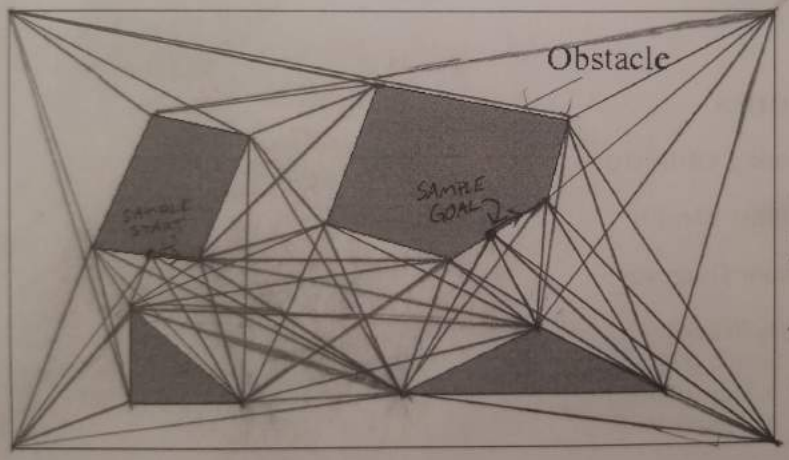
2.)

Here, we will create a visibility graph for a sample workspace with polygonal obstacles. Assume that the start and goal nodes are on the edge of the obstacles. The edges of the image are also obstacles so don't forget those.

Create the visibility graph for the following environment. You may hand-draw this, use an image editor or create a program to accomplish this task.



☆ IF STARTS AND GOALS EXIST ON EDGES OF OBSTACLES; THEN, THEY NEED ONLY FOLLOW THEIR EDGE TO ACCESS/DEPART THE VISIBILITY GRAPH ROADMAP (i.e. ignore the visibility lines emanating from the start & goal samples)

Figure 1: Sample Environment 1.

3.)

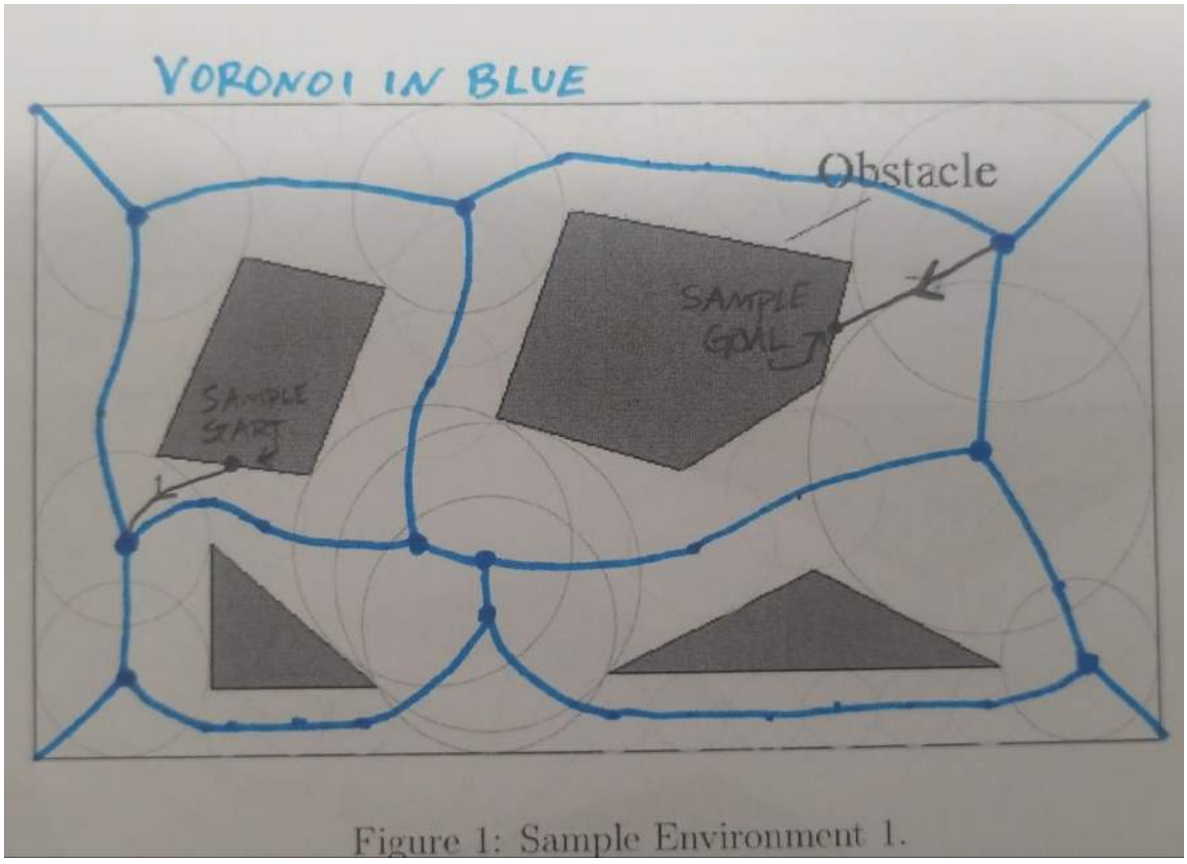
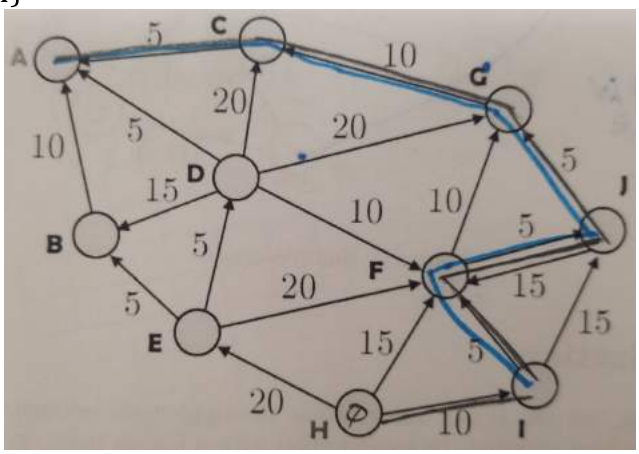


Figure 1: Sample Environment 1.

4.)

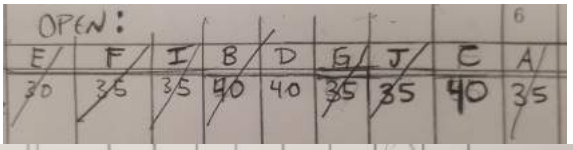
1. Search the Graph:
BFS: $\{H, E, F, I, D, G, J, A\}$
DFS: $\{E, B, A\}$

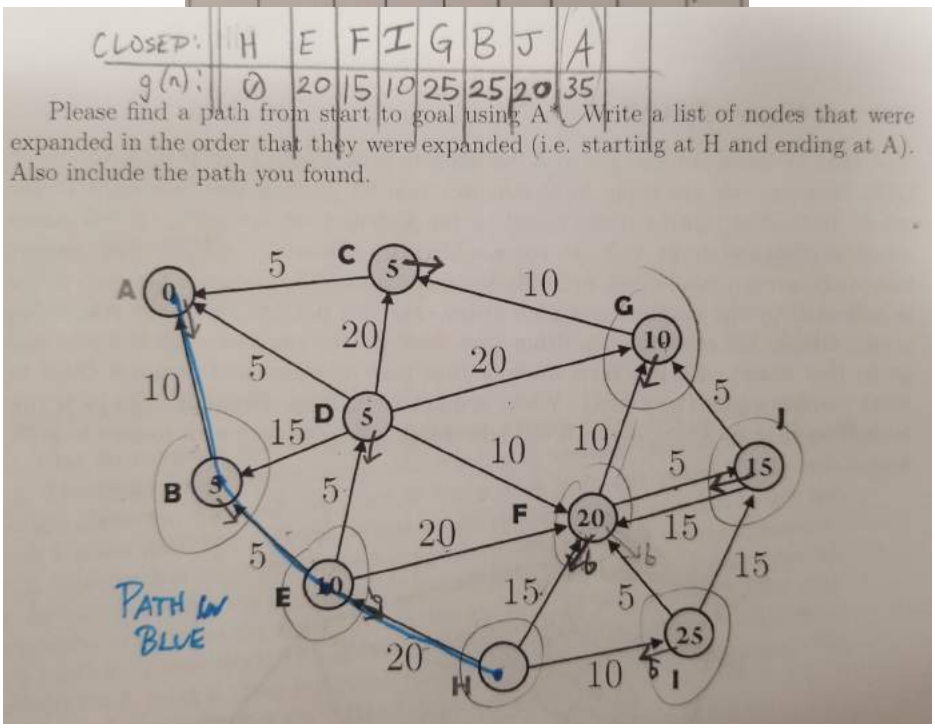
2. A Short Path?
$\{H, I, F, J, G, C, A\}$



3. A Shorter Path
Ordered List of Expanded Nodes (Closed List): $\{H, E, F, I, G, B, J, A\}$

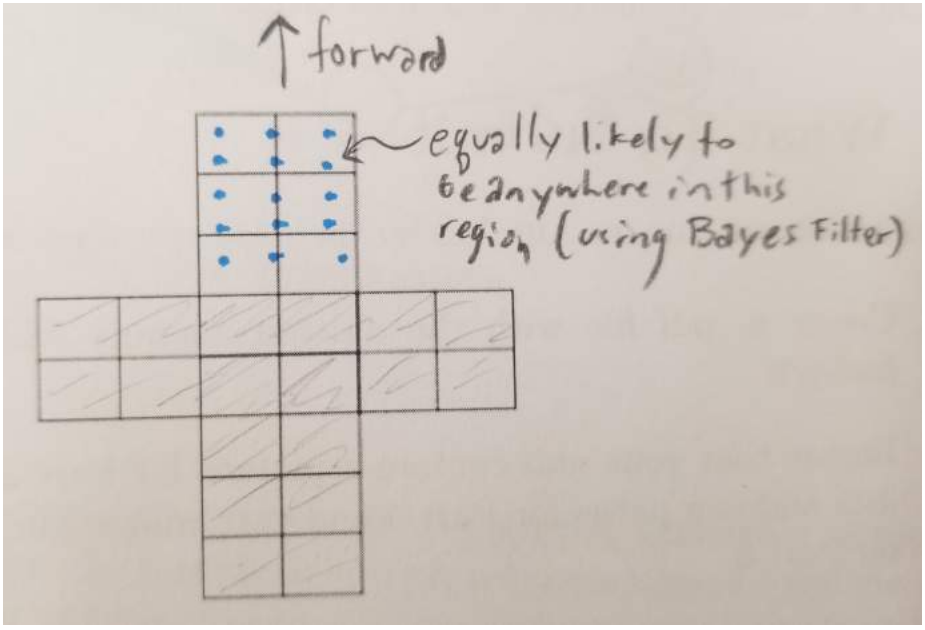
Path:



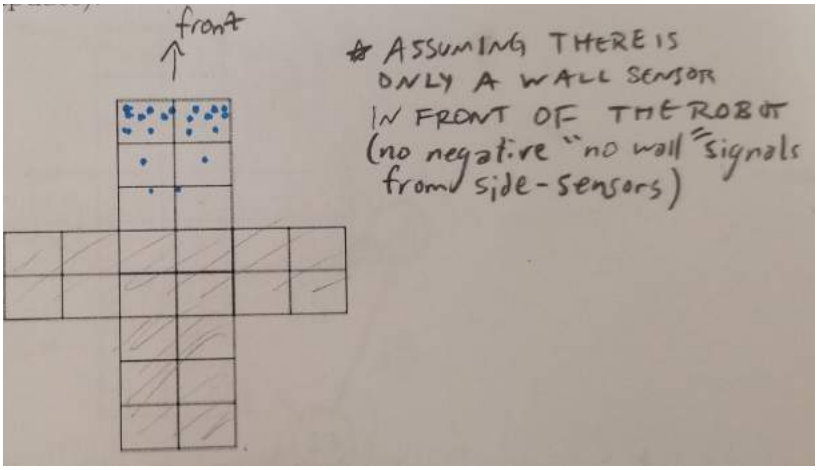


5.)

a.



b.



While a very good estimate of the robot's location now exists, *it's exact location is not definitively known* since there are many regions with non-zero (and non-effectively zero) probabilities and even a series of lateral positions near the detected wall with equally high-probabilities (assuming there aren't side sensors not detecting the presence of a wall).