Question 1

Zeal Liang

Z5325156

(a)

Let's take an example when n = 3 the grid is of the form:

$$\mathbf{5}_{\mathbf{T(1,1)}}$$

$$\mathbf{5}_{\mathbf{T(2,1)}} \quad \mathbf{8}_{\mathbf{T(2,2)}}$$

$$\mathbf{10}_{\mathbf{T(3,1)}} \quad \mathbf{2}_{\mathbf{T(3,2)}} \quad \mathbf{0}_{\mathbf{T(3,3)}}$$

where all $\mathbf{T(i, j) \geq 0}$.

According to the greedy algorithm given by the problem, if we just choose the direction in which the next entry consists of the largest value, we end up with a sum of 15:

$$\mathbf{T(1,1) \rightarrow T(2,2) \rightarrow T(3,2)},$$

$$5 + 8 + 2 = 15.$$

However, the correct answer is actually:

$$\mathbf{T(1,1) \rightarrow T(2,1) \rightarrow T(3,1)},$$

$$5 + 5 + 10 = 20.$$

Therefore, this counterexample can prove that the greedy algorithm provided does not produce the correct answer.

(b)

Solution:

We recursively work from the bottom of the triangle upwards. Except for the last row, the maximum value of each point in each row is equal to itself plus the maximum value of the corresponding left and right points in the row below. Recursively from the bottom to the top, the topmost one is the answer. In this process we can use a two-dimensional array to store the current optimal solution.

Subproblems:

For $1 \leq j \leq i \leq n$, the jth entry in row i is denoted $\mathrm{T(i,j)}$ and let $\mathrm{T(i,j)}$ be the problem of determining $MaxSum(i,j)$, that is, the sum of the largest numbers that can be encountered on a route from the bottom to the current entry.

Recurrence:

for $i \geq 1$,

$$MaxSum(i,j) = \max\big(T(i-1,j-1), T(i-1,j)\big) + \mathrm{T(i,j)}.$$

Base cases:

$$MaxSum(n,1) = T(n,1) \ \cdots \ MaxSum(n,n) = T(n,n).$$

Because we save the current optimal solution for each time and the sum of the numbers of the triangles is $\boldsymbol{n(n+1)/2}$. So, the time complexity of this algorithm is $\boldsymbol{O(n^2)}$.