COMP3121 Assignment 2 - Q4

Demiao Chen z5289988

July 11, 2021

Answer

We can have a greedy method to solve the problem. Loop array $A[i]$ from $i = 1$ to $i = n - 1$. We check the i^{th} stack if

$$A[i] < A[i+1],$$

if so, the i^{th} stack and the $(i+1)^{th}$ stack are strictly increasing, so from the 1^{st} to the $(i+1)^{th}$ stack are strictly increasing in number of blocks, then go to the next index of array A ;
if not, we have to satisfy that if there can be x $(0 < x \leq i)$ blocks moved from i^{th} stack to $(i+1)^{th}$ stack, such that the two following conditions hold:

$$A[i+1] + x \geq A[i] - x + 1 \tag{1}$$

$$A[i] - x \geq A[i-1] + 1 \tag{2}$$

that is to add a minimum integer x to $A[i+1]$ that makes $A[i+1]$ strictly bigger than $A[i]$, but also keep $A[i]$ strictly bigger than $A[i-1]$ after subtract x.
Transform the two conditions (1) (2) we get the following

$$x \geq \frac{A[i] - A[i+i] + 1}{2} \tag{3}$$

$$x \leq A[i] - A[i-1] - 1 \tag{4}$$

As x must be an integer, we can get x by

$$x = \ulcorner \frac{A[i] - A[i+i] + 1}{2} \urcorner \tag{5}$$

that is get the celling of the right hand of (3).
Then we check if the x from the result of (5) satisfies

$$x \leq i \tag{6}$$

and (4).
If either (4) or (6) x does not satisfy, we conclude that such movements does not exist and end the algorithm.
Else we add the x to $A[i+1]$ and go to the next index of array A.

If we finish the loop of array A, we conclude that such movements exists. Algorithm finished.

Time complexity: compute x from (5) then do comparisons (4) and (6), or just one $A[i] < A[i+1]$ comparison take time $O(1)$, and the length of array A is n so we need to do $n-1$ times such calculations. Therefore, the time complexity of the algorithm is $O((n-1) * 1) = O(n)$