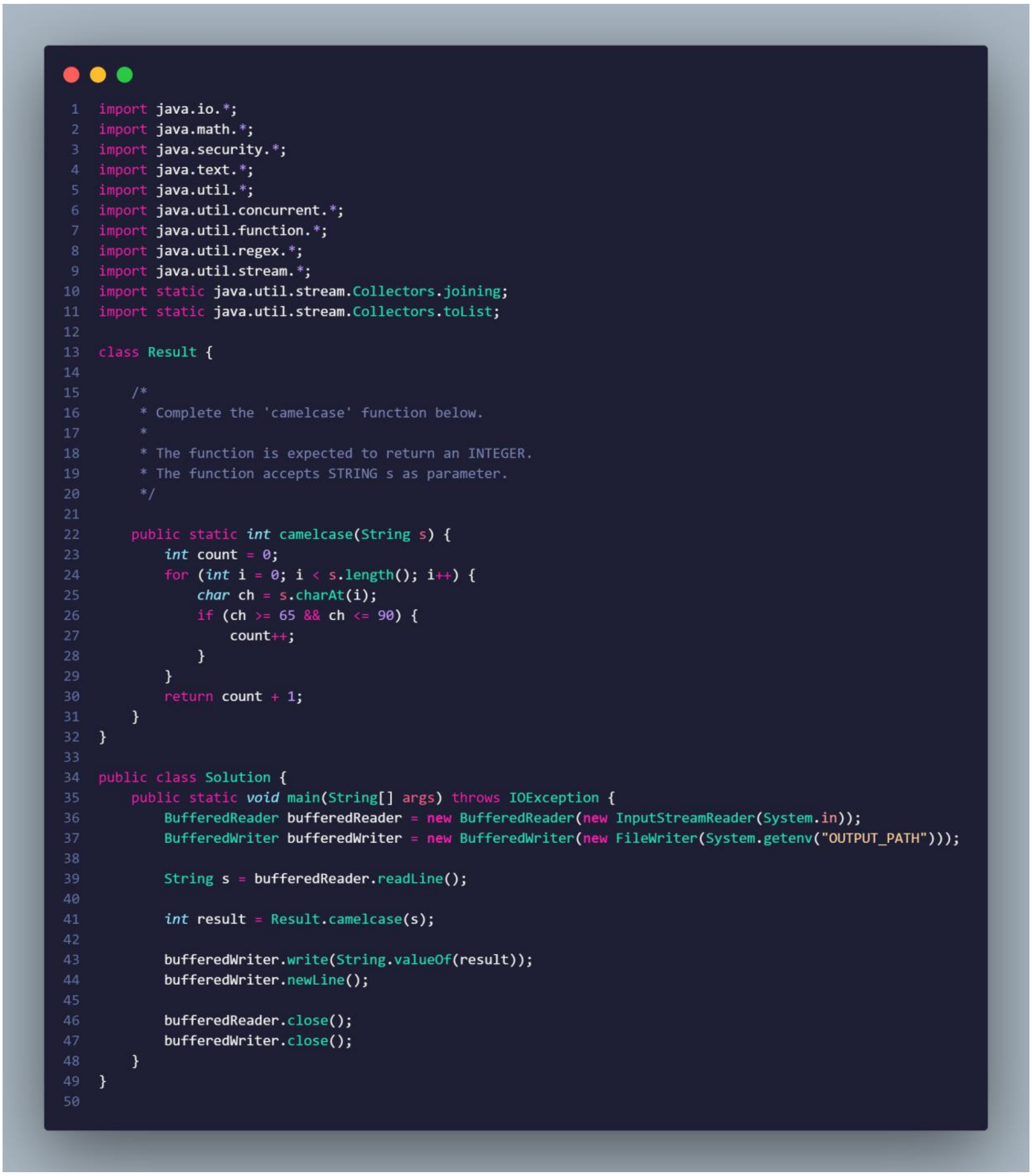
Explanation

String s contains five words:

1. save
2. Changes
3. In
4. The
5. Editor

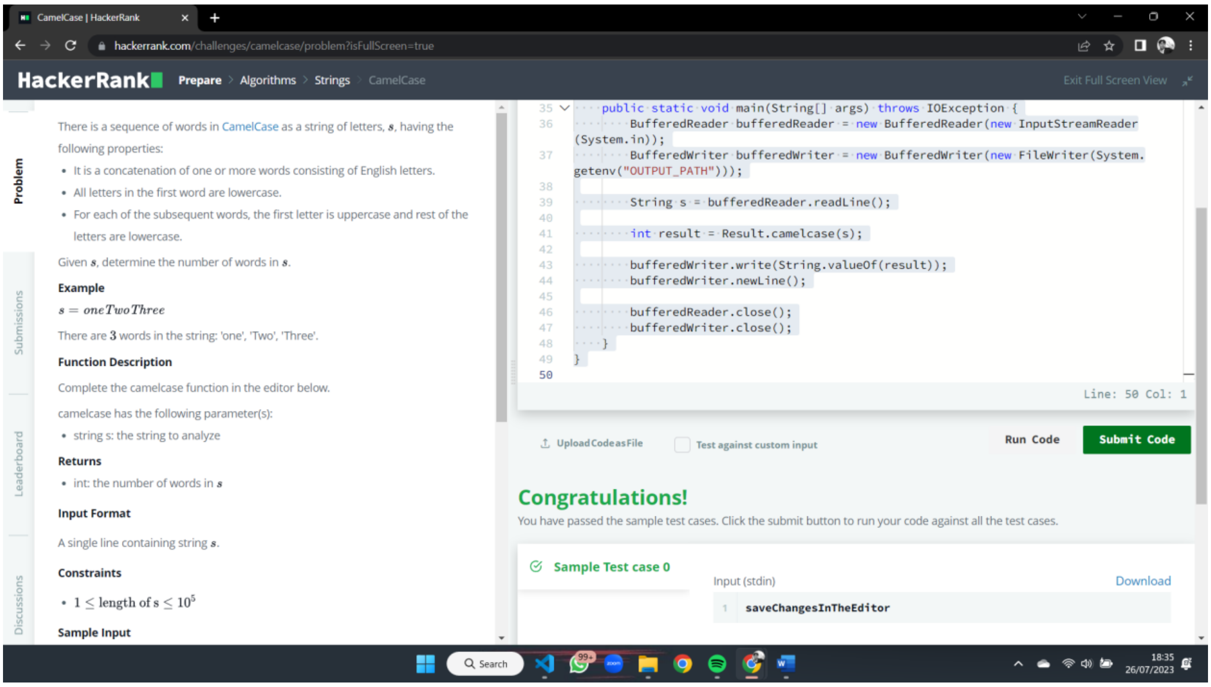
Need help? Try this problem first to get familiar with HackerRank environment.

Berikut Problem Solving yang saya lakukan pada masalah tersebut:



```java
import java.io.*;
import java.math.*;
import java.security.*;
import java.text.*;
import java.util.*;
import java.util.concurrent.*;
import java.util.function.*;
import java.util.regex.*;
import java.util.stream.*;
import static java.util.stream.Collectors.joining;
import static java.util.stream.Collectors.toList;

class Result {

    /*
     * Complete the 'camelcase' function below.
     *
     * The function is expected to return an INTEGER.
     * The function accepts STRING s as parameter.
     */

    public static int camelcase(String s) {
        int count = 0;
        for (int i = 0; i < s.length(); i++) {
            char ch = s.charAt(i);
            if (ch >= 65 && ch <= 90) {
                count++;
            }
        }
        return count + 1;
    }
}

public class Solution {
    public static void main(String[] args) throws IOException {
        BufferedReader bufferedReader = new BufferedReader(new InputStreamReader(System.in));
        BufferedWriter bufferedWriter = new BufferedWriter(new FileWriter(System.getenv("OUTPUT_PATH")));

        String s = bufferedReader.readLine();

        int result = Result.camelcase(s);

        bufferedWriter.write(String.valueOf(result));
        bufferedWriter.newLine();

        bufferedReader.close();
        bufferedWriter.close();
    }
}
```

Untuk hasil pada HackerRank:



There is a sequence of words in CamelCase as a string of letters, s, having the following properties:

- It is a concatenation of one or more words consisting of English letters.
- All letters in the first word are lowercase.
- For each of the subsequent words, the first letter is uppercase and rest of the letters are lowercase.

Given s, determine the number of words in s.

Example

$s = oneTwoThree$

There are 3 words in the string: 'one', 'Two', 'Three'.

Function Description

Complete the camelcase function in the editor below.

camelcase has the following parameter(s):

- string s: the string to analyze

Returns

- int: the number of words in s

Input Format

A single line containing string s.

Constraints

- $1 \leq \text{length of s} \leq 10^5$

Sample Input

```java
    public static void main(String[] args) throws IOException {
        BufferedReader bufferedReader = new BufferedReader(new InputStreamReader(System.in));
        BufferedWriter bufferedWriter = new BufferedWriter(new FileWriter(System.getenv("OUTPUT_PATH")));

        String s = bufferedReader.readLine();

        int result = Result.camelcase(s);

        bufferedWriter.write(String.valueOf(result));
        bufferedWriter.newLine();

        bufferedReader.close();
        bufferedWriter.close();
    }
}
```

Line: 50 Col: 1

☐ Test against custom input

Congratulations!

You have passed the sample test cases. Click the submit button to run your code against all the test cases.

✓ Sample Test case 0

Input (stdin)

```
saveChangesInTheEditor
```

2. Data Structures
   Soal pada HackeeRank pada point QHEAP1 pada java8 soalnya sebagai berikut:

This question is designed to help you get a better understanding of basic heap operations.

There are 3 types of query:

- "$1\ v$" - Add an element v to the heap.
- "$2\ v$" - Delete the element v from the heap.
- "3" - Print the minimum of all the elements in the heap.

NOTE: It is guaranteed that the element to be deleted will be there in the heap. Also, at any instant, only distinct elements will be in the heap.

Input Format

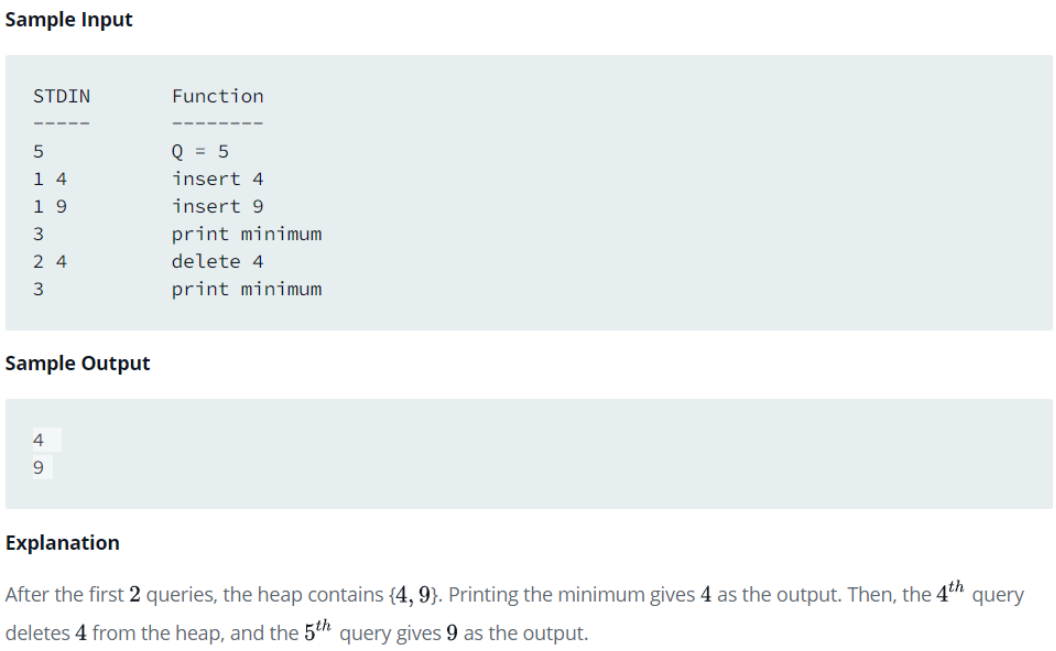
The first line contains the number of queries, Q.
Each of the next Q lines contains one of the 3 types of query.

Constraints

$1 \leq Q \leq 10^5$
$-10^9 \leq v \leq 10^9$

Output Format

For each query of type 3, print the minimum value on a single line.



Sample Input

```
STDIN       Function
-----       --------
5           Q = 5
1 4         insert 4
1 9         insert 9
3           print minimum
2 4         delete 4
3           print minimum
```

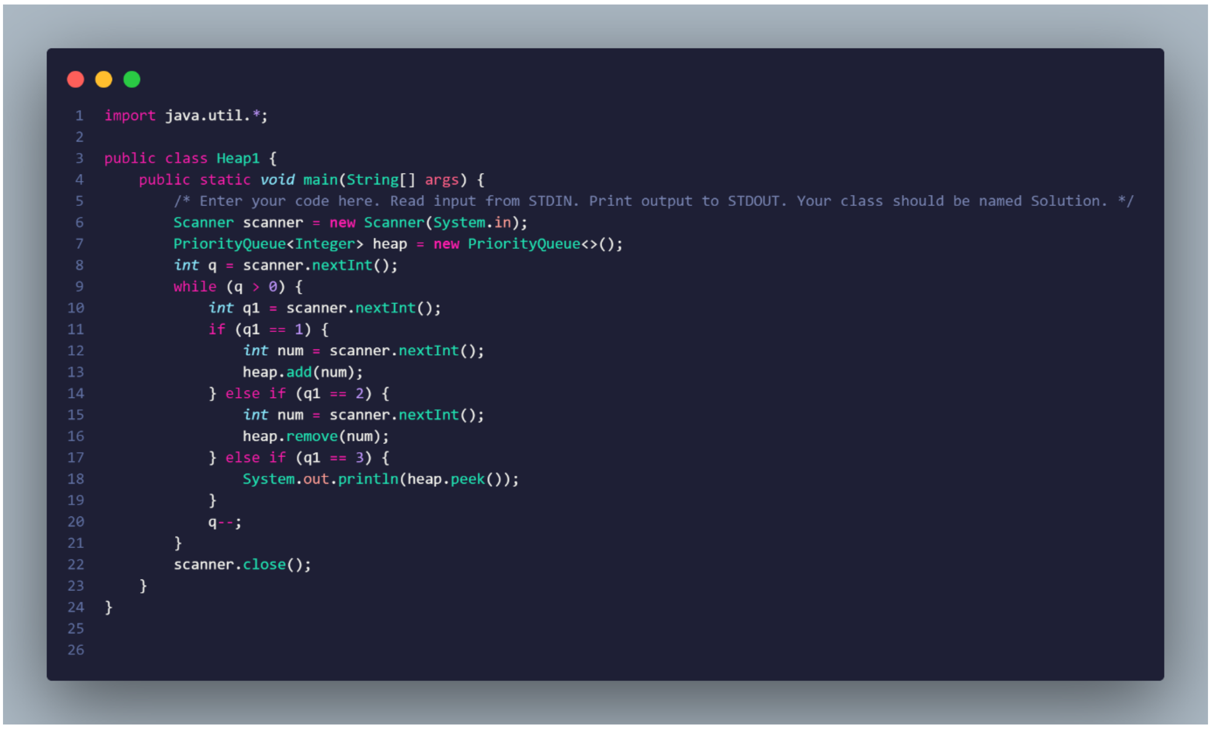
Sample Output

```
4
9
```

Explanation

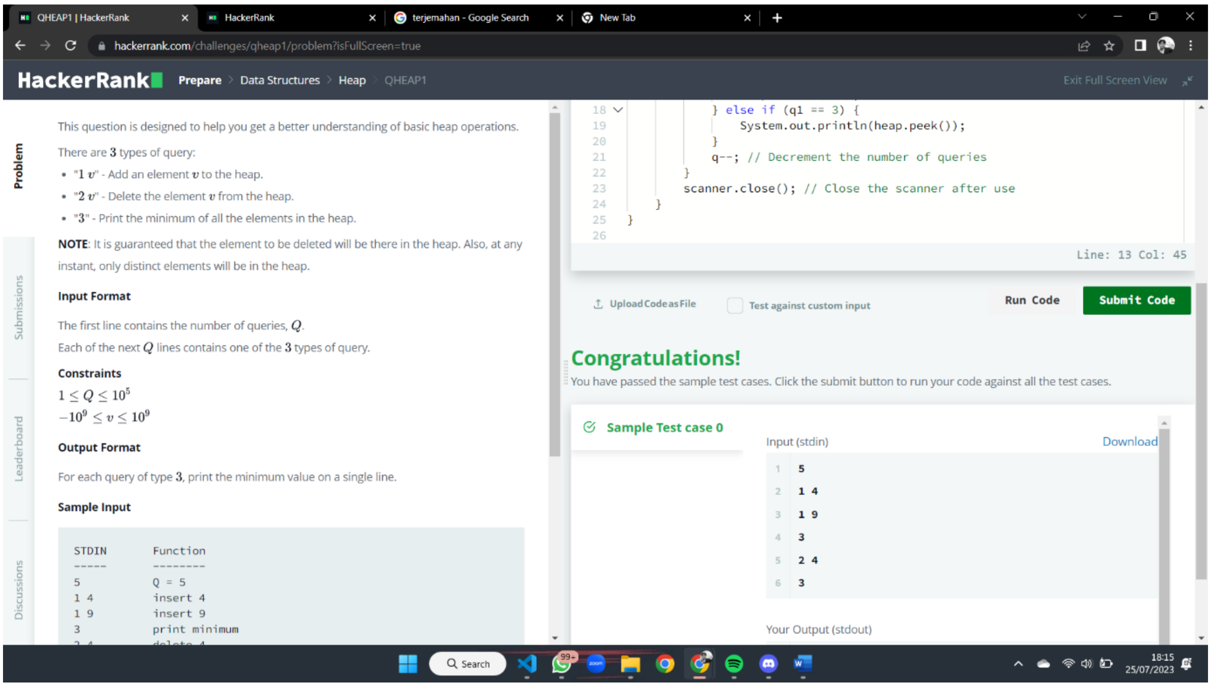
After the first 2 queries, the heap contains $\{4, 9\}$. Printing the minimum gives 4 as the output. Then, the 4^{th} query deletes 4 from the heap, and the 5^{th} query gives 9 as the output.

Berikut Kode Problem Solving terhadap soal tersebut:



```java
import java.util.*;

public class Heap1 {
    public static void main(String[] args) {
        /* Enter your code here. Read input from STDIN. Print output to STDOUT. Your class should be named Solution. */
        Scanner scanner = new Scanner(System.in);
        PriorityQueue<Integer> heap = new PriorityQueue<>();
        int q = scanner.nextInt();
        while (q > 0) {
            int q1 = scanner.nextInt();
            if (q1 == 1) {
                int num = scanner.nextInt();
                heap.add(num);
            } else if (q1 == 2) {
                int num = scanner.nextInt();
                heap.remove(num);
            } else if (q1 == 3) {
                System.out.println(heap.peek());
            }
            q--;
        }
        scanner.close();
    }
}
```

Hasil problem solving pada HackRank:


QHEAP1 | HackerRank
HackerRank
terjemahan - Google Search
New Tab
hackerrank.com/challenges/qheap1/problem?isFullScreen=true
HackerRank
Prepare > Data Structures > Heap > QHEAP1
Exit Full Screen View
Problem
Submissions
Leaderboard
Discussions

This question is designed to help you get a better understanding of basic heap operations.

There are 3 types of query:

- "1 v" - Add an element v to the heap.
- "2 v" - Delete the element v from the heap.
- "3" - Print the minimum of all the elements in the heap.

NOTE: It is guaranteed that the element to be deleted will be there in the heap. Also, at any instant, only distinct elements will be in the heap.

Input Format

The first line contains the number of queries, Q.
Each of the next Q lines contains one of the 3 types of query.

Constraints

$1 \leq Q \leq 10^5$
$-10^9 \leq v \leq 10^9$

Output Format

For each query of type 3, print the minimum value on a single line.

Sample Input

```
STDIN       Function
-----       --------
5           Q = 5
1 4         insert 4
1 9         insert 9
3           print minimum
2 4         delete 4
```

```
            } else if (q1 == 3) {
                System.out.println(heap.peek());
            }
            q--; // Decrement the number of queries
        }
        scanner.close(); // Close the scanner after use
    }
}
```

Line: 13 Col: 45

Congratulations!

You have passed the sample test cases. Click the submit button to run your code against all the test cases.

✓ Sample Test case 0

Input (stdin)

```
5
1 4
1 9
3
2 4
3
```

Your Output (stdout)