Experiment- 2.3

Student Name: Jatin UID: 20BCS5951

Branch: BE CSE Section/Group: NTPP_605-B

Semester: 6th Subject Code: 20CSP-351

Subject Name: CC Lab

Aim: To demonstrate the concept of Divide and Conquer Problem1:

Count and say https://leetcode.com/problems/count-and-say/

Given a binary tree, determine if it is height-balanced.

Given a positive integer n, return the nth term of the count-and-say sequence.

Example 1:

Input: n = 1

Output: "1"

Explanation: This is the base case.

Example 2:

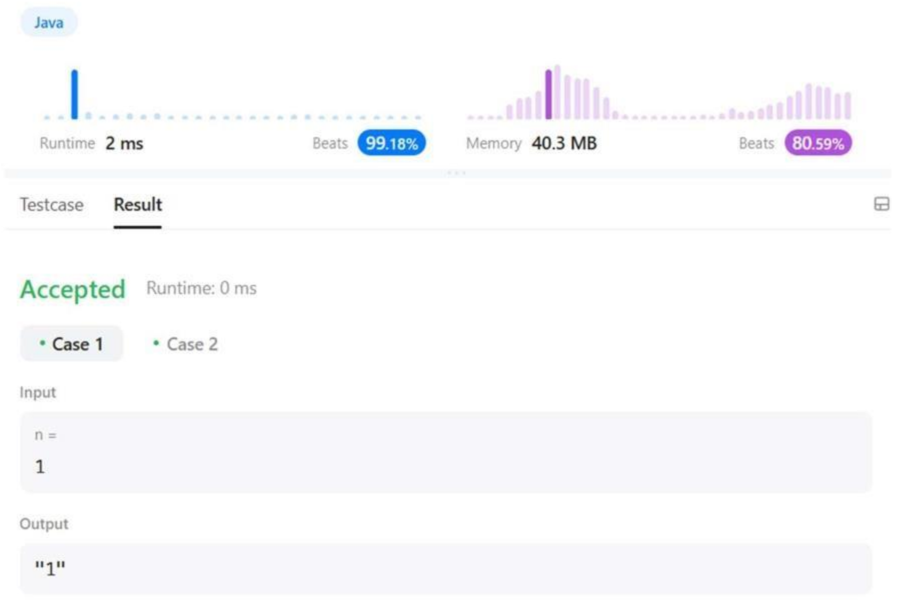
Input: n = 4

Output: "1211"

Code:

```
class Solution {     public String Count(String s){          int cnt = 1;          char ch =
s.charAt(0);
        StringBuilder curr = new StringBuilder();          for(int
i=1;i<s.length();i++){              if(s.charAt(i)==ch){                  cnt++;
}              else{                  curr.append(cnt);
curr.append(ch);                  ch = s.charAt(i);                  cnt = 1;
            }          }          curr.append(cnt);
curr.append(ch);          return curr.toString();
    }
    public String countAndSay(int n) {          String s
= "1";          for(int i=1;i<n;i++){              s =
Count(s);          }          return s;
    }
}
```

Output:



Java

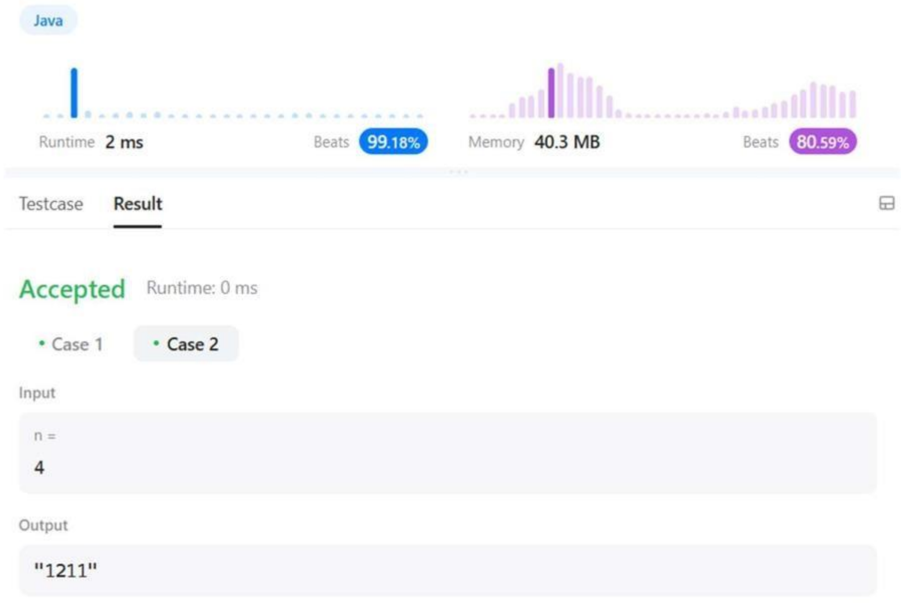
Accepted Runtime: 0 ms

Input

```
n =
1
```

Output

```
"1"
```



Java

Accepted Runtime: 0 ms

Input

```
n =
4
```

Output

```
"1211"
```

Problem 2: Water and Jug Problem https://leetcode.com/problems/water-and-jugproblem/

Probem2: Water and Jug Problem

You are given two jugs with capacities jug1Capacity and jug2Capacity liters. There is an infinite amount of water supply available. Determine whether it is possible to measure exactly targetCapacity liters using these two jugs.

If targetCapacity liters of water are measurable, you must have targetCapacity liters of water contained within one or both buckets by the end.

Operations allowed:

- Fill any of the jugs with water.
- Empty any of the jugs.
- Pour water from one jug into another till the other jug is completely full, or the first jug itself is empty.

Example 1:

Input: jug1Capacity = 3, jug2Capacity = 5, targetCapacity = 4

Output: true

Explanation: The famous Die Hard example

Example 2:

Input: jug1Capacity = 2, jug2Capacity = 6, targetCapacity = 5

Output: false

Example 3:

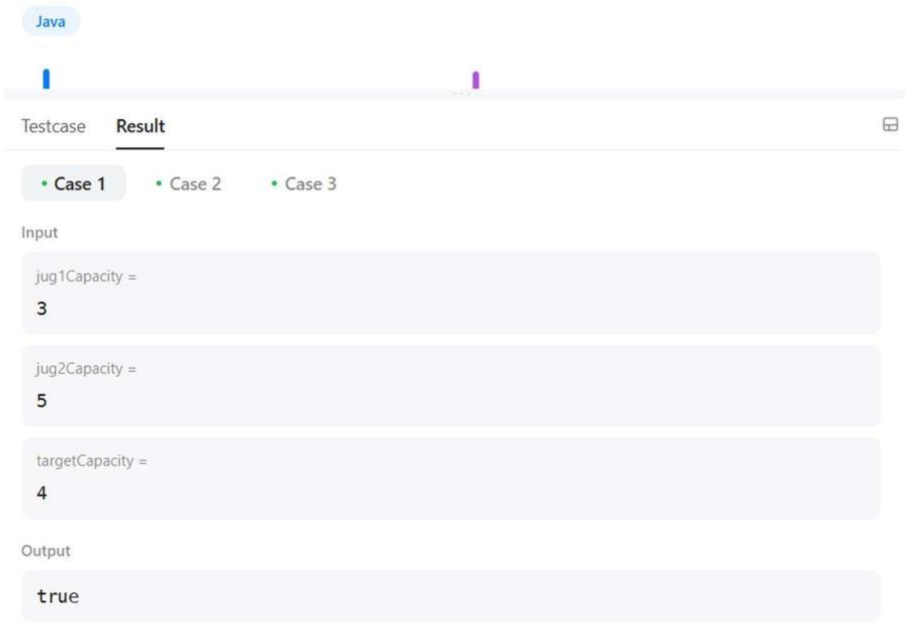
Input: jug1Capacity = 1, jug2Capacity = 2, targetCapacity = 3

Output: true

Code:

```
class Solution {
  public boolean canMeasureWater(int jug1Capacity, int jug2Capacity, int targetCapacity)
  {
    return targetCapacity == 0 || jug1Capacity + jug2Capacity >= targetCapacity && targetCapacity %
gcd(jug1Capacity, jug2Capacity) == 0;
  }     private int gcd(int a,
int b)
  {      return b == 0 ? a : gcd(b, a
% b);
  }
}
```

Output:



Input

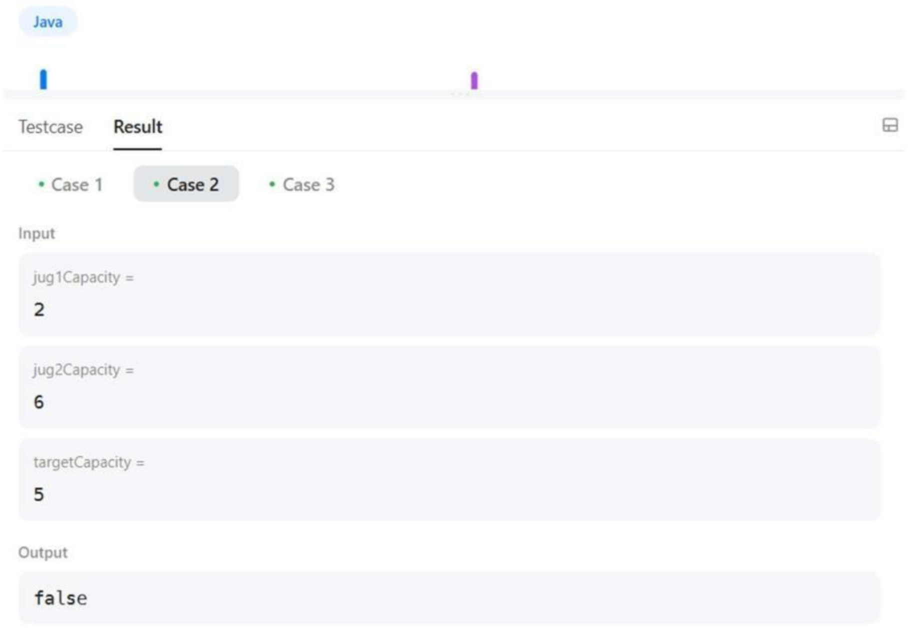
jug1Capacity =
3

jug2Capacity =
5

targetCapacity =
4

Output

```
true
```



Input

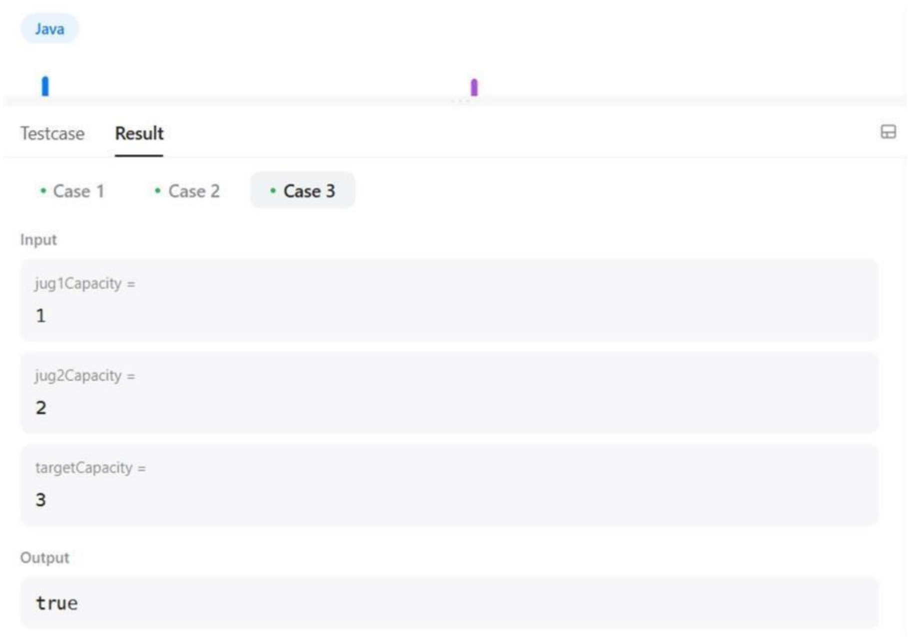
jug1Capacity =
2

jug2Capacity =
6

targetCapacity =
5

Output

```
false
```



Input

jug1Capacity =

```
1
```

jug2Capacity =

```
2
```

targetCapacity =

```
3
```

Output

```
true
```