**Question\_2**

**Reference:** https://www.geeksforgeeks.org/egg-dropping-puzzle-dp-11/

**Time Complexity**: O (N \* log K)

**Auxiliary Space:** O (N)

Follow the below steps to solve the problem:

1. Create an array dp of size R+1 and an integer m
2. Run a for loop from m equal to zero till dp[R] < k
3. Run a nested for loop from x equal to n till x is greater than zero
4. Set dp[x] equal to 1 + dp[x-1]
5. Return m

**Explanation:**

When the stand at test spot n, there can be two cases (1) The signal cannot be received (2) The signal can be received.

If the signal is not received after the ‘nth’ test spot, then we only need to check for test spots lower than ‘n’ with remaining calls as some test sports should exist lower than ‘n’ in which the signal will be received, so the problem reduces to x-1 floors and R-1 calls.

If the signal isn’t received after the ‘nth’ test floor, then we only need to check for test spots higher than ‘x’; so the problem reduces to ‘k-x’ test spots and n calls.

Since we need to minimize the number of calls in the worst case, we take a maximum of two cases. We consider the max of the above two cases for every test spot and choose the test spot that yields the minimum number of calls.

**Reference:**

* <https://leetcode.com/problems/super-egg-drop/>