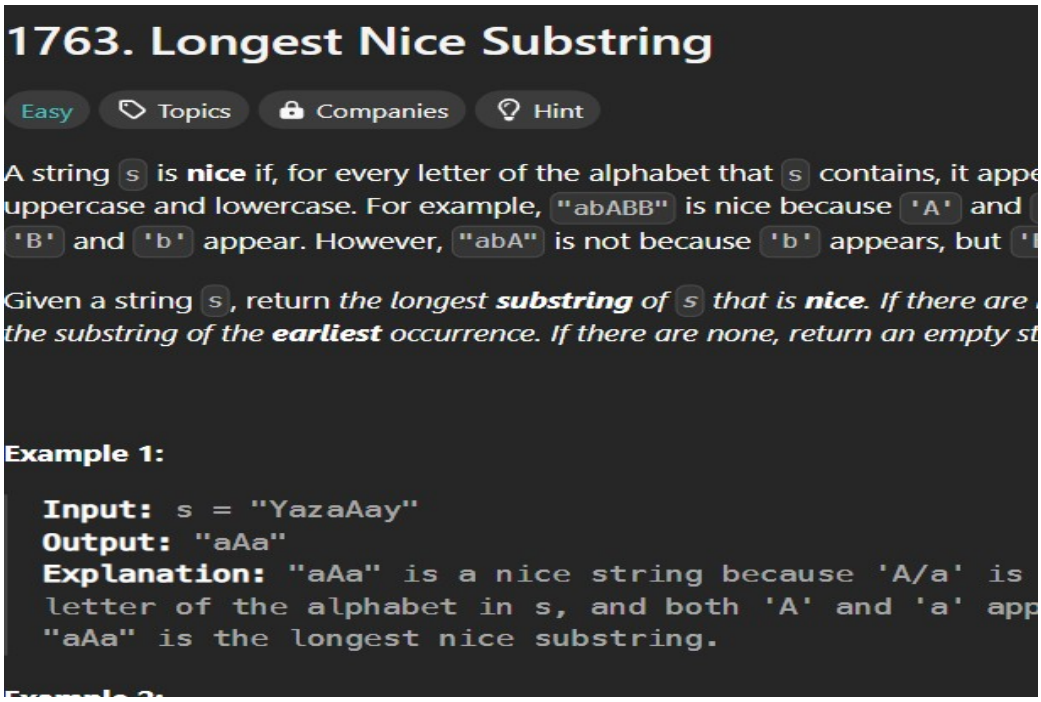
# Assignment-1

**Student Name: Tanvi Chhabra** **UID: 22BCS15679**
**Branch: CSE** **Section: 22BCS_IOT_605 B**
**Semester: 6th** **DOP: 05-02-25**
**Subject: Advanced Programming** **Subject Code: 22CSH-351**

## 1.Question:



1763. Longest Nice Substring

Easy | Topics | Companies | Hint

A string `s` is **nice** if, for every letter of the alphabet that `s` contains, it appe
uppercase and lowercase. For example, `"abABB"` is nice because `'A'` and
`'B'` and `'b'` appear. However, `"abA"` is not because `'b'` appears, but `'

Given a string `s`, return *the longest* ***substring*** *of* `s` *that is* ***nice****. If there are*
*the substring of the* ***earliest*** *occurrence. If there are none, return an empty st*

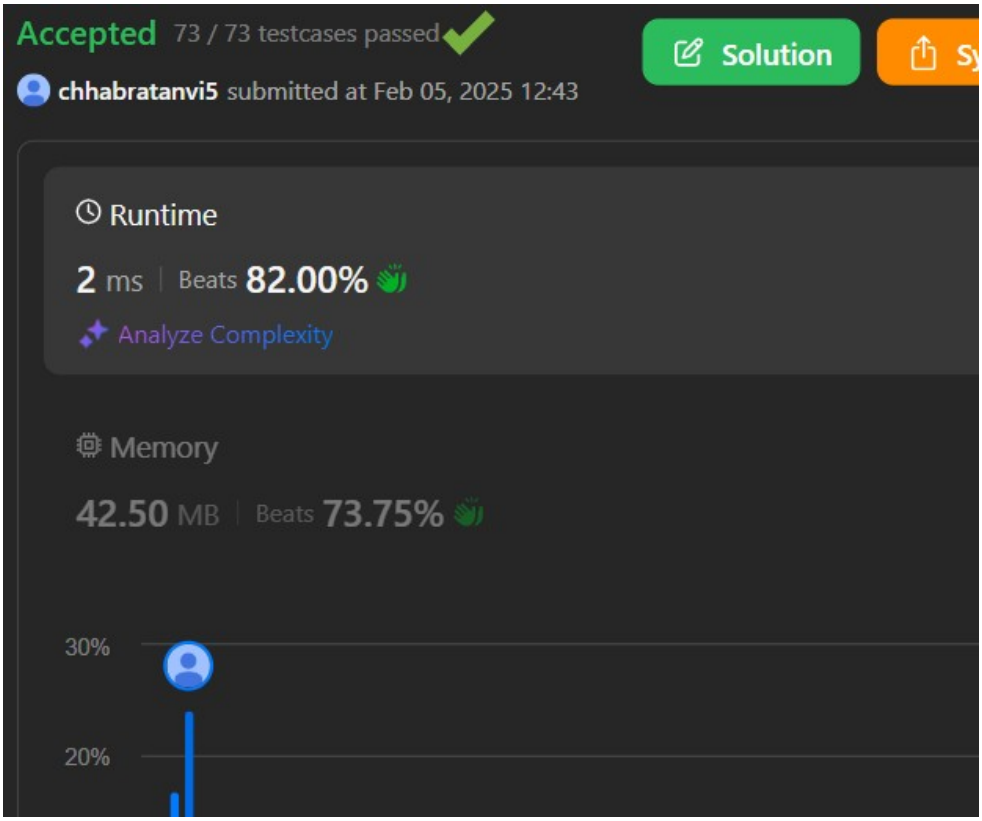
**Example 1:**

```
Input: s = "YazaAay"
Output: "aAa"
Explanation: "aAa" is a nice string because 'A/a' is
letter of the alphabet in s, and both 'A' and 'a' app
"aAa" is the longest nice substring.
```

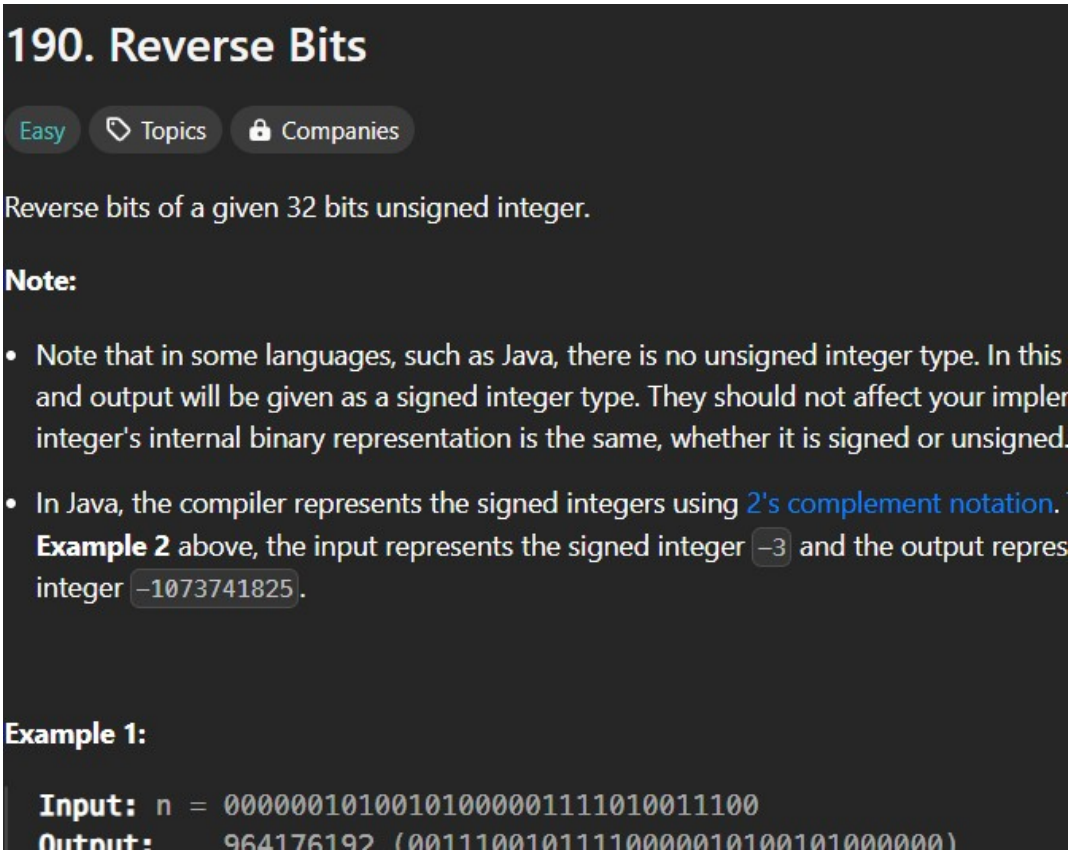
## Code:

```
class Solution {
    public String longestNiceSubstring(String s) {
        if (s.length() < 2) return "";
        char[] arr = s.toCharArray();
        Set<Character> set = new HashSet<>();
        for (char c: arr) set.add(c);
        for (int i = 0; i < arr.length; i++) {
            char c = arr[i];
            if (set.contains(Character.toUpperCase(c)) && set.contains(Character.toLowerCase(c)))
continue;
            String sub1 = longestNiceSubstring(s.substring(0, i));
            String sub2 = longestNiceSubstring(s.substring(i+1));
            return sub1.length() >= sub2.length() ? sub1 : sub2;
        }
        return s;
    }
}
```

**Output:**


Accepted 73 / 73 testcases passed
Solution
chhabratanvi5 submitted at Feb 05, 2025 12:43
Runtime
2 ms | Beats 82.00%
Analyze Complexity
Memory
42.50 MB | Beats 73.75%
30%
20%

## 2.Question:



190. Reverse Bits

Easy Topics Companies

Reverse bits of a given 32 bits unsigned integer.

**Note:**

- Note that in some languages, such as Java, there is no unsigned integer type. In this and output will be given as a signed integer type. They should not affect your impler integer's internal binary representation is the same, whether it is signed or unsigned.
- In Java, the compiler represents the signed integers using 2's complement notation. **Example 2** above, the input represents the signed integer `-3` and the output repres integer `-1073741825`.

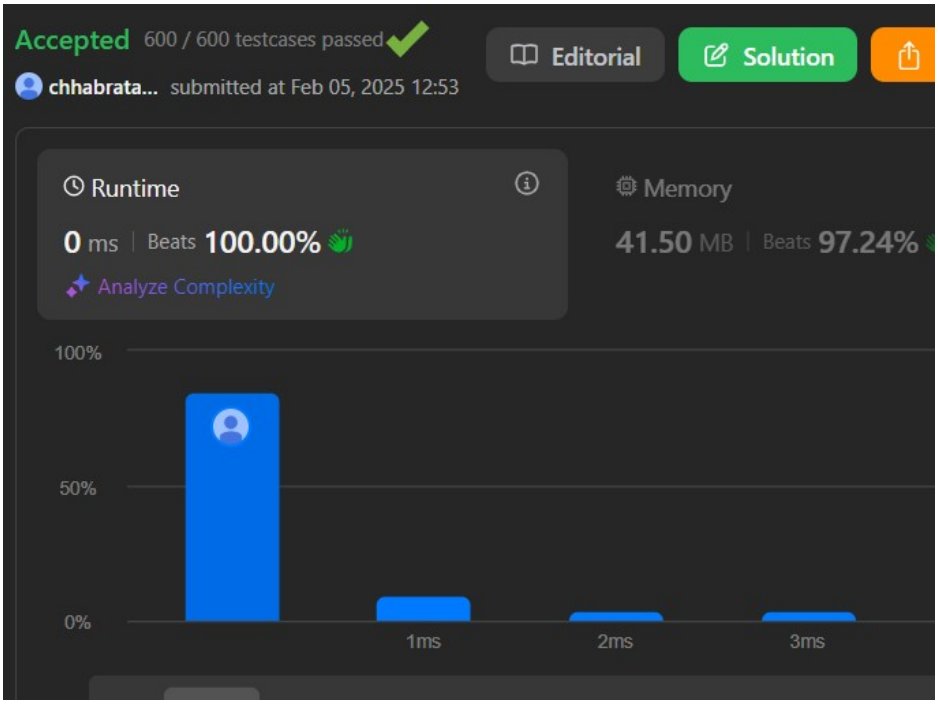
**Example 1:**

```
Input: n = 00000010100101000001111010011100
Output:    964176192 (00111001011110000010100101000000)
```

## Code:

```
public class Solution {
    // you need treat n as an unsigned value
    public int reverseBits(int n) {
        int result=0;
        for(int i=0;i<32;i++){
            result<<=1;
            result|=(n&1);
            n>>>=1;
        }
        return result;
    }
}
```

**Output:**

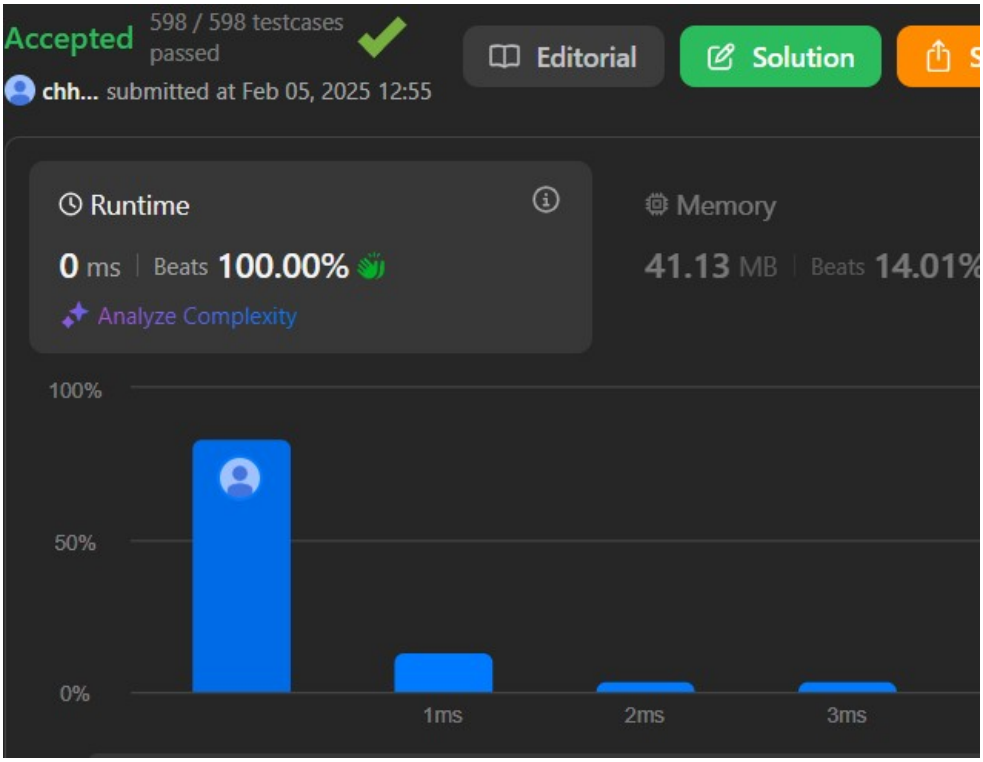

Accepted 600 / 600 testcases passed
chhabrata... submitted at Feb 05, 2025 12:53
Editorial
Solution
Runtime
0 ms | Beats 100.00%
Analyze Complexity
Memory
41.50 MB | Beats 97.24%
100%
50%
0%
1ms
2ms
3ms

**3.Question:**

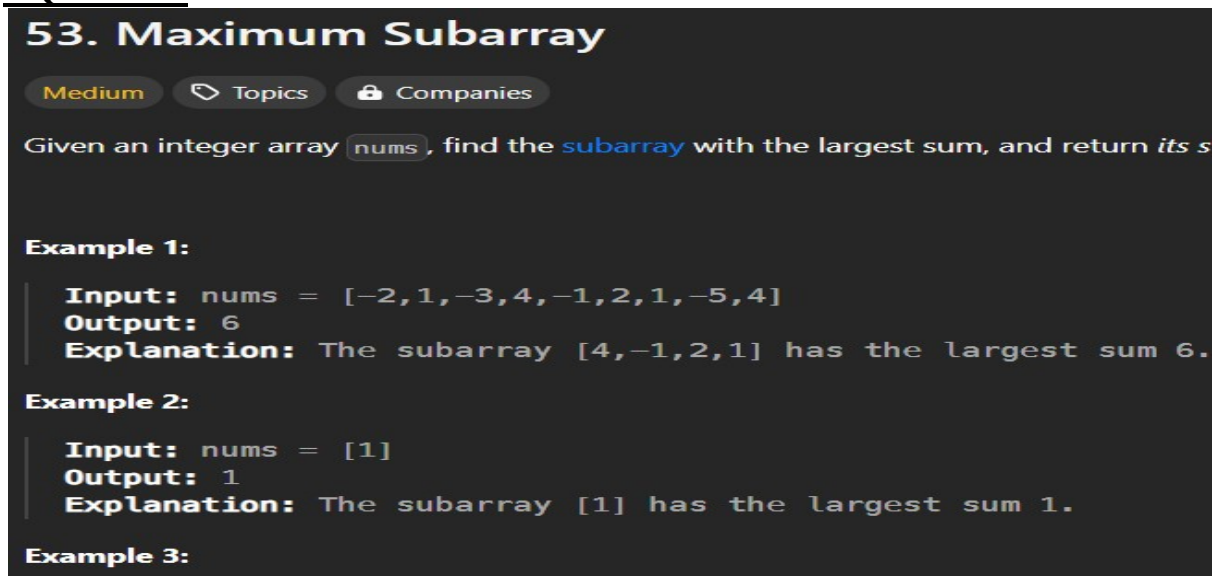

**Code:**

```
public static int hammingWeight(int n) {
        int ones = 0;
        while(n!=0) {
                ones = ones + (n & 1);
                n = n>>>1;
        }
        return ones;
}
```

**Output:**



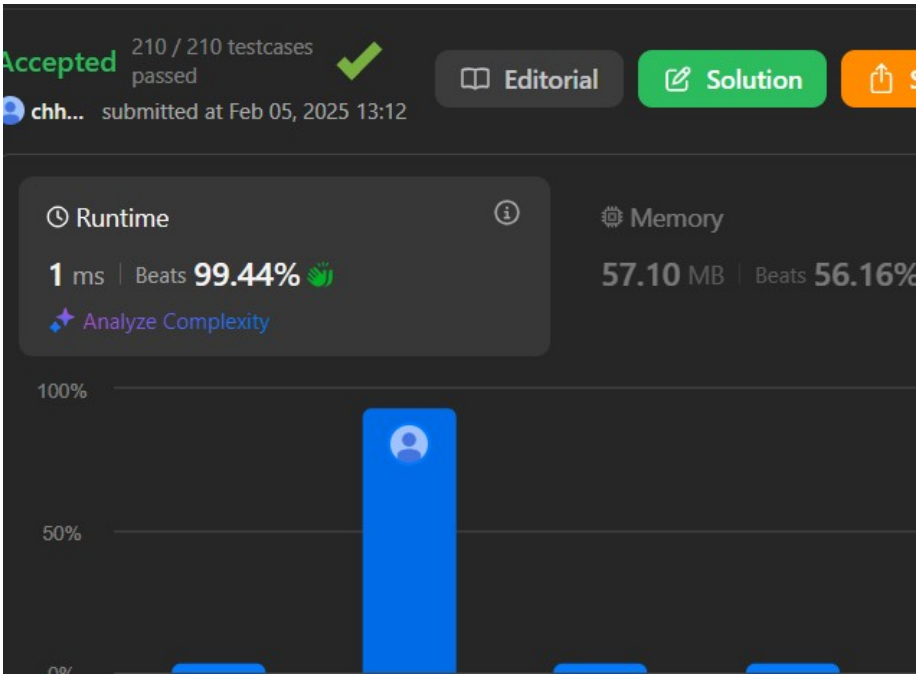
**4.Question:**



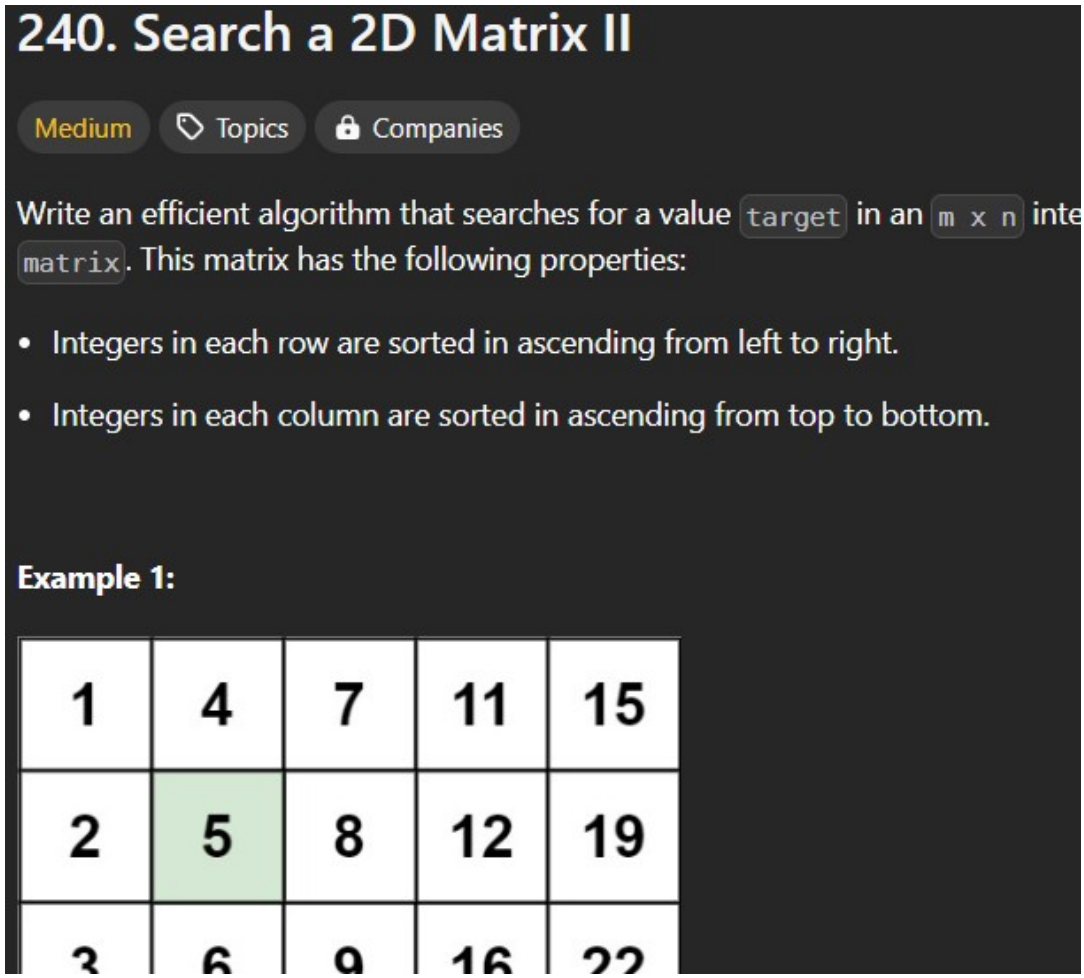

## Code:

```
class Solution {
    public int maxSubArray(int[] nums) {
        int maxSum = Integer.MIN_VALUE;
        int currentSum = 0;

        for (int i = 0; i < nums.length; i++) {
            currentSum += nums[i];

            if (currentSum > maxSum) {
                maxSum = currentSum;
            }

            if (currentSum < 0) {
                currentSum = 0;
            }
        }

        return maxSum;
    }
}
```

**Output:**


Accepted
210 / 210 testcases passed
chh... submitted at Feb 05, 2025 13:12
Editorial
Solution
Runtime
1 ms | Beats 99.44%
Analyze Complexity
Memory
57.10 MB | Beats 56.16%
100%
50%

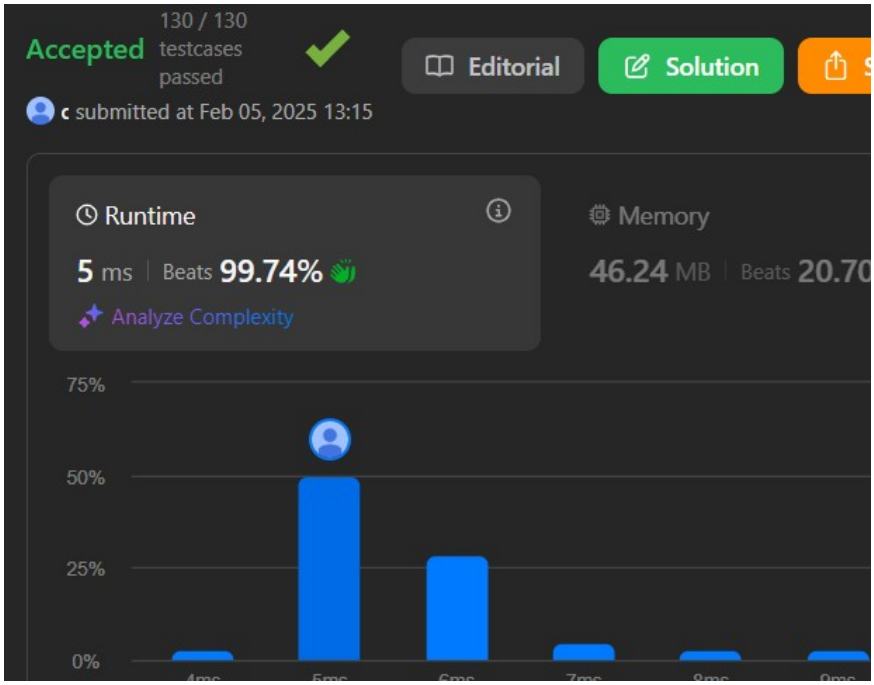
**5.Question:**



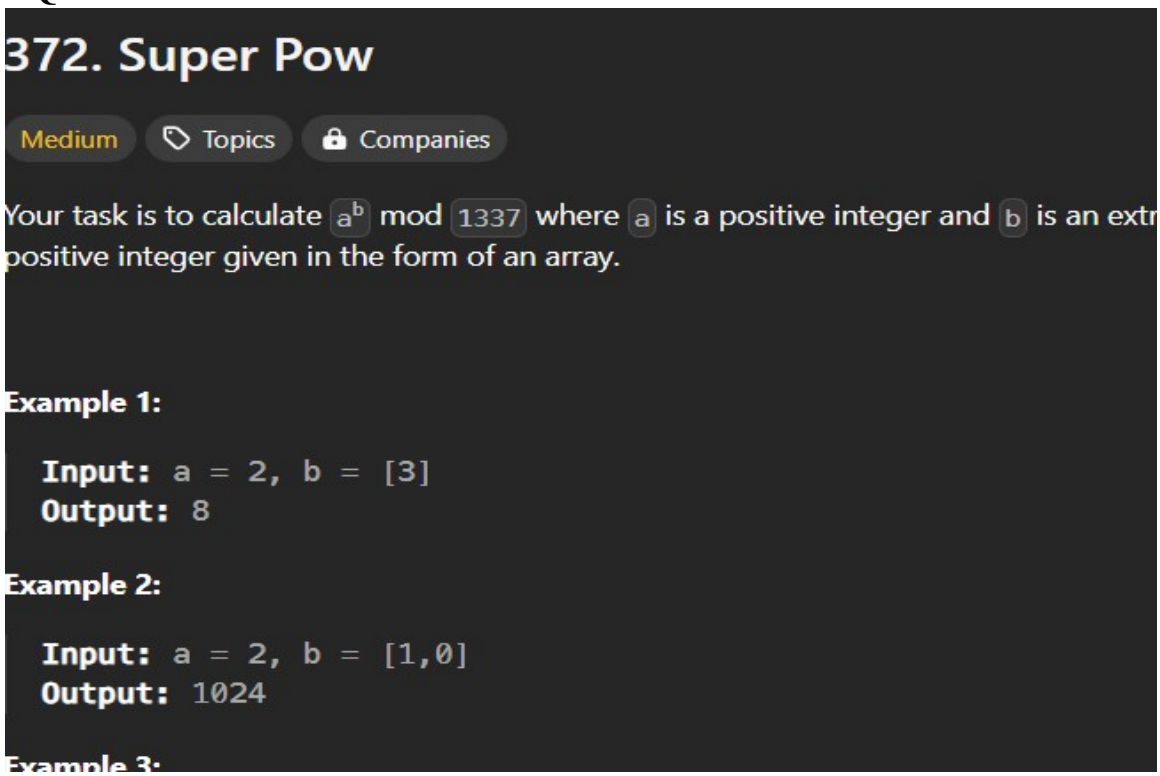

**Code:**

```
public class Solution {
    public boolean searchMatrix(int[][] matrix, int target) {
        if(matrix == null || matrix.length < 1 || matrix[0].length <1) {
            return false;
        }
        int col = matrix[0].length-1;
        int row = 0;
        while(col >= 0 && row <= matrix.length-1) {
            if(target == matrix[row][col]) {
                return true;
            } else if(target < matrix[row][col]) {
                col--;
            } else if(target > matrix[row][col]) {
                row++;
            }
        }
        return false;
    }
}
```

**Output:**


Accepted
130 / 130 testcases passed
Editorial
Solution
c submitted at Feb 05, 2025 13:15
Runtime
5 ms | Beats 99.74%
Analyze Complexity
Memory
46.24 MB | Beats 20.70
75%
50%
25%
0%

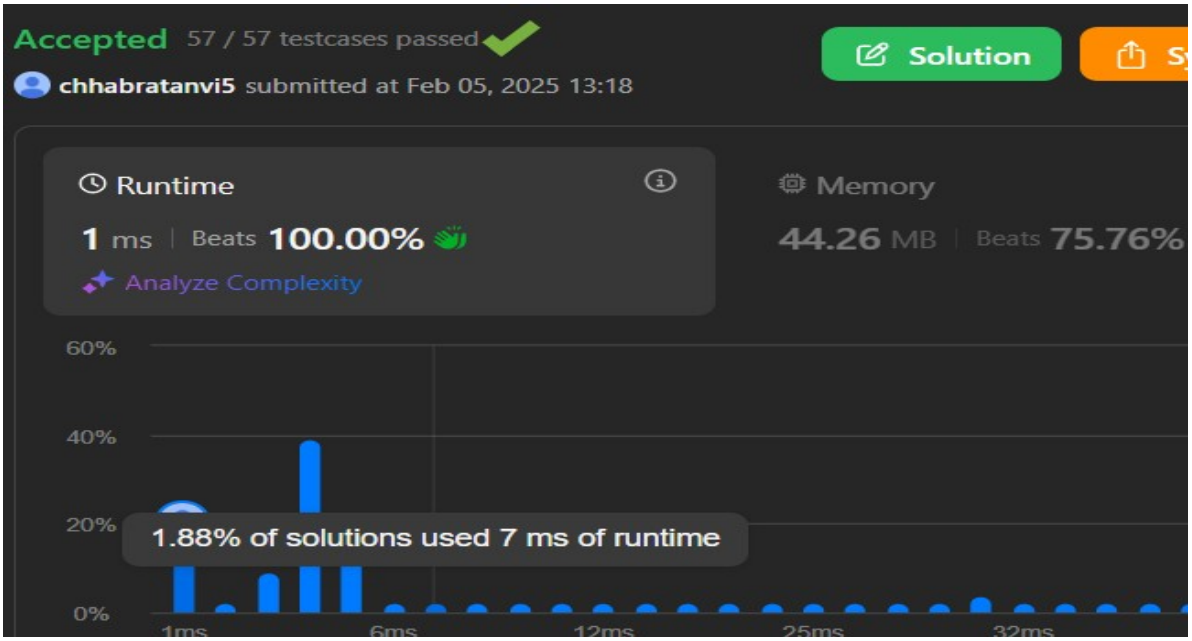
**6.Question:**




**Code:**

```
public int superPow(int a, int[] b) {
    if (a % 1337 == 0) return 0;
    int p = 0;
    for (int i : b) p = (p * 10 + i) % 1140;
    if (p == 0) p += 1440;
    return power(a, p, 1337);
}
public int power(int a, int n, int mod) {
    a %= mod;
    int ret = 1;
    while (n != 0) {
        if ((n & 1) != 0) ret = ret * a % mod;
        a = a * a % mod;
        n >>= 1;
    }
    return ret;
}
```

**Output:**


Accepted 57 / 57 testcases passed
chhabratanvi5 submitted at Feb 05, 2025 13:18
Solution
Runtime
1 ms | Beats 100.00%
Analyze Complexity
Memory
44.26 MB | Beats 75.76%
60%
40%
20%
0%
1.88% of solutions used 7 ms of runtime
1ms
6ms
12ms
25ms
32ms