RI 16x62: Lab 7: *The Beginning*

Due: Tuesday, Week 11

Introduction:

This lab will bring you joy, tears, laughs, depression, nausea, and good clean fun. Your task is rewarding, arduous, challenging, deceptively easy, stimulating, and will desensitize you to all your other classes, gradually, over the course of the next four weeks. First, we give you some important context for this lab, or game. Then, we actually spell out the assignment for those of weak imagination.

Preamble: The Early Years

Long, long, long ago, the people of your planet Zwytwerp started planning to mine the asteroids of all the nearby planetary systems. "Mining what?" you may wonder. Well, wonder no more. Your planet ran out of terrestrial gold long ago; today, your only hope for continuing to produce gaudy watches for the neighborhood galaxies is to import extraterrestrial gold.

Now, asteroids are not very amenable to mining, having a fairly complex surface topology. So, your fellow Zwytwerps charged the government's space agency, ZASA, with figuring out and implementing a solution to the Great Mining Problem, referred to as the GMP by most in-the-know Zwytwerpians.

Being a government agency, ZASA discussed the problem for many moons and had great trouble reaching consensus. They decided on a way of mining the gold, but couldn't decide on a way to bring the gold back to Zwytwerp.

ZASA started by discretizing the asteroid surface, marking adjacent nodes based on their connectivity. Incredibly, the Chief Discretizer at ZASA was actually the great-grandson of a graduate of an ancient robot programming class at a terrestrial university called CMU. Believe it or not, ZASA ended up using almost precisely the same maze format that you learned for labs 4 through 6! Isn't that something?

All the news is not good, however, The Chief Mining-Problem-Solver was not blood-related to that ancient, mysterious class. So, ZASA used an especially corny method for the actual mining process. Several thousand digger robots were fired at each asteroid, carefully controlled so that one digger landed in each discrete cell. These little monsters would locate the overgrown gold nuggets. The digger robots would dig for gold-- but because of budget cuts, they were altogether wimpy. Once they found gold, they would do silly things with the gold, such as balancing it precariously on the walls of their node. When these digger robots found gold, they would broadcast the location of the gold back to Zwytwerp on that encrypted communication line that the ruthless Bodorks (the menace of the galaxies) hadn't unscrambled yet.

Postamble: The Rather Recent Years

But that is ancient history now. The folks at ZASA have finally decided how to retrieve the gold. They decided to hire someone. After examining the conventional pay scales for computer professionals, they decided to hire a university student, who would work for free. That's you.

The plan is to land a robotic mother ship on the asteroid surface, with one or two robots on-board (the uncertainty is because of possible future budget problems). The bad news is that because of current budget cuts, your robots' arms will be incredibly weak. As a result, a robot can carry at most one piece of gold.

Due to incredibly bad foresight, the ZASA crew installed only a few small cargo bays (between three and ten, depending on the asteroid) on the mother ship. Each cargo bay can only hold only one piece of gold, and so your job is to collect the gold and deliver it to the mother ship by dropping off the pieces of gold, one in each cargo bay.

You are charged with the economic security of Zwytwerp and the mental security of the entire neighborhood of galaxies – Zwytwerp is not only the largest producer of gaudy watches, but the producer of the gaudiest watches on the market. Your job is to retrieve the gold, fast!

Assignment 7.0:

Write your first game robot controller. You will read in a GameWorld file (described below). You **must** abide precisely to all the standards we describe below. For this assignment, you will write a controller for a 1-robot version of the Game. This means that you will have no teammate. Using vision, your robot will detect the gold to verify that it is still there. Read more about this below. You will then make-believe drop the gold off at your cargo bay. Again read below on this.

- A node in the maze will have access to at most one piece of gold
- Your robot may not drop gold at any location other than a cargo bay
- Each cargo bay can hold a maximum of one piece of gold
- You will only know how many cargo bays there are at run-time

Note: not all maze cells are necessarily reachable from your robot's starting position.

We have a specific protocol that your robot must use when interacting with the gold. Your robot will signal its intentions to pick up gold (both verbally and on the screen), then look for the gold with vision. If your robot sees the gold, it should announce that it is picking it up although, this week, you shouldn't really try (unless you really want to).

To drop gold off, you must stop at a destination and verbalize your desire as above then wait **5 seconds**. If the node is a destination and doesn't have gold yet, we will remove your gold and put it in the node. If the node is full already or if you're not at a destination cargo bay, you're in deep trouble. We'll remove the gold and it will disappear from the world.

We will be testing your code by running a game with just your one robot in a simple gameworld. In this test, we aren't looking for any smarts yet! We just want to ensure that you've put everything together: you're getting to the gold, you're detecting the gold; delivering it properly; you're using the protocol for depositing gold.

Just in case you missed the point of that last paragraph, **Do not try out a really smart strategy yet. Just put all the pieces together this week.**

Second warning: This week you will lose more than the simple 1 point for silly bugs. Look at the appropriate directory, games, for gameworlds. Make sure your controller works on the simple words in this directory, as we will be using one. A simple one. On Tuesday. Yes indeed.

Hints

- There is a good chance that your robot(s) will occasionally get lost. Since it cannot stop and ask for directions, your robot should have a strategy for becoming un-lost. You already know how to do this, right?

- It is a good idea to make sure the robot is not lost when it is about to do something important, such as drop gold at the destination, because dropping gold in the wrong place results in lost gold, which is a bad thing.

Appendix A: File Format

The file format is similar to the MazeWorld you have been using for your maze specifications until now. The class is called GameWorld. When the game begins, we specify the GameWorld, such as c:\16x62\games\m2, and you must load in that file.

The new, additional information in this new GameWorld class includes:

dimensions of the mazeworld and complete spec of its interior walls
number of robots on your team and your particular robot's position
position of your gold delivery ports (and, by virtue of measuring the vector's size, the number of ports)
position of every piece of gold. Although the GameWorld data structure allows a gold to be contained wholly within a cell, there will be **no such gold**. All gold will be on walls between two cells. Sometimes you will be able to reach both cells, sometimes only one cell (the other being in closed enemy territory).

Appendix B: Displaying and Editing Games

You will find GameDisplay and GameEditor to be extremely useful. They are self-explanatory and are located in Samples. Remember, don't double-click basically ever when using GameEditor.

Appendix C: Sample Games

There are many example games for you to look at. This week you should just look in 16x62/games. NewGames is in a state of flux and will be changing shortly, so don't spend any time there. Specifically, make sure you can deal with g1, g2 and g3. In all probability, one of these three will be the game we test you with on Tuesday. The g4sm series provides a two-player cooperating game, and g4smsingle provides a single player version. This will fit in REL and should be fun for you. Same goes for the g5 set.

Then, g6 and above, including the named games (such as Pick One) are all games in which you have a real opponent, as you can see from the symmetry of the world. In these worlds, you and your opponents inhabit separate, permanently unreachable sides of a maze. So g6single, for example, is when you have one robot playing against an opponent who will be in the mirror image, same position on the other side of the world.

By and large, for this assignment, you should be satisfied with working on g1, g2, g3 and maybe g4smsingle and the s5 single version.