Homework 8

CS 331H

Due Wednesday, March 22

1. Recall that BFS computes shortest paths in $O(E)$ time on an unweighted graph, while Dijkstra takes $O(E + V \log V)$ for weighted graphs with nonnegative edge weights. In this problem, we consider how to speed this up for "small" edge weights, where $1 \leq w(u \to v) < C$ for some integer C.

   (a) First, suppose all edge weights $w(u \to v)$ are in $\{1, 2\}$. Give an $O(E)$ algorithm to find the shortest path distances from a source s.

   (b) Dijkstra's algorithm normally visits vertices in order of increasing $c(u)$, and relaxes every edge out of the vertices it visits. Consider a variant of Dijkstra's algorithm that instead visits vertices in order of increasing $\lfloor c(u) \rfloor$, with ties broken arbitrarily.

   Suppose that $1 \leq w(u \to v)$ for all edges $u \to v$. Show that, on any shortest path $s = u_1 \to u_2 \to u_3 \to \cdots \to u_k$, this "rounded" variant of Dijkstra will visit u_{k-1} before u_k.

   Conclude that, by our lemma in class (namely: after the edges of a shortest $s \rightsquigarrow t$ path have been relaxed in order, $c(t) = c^*(t)$), this "rounded" variant of Dijkstra will correctly compute shortest path distances on graphs with $w(u \to v) \geq 1$.

   (c) Now suppose that $1 \leq w(u \to v) < 2$ for all edges $u \to v$. Give an $O(E)$ algorithm to find the shortest path distances from s.

   Hint: Eha gur "ebhaqrq" inevnag bs Qvwxfgen, ohg engure guna fgber iregvprf va n urnc, xrrc n frcnengr dhrhr sbe rnpu inyhr bs $\lfloor c(u) \rfloor$.

   (d) Extend the above algorithm to $O(EC)$ time when $1 \leq w(u \to v) < C$.

2. You are given a three dimensional object. On the horizontal plane it is an $n \times n$ square, and on the vertical axis each square (x, y) is a square pillar rising to height $h_{x,y} \geq 1$. Adjacent pillars, even ones sharing corners, are fused together.

   You submerse this object into a bucket of water, then carefully lift it out. Water will then drain off the sides, but it cannot drain through pillars. How many units of water will be captured in the object? Give an $O(n^2 \log n)$ algorithm.

As an example, in the following grid 2 units will be captured, all in the center tile:

1	5	9
7	3	6
7	5	2

Hint: Imagine you put the object at the bottom of an empty bucket, then slowly pour water in from the side to raise the water level in the bucket. Can you mark down, for each square, at what height the water floods in? (rot-13) Lbh znl jnag n urnc gb xrrc genpx bs juvpu fdhner vf arkg gb sybbq.