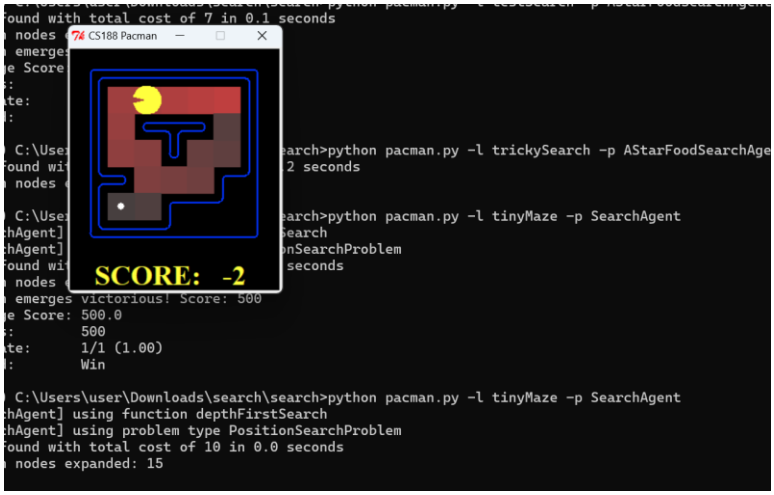
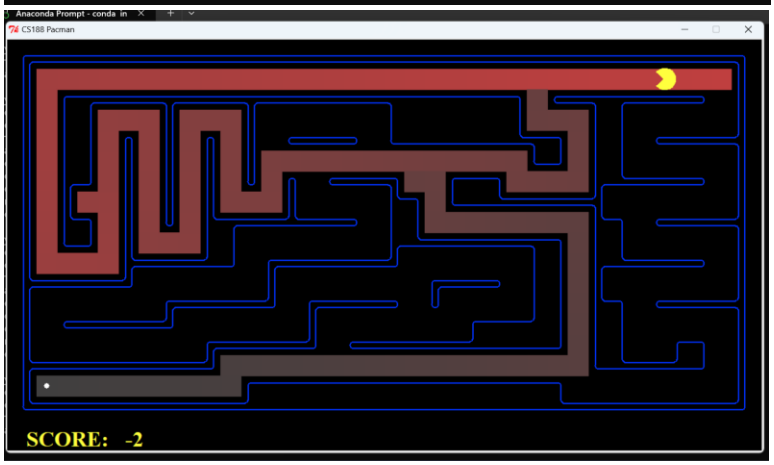
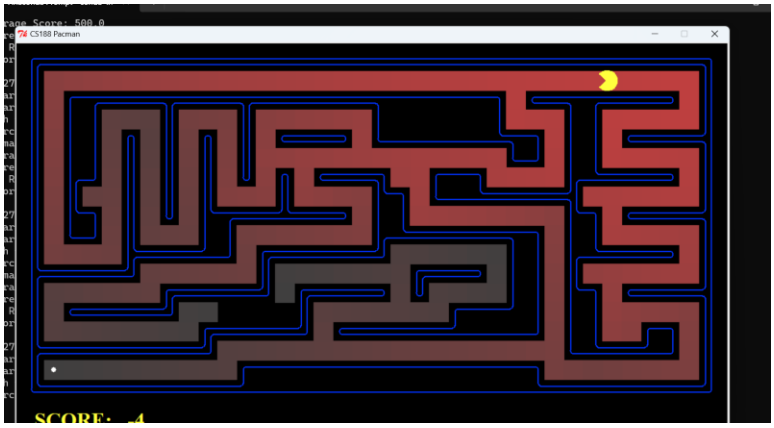
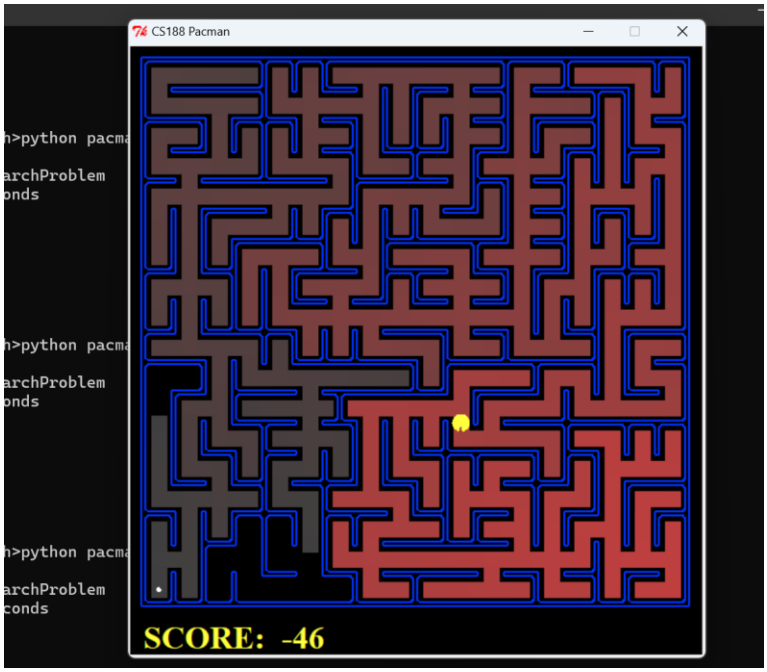
1. Depth-First Search (DFS)

After implementing the depth-first search, and running the command in the terminal, this is the solution it found:

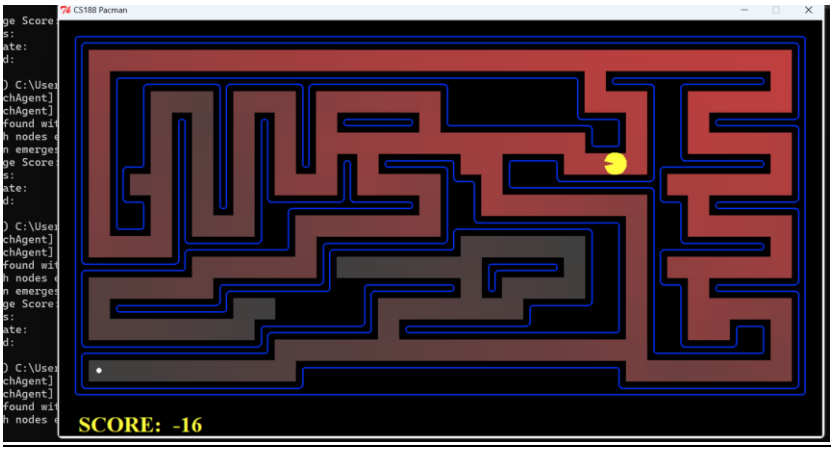
- The exploration order is not what I had expected.
- Pacman doesn't go through all the squares.

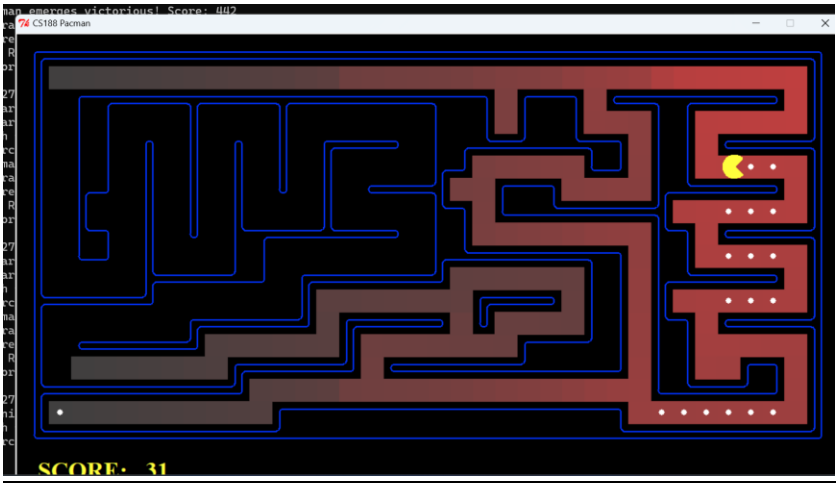


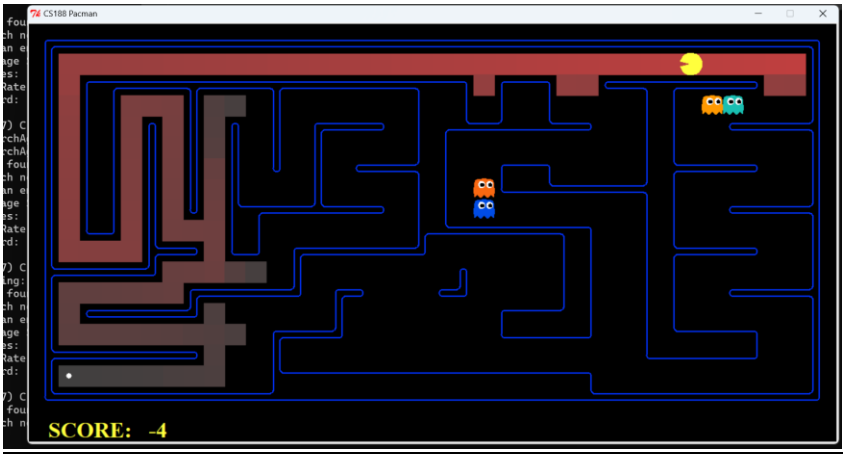


2. **Breadth-First Search(BFS)**
    - It indeed does get a least cost solution.
    - 
    - 

3. **Implement the uniform-cost graph search algorithm**

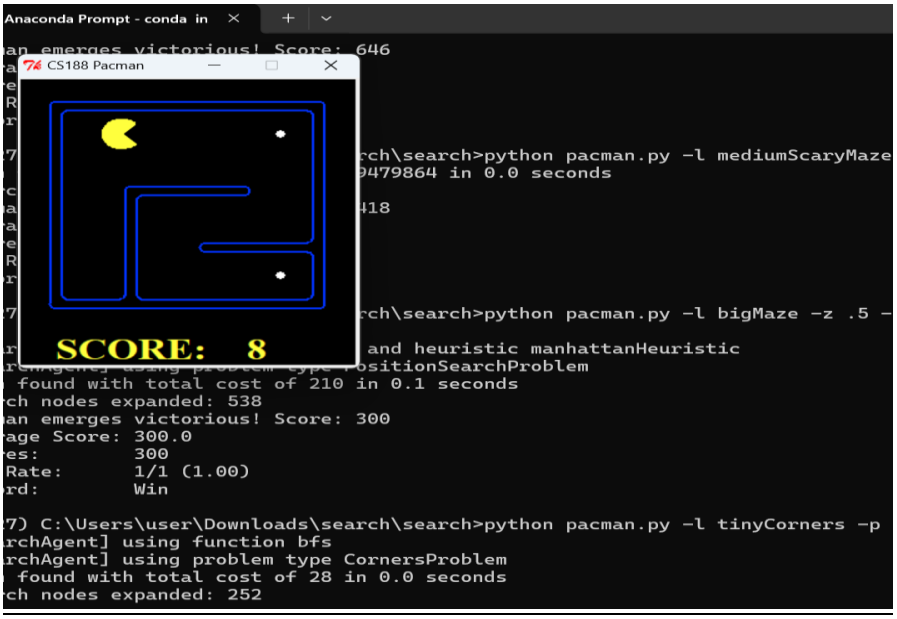


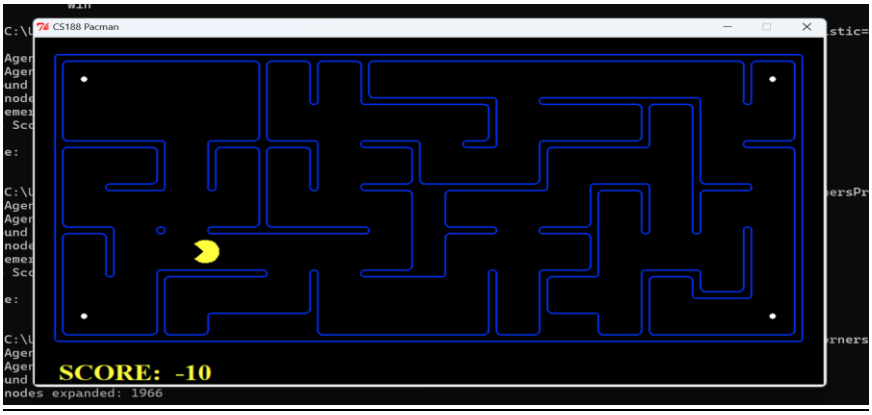




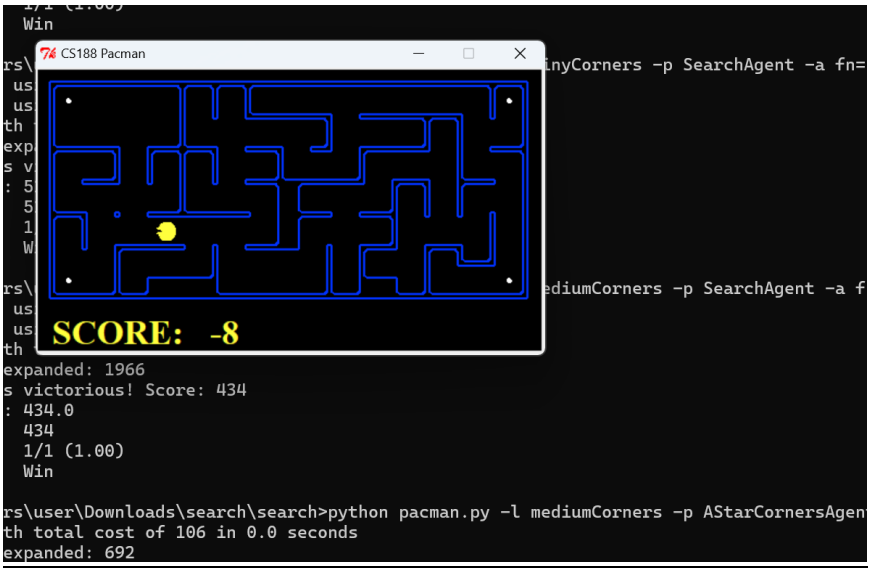
4. A* Search

5. Finding All the Corners

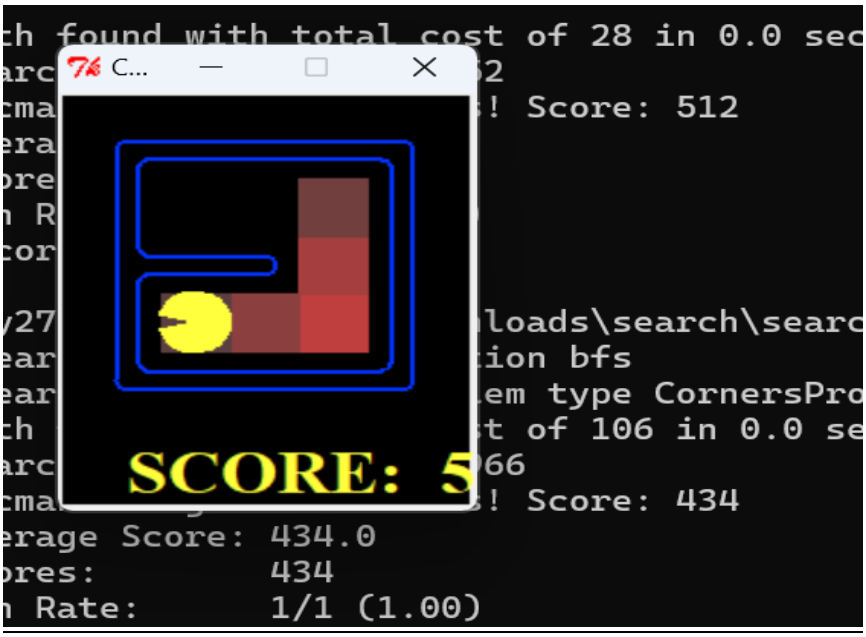




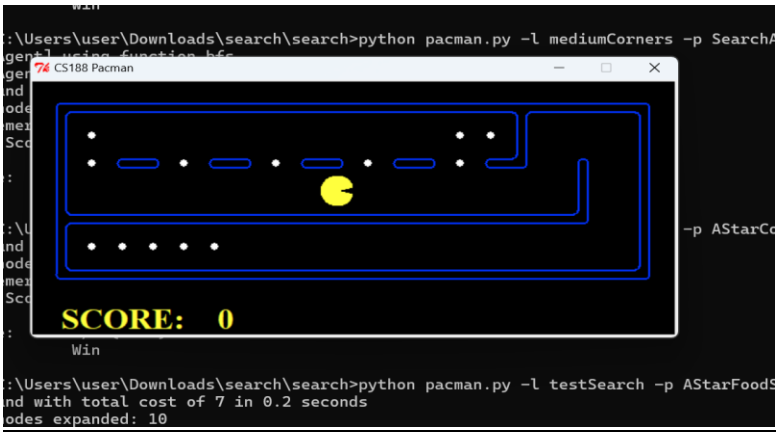
6. **Implementing a non-trivial**



7. **Eating all the dots**



8. **Filling in foodHeuristic**


CS188 Pacman
SCORE: 0
Win
:\Users\user\Downloads\search\search>python pacman.py -l testSearch -p AStarFoodS
nd with total cost of 7 in 0.2 seconds
odes expanded: 10