6331 - Algorithms, Spring 2014, CSE, OSU
Homework 8
Instructor: Anastasios Sidiropoulos

Problem 1: Robot navigation. Consider a robot that moves inside a room, represented by a 2-dimensional $n \times n$ square grid. Every location of the grid is indexed by a pair of integers (i, j), where $i, j \in \{1, \ldots, n\}$. Location (i, j) may either be empty, or it might contain an obstacle. The robot is allowed to move only in empty locations.

Apart from its position, the state of the robot also consists of its orientation, which can be either North, South, West, or East. At every step, the robot can either move by one position forward in its current orientation, or it can change its orientation by $90°$. For example, if the robot has orientation North, and is located in position (i, j), then in one step it can do one of the following:

(1) Move to position $(i, j + 1)$, and maintain orientation North.

(2) Stay in position (i, j), and change its orientation to either East, or West.

The initial position of the robot is $(1, 1)$, and has orientation North, and the goal is to reach position (n, n) (in any orientation). Design an algorithm that computes a routing for the robot, with a minimum number of steps. Your algorithm should run in time polynomial in n.

Problem 2: Finding a cheap flight. Let $G = (V, E)$ be a directed graph, where V is a set of cities, and E represents all possible flights between the cities in V.

For every edge $\{u, v\} \in E$, you are given the duration of a direct flight from u to v, denoted by $d(u, v)$, which is an integer. For example, if you are at city u at time t, and you take a direct flight to v, departing at time $t' \geq t$, then you arrive at v at time $t' + d(u, v)$.

For every $\{u, v\} \in E$, you are given a timetable of all available direct flights from u to v, for some interval $\{0, \ldots, T\}$, where $T > 0$ is an integer. That is, for any $\{u, v\} \in E$, you are given a list of pairs of integers $((t_{u,v,1}, c_{u,v,1}), \ldots, (t_{u,v,k}, c_{u,v,k}))$, where the pair $(t_{u,v,i}, c_{u,v,i})$ denotes the fact that there is a direct flight from u to v that departs at time $t_{u,v,i}$, and costs $c_{u,v,i}$ dollars.

Design an algorithm that given a pair of cities $u, v \in V$, computes the cheapest possible route that starts at u at time 0, and ends at v at time at most T. The running time of your algorithm should be polynomial in $|V|$, and T.