EXPERIMENT -1.3

NAME- Jatin UID- 20BCS5951

BRANCH- BE- CSE SECTION- DM 605 B

SEMESTER- 6th DATE- 06/03/2023

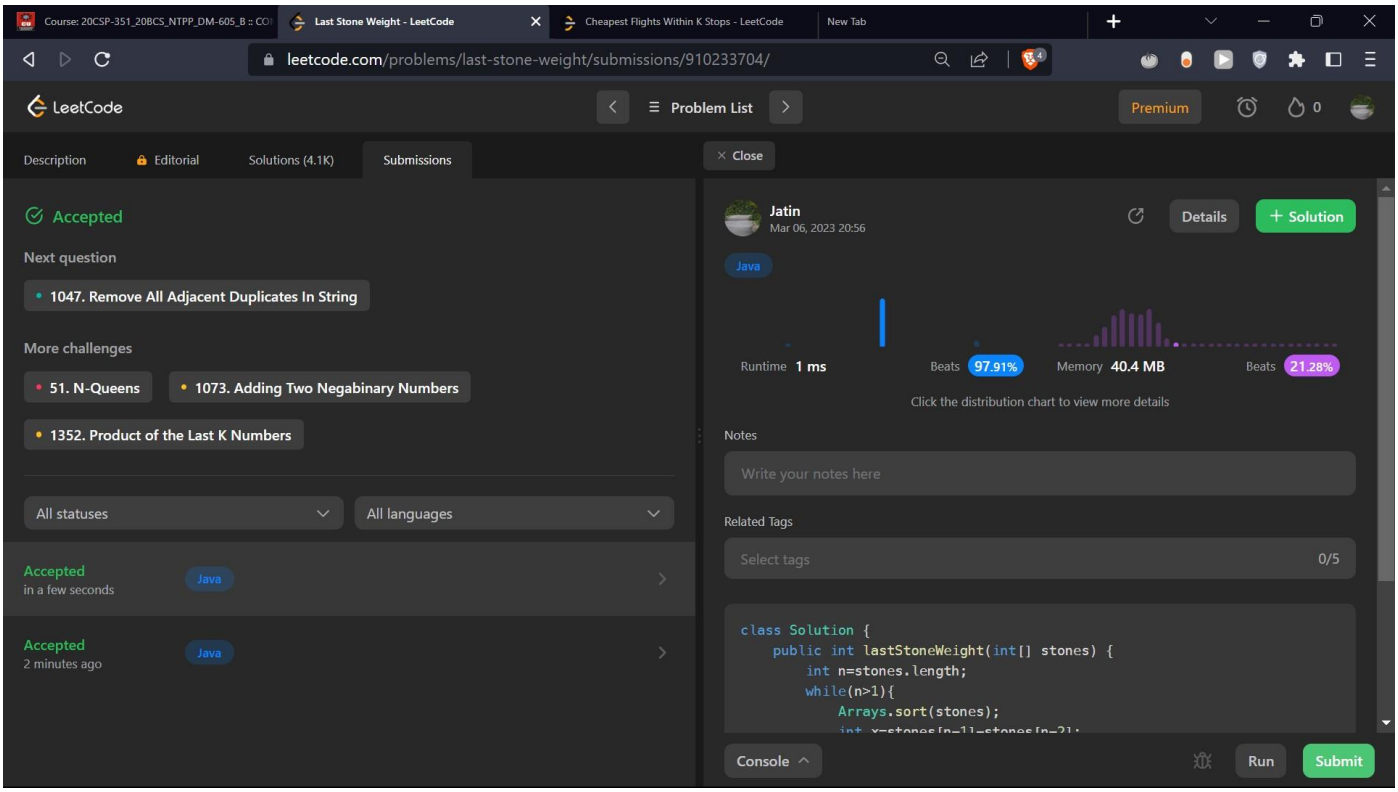
SUBJECT- CC Lab- 2 SUBJECT CODE- 20CSP-351

Q1. Last Stone Weight

CODE:

```
class Solution {
    public int lastStoneWeight(int[] stones)
{         int n=stones.length;
while(n>1){
            Arrays.sort(stones);
int x=stones[n-1]-stones[n-2];
n--;
            stones[n-1]=x;
        }
        return stones[0];
    }
}
```

Output :

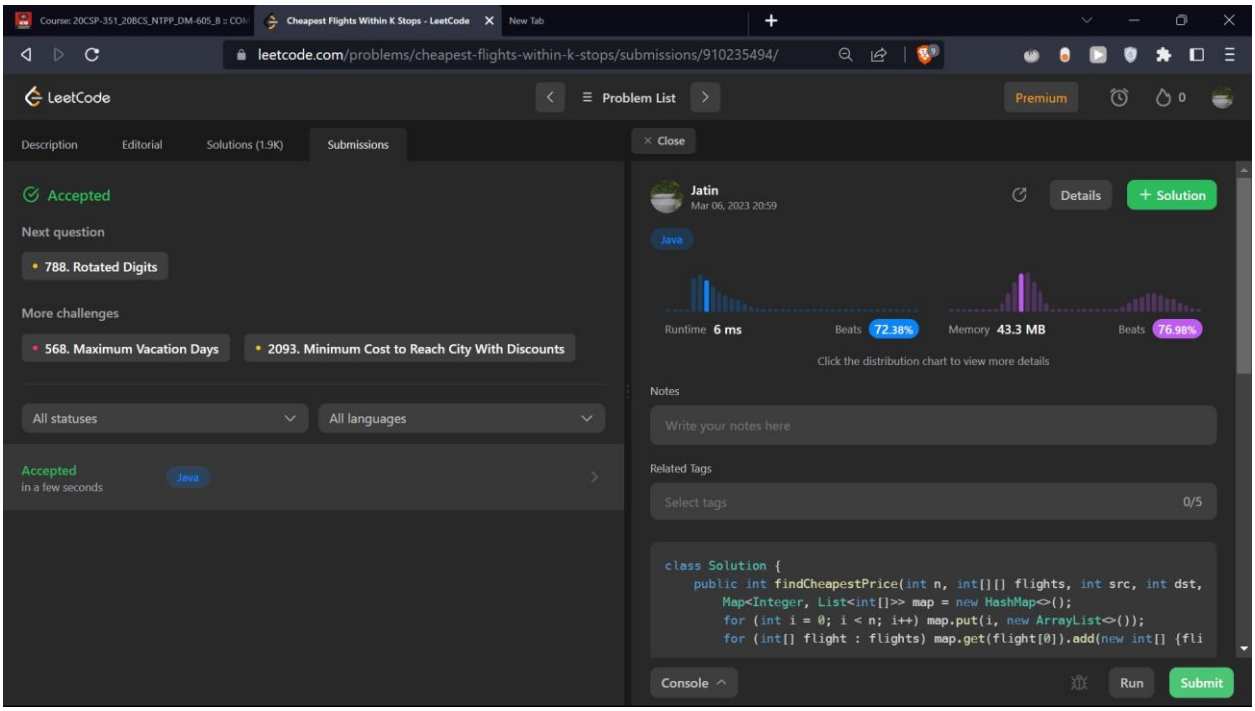


Q2 Cheapest Flights Within K Stops

Code :

```
class Solution {
    public int findCheapestPrice(int n, int[][] flights, int src, int dst, int k)
{          Map<Integer, List<int[]>> map = new HashMap<>();          for (int i = 0;
i < n; i++) map.put(i, new ArrayList<>());
        for (int[] flight : flights) map.get(flight[0]).add(new int[] {flight[1],
flight[2]});

        LinkedList<int[]> bfs = new LinkedList<>();
bfs.add(new int[] {src, 0});

        int[] visited = new int[n];
        Arrays.fill(visited, Integer.MAX_VALUE);

        int res = Integer.MAX_VALUE;
int stops = 0;
         while (stops++ <= k) {
int size = bfs.size();

            while (size-- > 0) {
                int[] dest = bfs.remove();
                 for (int[] route : map.get(dest[0]))
{                    int price = dest[1] + route[1];

                    if (dst == route[0]) {
                        res = Math.min(res, price);
                    } else {
                        if (visited[route[0]] > price) {
visited[route[0]] = price;
                            bfs.add(new int[] {route[0], price});
                        }
                    }
                }
            }
        }

        return res == Integer.MAX_VALUE ? -1 : res;
    }
}
```

Output :


Cheapest Flights Within K Stops - LeetCode
leetcode.com/problems/cheapest-flights-within-k-stops/submissions/910235494/
LeetCode
Problem List
Premium
Description
Editorial
Solutions (1.9K)
Submissions
Close
Accepted
Next question
788. Rotated Digits
More challenges
568. Maximum Vacation Days
2093. Minimum Cost to Reach City With Discounts
All statuses
All languages
Accepted
in a few seconds
Java
Jatin
Mar 06, 2023 20:59
Details
+ Solution
Java
Runtime 6 ms
Beats 72.38%
Memory 43.3 MB
Beats 76.98%
Click the distribution chart to view more details
Notes
Write your notes here
Related Tags
Select tags
0/5
class Solution {
public int findCheapestPrice(int n, int[][] flights, int src, int dst,
Map<Integer, List<int[]>> map = new HashMap<>();
for (int i = 0; i < n; i++) map.put(i, new ArrayList<>());
for (int[] flight : flights) map.get(flight[0]).add(new int[] {fli
Console
Run
Submit