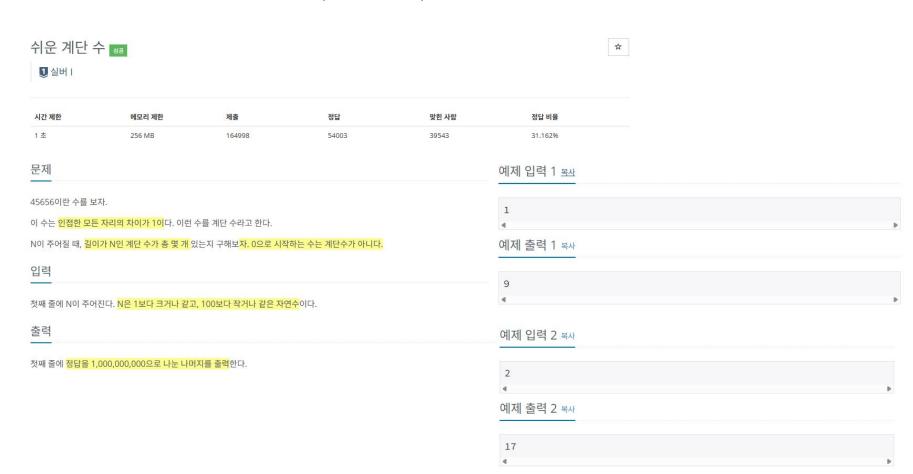
Algorithm Study 04.07.2024 21 \$ 7

#10844. 쉬운 계단수/Silver 1

Q. #10844. 쉬운 계단수(Silver 1)



Hint

1. 문제 유형: DP

● 부분 문제 O

뒷자리: j

DP[i][j] = 길이 i이고 마지막 자리 숫자가 j 인 총 <u>계단수 갯수</u>

 0
 1
 2
 3
 4
 5
 6
 7
 8
 9

 0
 0
 0
 0
 0
 0
 0
 0
 0
 0
 0
 0

 1
 0
 1
 1
 1
 1
 1
 1
 1
 1
 1
 1

 2
 1
 2
 2
 2
 2
 2
 2
 2
 2
 1

 3
 2
 4
 4
 4
 4
 4
 4
 4
 4
 4

 1.

My Solution

```
4 n = int(input())
 5 # dp [총 길이 i ][마지막 자리수가 j] = 인 개수
 6 Mod = 1000000000
7 # 1. 초기화
 8 #(1) 0을 시작하는 수는 계단수가 아닌다 - dp[1][0]
9 # (2) N=1 인 계단수
10 dp = [[0]*10 for _ in range(n+1)]
11 dp[1][0] = 0
12 for j in range(1,10):
     dp[1][j]= 1
14
15 # (3) 점화식 N>=2
16 for i in range(2,n+1):
      for j in range(10):
        if j== 0 :
             dp[i][j] = dp[i-1][j+1]
20
       elif j==9:
              dp[i][j] = dp[i-1][j-1]
22
        else:
             dp[i][j] = dp[i-1][j-1] + dp[i-1][j+1]
24
25 print(sum(dp[n])%Mod)
```

1. DP 테이블 정의: 2차원 배열(1차도 가능)

<u>`DP[i:1~N][j:0-9]` **: 길이가 i 고, 뒷자리가 J 인 계단수의 개수**</u>

i : 길이 j : 마지막 숫자

2. 초기화

- (1) **dp [1][0] = 0** # 0 으로 시작하는 계단수가 아님
- (2) dp[1][1-9]=1

3. 점화식

(1) j 가 1~ 8일때 dp[i][j] = d[i-1][j-1] +dp[i-1][j+1]

(2) i == 0 일 때 → 십의 자리 숫자가 1만 가능

dp[i][0] = dp[i-1][1]

(3) j == 9 일때 → 십의 자리 숫자가 8일 때만 가능

dp[i][9] = dp[i-1][8]

다른 Solution

1차원 DP

```
dp = [1] * 10
dp[0] = 0
n = int(input())
for i in range(n - 1):
   temp = [0] * 10
   temp[0] = dp[1]
   temp[9] = dp[8]
    for j in range(1, 9):
       temp[j] = dp[j - 1] + dp[j + 1]
    dp = temp
print(sum(dp) % 1000000000)
```

Assignment

[백준#1562.계단수] 골드1

- 문제: <u>https://www.acmicpc.net/problem/1562</u>

#비트마스킹,#DP

참고자료: https://wikidocs.net/206816



☆



시간 제한	메모리 제한	제출	정답	맞힌 사람	정답 비율
2 초	128 MB	15386	8195	6222	52.751%

문제

45656이란 수를 보자.

이 수는 인접한 모든 자리의 차이가 1이다<mark>. 이런 수를 계단 수라</mark>고 한다.

N이 주어질 때, <mark>길이가 N이면서 0부터 9까지 숫자가 모두 등장하는 계단 수</mark>가 총 몇 개 있는지 구하는 프로그램을 작성하시오. 0으로 시작하는 수는 계단수가 아니다.