**CS26N**
**Motion Planning for Robots, Digital Actors and Other Moving Objects**
**(Winter 2012)**

# Homework #1
# Posted: January 25 – Due: February 1

**How to complete this HW:** First copy this file; then type your answers in the file immediately below each question; finally print this file and return it to the Wednesday class on the day indicated above. If you need to draw anything, you can do it by hand.

**Your name:** ……………………………………………………………………………………

**Your email address:** ………………………………………………………………………………

**Note:** You may discuss this problem set with other students in the class (in fact, you are encouraged to do so).

## Grading:

| Problem# | Max. grade | Your grade |
|---|---|---|
| 1 | 30 | |
| 2 | 30 | |
| 3 | 20 | |
| 4 | 20 | |
| Total | 100 | |

## Problem 1:

1. Consider the environment shown in Figure 1. The brown polygonal regions depict obstacles. The robot is at the position Start and must reach position Finish. Draw the shortest path between Start and Finish on the figure.

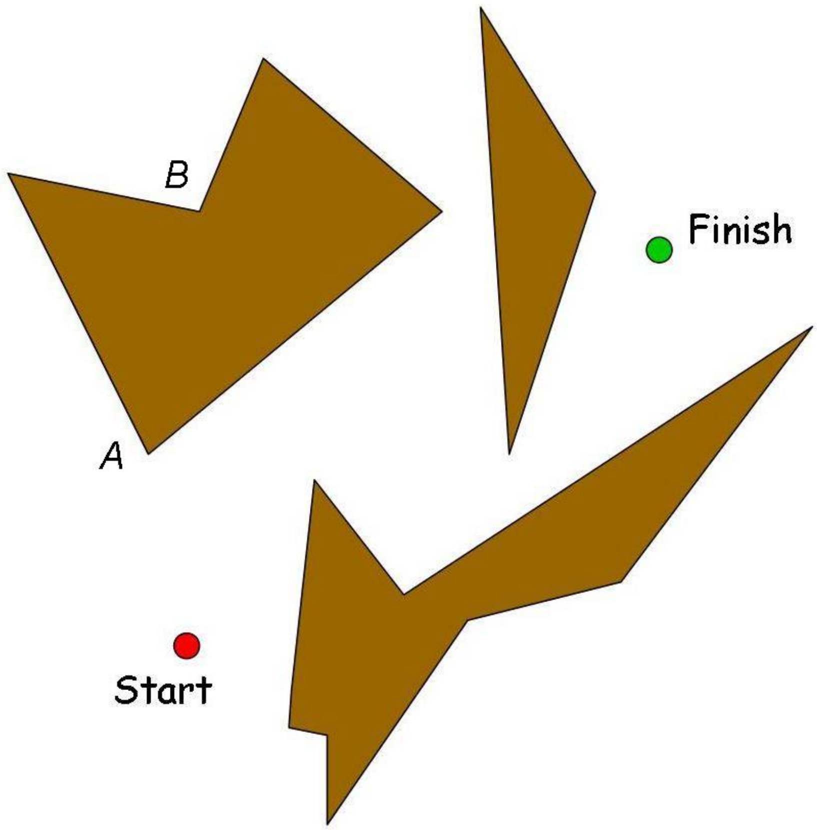


**Figure 1:** Environment with polygonal obstacles

2. Explain why in an environment where all obstacles are polygonal regions the shortest path between any two given points is a polygonal line whose vertices, if any, are obstacle vertices.

3. Let us call an obstacle vertex like vertex *A* in Figure 1 a *convex* vertex and an obstacle vertex like vertex *B* a *reflex* vertex. Explain why, in an environment where all obstacles are polygonal regions, the shortest path between any two given points (other than obstacle vertices) is a polygonal line whose vertices are convex obstacle vertices.

4. How could you use this result to speed up the Visibility Graph method?

## Problem 2:

Assume that a point robot operates in a two-dimensional environment like the one shown in Figure 2, where all obstacles are circular discs, such that no two obstacles overlap or touch one another.

1. Of Bug-1 and Bug-2, which algorithm always generates paths that are no longer than the paths generated by the other algorithm? Explain briefly why.

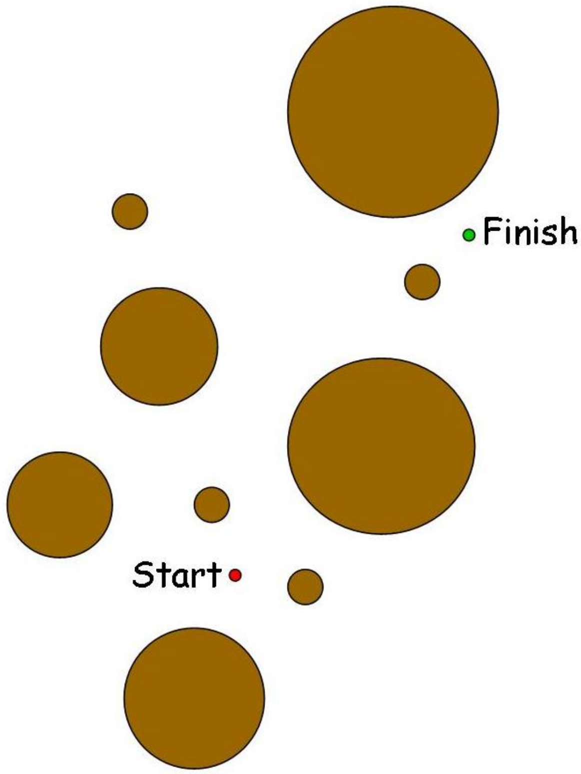


**Figure 2:** Example of an environment where all obstacles are circular discs

2. Imagine a Bug-3 algorithm that would follow the steepest descent of a potential field computed as the sum of an attractive potential and a repulsive potential (defined as in class). Like its siblings Bug-0, Bug-1, and Bug-2, Bug-3 does not use any grid. How the distance of influence of the obstacles (denoted by $d_{min}$ in the slides) should be set so that Bug-3 is guaranteed to always reach any given goal? Assuming that you answered correctly to the previous question, there is a small detail to take care of. Which one?

3. Consider the variant of the algorithm Bug-3 – let us call it Bug-3* – where the repulsive potential is always null. How does Bug-3* differ from Bug-2?

## Problem 3:

A robot moves from vertices to vertices in the unbounded regular 2D grid shown in Figure 5. The initial position of the robot is the vertex (0,0) marked with a black dot. At each step the robot can move from its current position by a unit increment up, down, left, or right.

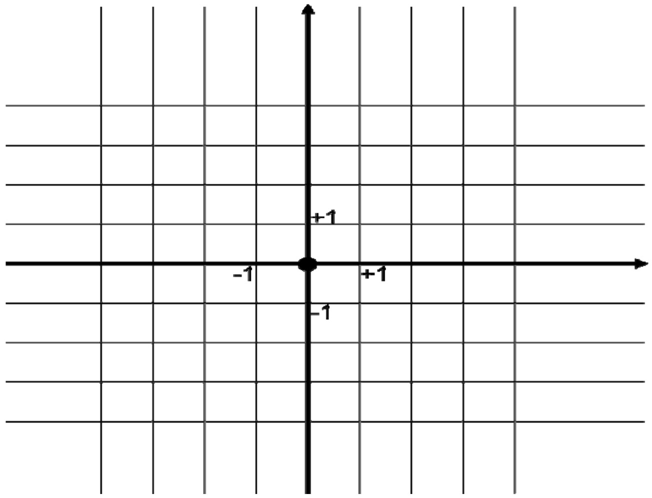


**Figure 5:** A portion of the grid used in Problem 4

How many distinct positions can the robot reach in $k > 0$ steps that it could not reach in less than $k$ steps?

(i) $4^k$ (ii) $4k$ (iii) $4k^2$

Explain your answer.

## Problem 4:

In a regular grid in 2D space, a vertex $(x,y)$ has 4 neighbors if only one coordinate is allowed to change: $(x-1,y)$, $(x+1,y)$, $(x,y-1)$, and $(x,y+1)$. It has 8 neighbors if one or two coordinates are allowed to change simultaneously: $(x-1,y)$, $(x+1,y)$, $(x,y-1)$, $(x,y+1)$, $(x-1,y-1)$, $(x-1,y+1)$, $(x+1,y-1)$, and $(x+1,y+1)$.

1. In a regular grid in 3D space, how many neighbors has a vertex $(x,y,z)$ if at most one coordinate is allowed to change? How many neighbors does it have if more than one coordinate (up to 3) are allowed to change?

2. In a regular grid in $n$D space, how many neighbors has a vertex $(x_1,x_2,...,x_n)$ if at most one coordinate is allowed to change? How many neighbors does it have if more than one coordinate (up to n) are allowed to change? Check your answer for n=2 and n=3.