Assignment: Scavenger Search

Description

This assignment is the next assignment in the Scavenger series. In the series, you will create an agent that
can navigate from place to place in its world.

In this assignment, you will use the `Problem`, `State`, and `Action` classes with the tree search algorithm to
find a safe path from your agent’s location to the goal location. You will not be required to move the agent to
the goal, only find a path and display it to `std::cout`. You will repeat the search with breadth first and depth
first frontiers.

Requirements

Do a Breadth First Tree Search

Modify your agent code to do a breadth first tree search from its current location to the goal location. If a path
is found, display the action sequence required to reach the goal, and the charge your agent should have when
it reaches the goal location. Also display the number of nodes generated, and maximum number of nodes
stored.

Do a Depth First Tree Search

Modify your agent code to do a depth first tree search from its current location to the goal location. If a path is
found, display the action sequence required to reach the goal, and the charge your agent should have when it
reaches the goal location. Also display the number of nodes generated, and maximum number of nodes stored.

Notes

- Sample search code can be found in `ai-agents/prog/Rectangle/RectDriver.cpp`
- Use one of the `search-small-world` files for your runs.

Passoff

- Save the two sets of output to files. Submit the files to the CIT Submission System linked at the top of the
page.