PROBLEM THREE SOLUTION

The problem can be refined as: where, 'n' is the number of rungs  
 'k' is the number of jars, such that, k>0

First part solution:

* One attempt is to do a binary search with the first jar, but that’s not efficient as if we drop first sphere from N/2’th floor and it breaks, we need to use linear search using the second jar starting from the 1st rung up- which has worst case of N/2-1.
* For example, let n=10 and k=2, then a good place to start would be the middle, at rung number 5, and if the jar breaks there, we have only one jar left and got to do a linear search on 5 rungs, which grabs time complexity of . So it's apparent that starting point needs to be dependent on the number of jars.
* Suppose, for sake of simplicity, if n is a perfect square, then a good approach would be to drop the first jar from heights that are multiples of . If we reach the top and jar doesn’t break, we are done to drop jar √n times.
* Likewise, if jar breaks at height of m√n , highest safest rung should be between

and . At this point, we shall start dropping the second jar starting from   
th rung, going up by one rung each time, which would atmost take

steps.

* Therefore, worst case scenario would be highest rung in at most steps. The efficient method would be to drop the first jar from heights that are multiple of. If isn’t a perfect square, then same strategy can be used though dividing the latter in multiples of and still obtaining an algorithm that runs in time complexity
* So , the idea is to throw the first jar from rung for increasing values of . So in total we can never do more than throws.

Second part solution:

1. We infer by induction that .
2. We begin by dropping the first jar from heights that are multiples of [n^(k-1)/k]. So , the first jar can be dropped atmost at times that reduces the possible rungs down to an interval of length atmost n^(k-1)/k , then when we apply the strategy for jars recursively.
3. By induction, we can conclude that , this uses atmost   
    = drops. By adding in the drops made using the first jar, we get a bound of finishing the induction.