Alex Kwong

Drainage Description

The goal of this problem was to find the distance of the longest path that the drainage could take given a height map of r rows and c columns using dynamic programming.

My approach to the solution was to have the given height map and create a distance array of the same size with all entries of 0. I started from the lowest height and looked at the 4 adjacent squares. If all the squares around a square were 0, I set the square to a distance of 1. Then I looked at the next highest height and set its distance to either the max distance adjacent to the square + 1 or I set it to a value of 1 if all the distances adjacent to it had a value of 0. This was repeated until the distance array was completely filled up and the largest value in the distance array was returned as the answer.

To program this, I read in the input of the height map which was stored in a 2D array of size r by c which runs in $\Theta(r*c)$. I then created another 2D array of size r by c to store the distances which runs in $\Theta(r*c)$. In order to sort the heights, I flattened the 2D array into a 1D array of length r*c which runs in $\Theta(r*c)$. I used the built-in sort which runs in $\Theta(r*c \log(r*c))$ and returned a list of indices that would sort the flattened map. The indices of the sorted 1D array could be converted back into the indices of the 2D array using r = i//columns and c = i % columns

Each index was then iterated through in sorted order which runs in $\Theta(r*c)$. The converted indices were used to index into the distance map where distances were filled in as described above. After the distance was calculated and inserted into the 2D distance array, it was compared against the current max value and the current max value was updated accordingly. After each index was iterated through, the max distance was printed out. Sorting the indices takes the longest so finding the distance will take $\Theta(r*c \log(r*c))$. The user specifies that n distances will be found so overall, the algorithm runs in $\Theta(n*r*c \log(r*c))$.