Qiyao Zhou

Z5379852

Question 1

1.1:

By analyzing the rules of the game, it is clear that the ultimate goal of the game is to change the original stack sequence into a sequence in non-decreasing order by splitting the stack over time.

To analyze the feasibility of the game one can discuss the worst case scenario, i.e. the original N stacks are arranged in decreasing order, $h_1 > h_2 \ldots > h_n$, so that the game can only be won by traversing and splitting from h_{n-1}, and the maximum height that can be obtained cannot exceed h_n, so that the minimum number of steps can only be achieved by splitting it into several blocks of height h_n (for example: h_n=2, h_{n-1}=4, then h_{n-1} is split into two stacks of height 2, while if h_n=2, h_{n-1}=5, then it needs to be split twice to obtain two stacks of height 2. If h_n=2, h_{n-1}=5, then two stacks of height 2 and one stack of height 1 are needed, in which case the stack before that can only be divided into several stacks of height 1). In the worst case, when h_n=1, all the previous stacks can only be divided into several stacks of height 1, and the game is solved.

1.2:

Thinking through question 1.1 and analyzing the problem, to obtain the minimum number of moves to win, it is necessary to keep the height of the stack leaning back as large as possible, while the stack before it is partitioned by it, with the following algorithm:

1. The stack sequence is traversed from back to front, and the height of the stack traversed is denoted as h_i.
2. Record the height h_n as the minimum value when traversing the last stack, and then count the value of h_i divided by the minimum value in steps each time you traverse forward, updating the minimum value to the remainder if the remainder is not 0, or unchanged if it is 0.
3. The number of steps obtained when the traversal is complete is the minimum number of moves required to win.

Such an algorithm requires a total of 1 iteration of the stack sequence of length n. Each iteration of the stack requires an integer division operation and a judgment operation, with a total time complexity of O(n).