Introduction to Artificial Intelligence

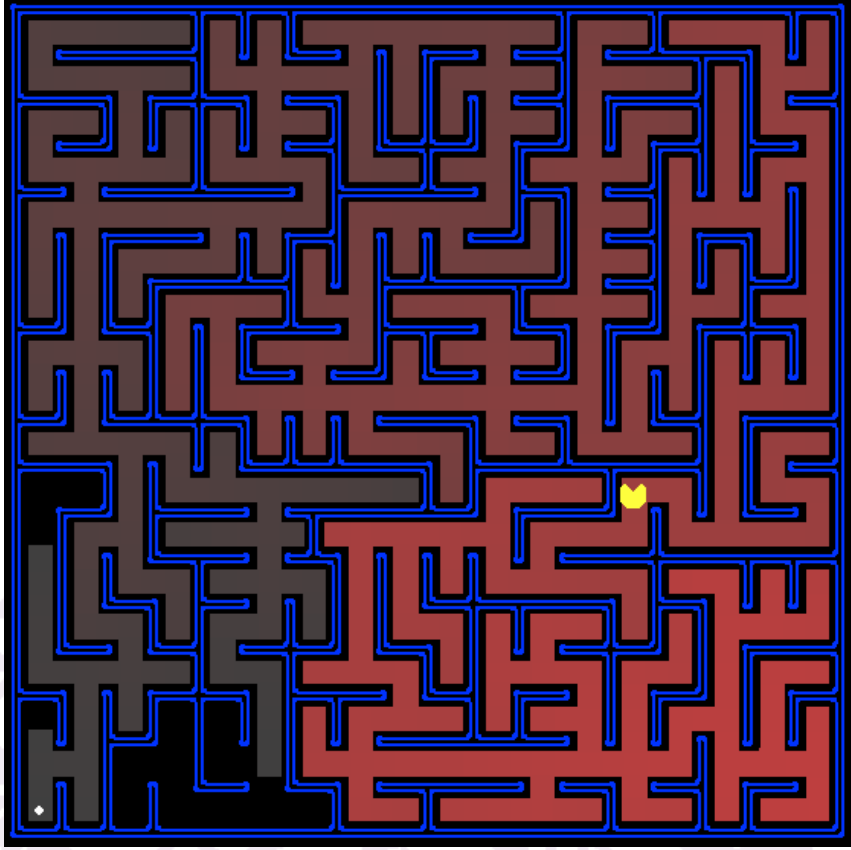
Project 1 - Search

Jianmin Li

Department of Computer Science and Technology
Tsinghua University

Spring, 2023

Search



- Berkeley Pac-Man Project 1
  - https://inst.eecs.berkeley.edu/~cs188/sp20/project1/
  - https://inst.eecs.berkeley.edu/~cs188/sp20/assets/files/search.zip

Search

- Demo
  - keyboardAgent
    - python pacman.py -l tinyMaze
  - trivial reflex agent
    - python pacman.py -l testMaze -p GoWestAgent
    - python pacman.py -l tinyMaze -p GoWestAgent

Basic Tasks

- Finding a Fixed Food Dot
  - Implement DFS algorithm in the depthFirstSearch function in search.py (2 points)
  - Implement BFS algorithm in the breadthFirstSearch function in search.py (2 points)
  - Implement the uniform-cost graph search algorithm in the uniformCostSearch function in search.py (2 points)
  - Implement *A** graph search in the empty function aStarSearch in search.py (2 points)

Basic Tasks

- Finding All the Corners
  - Implement the CornersProblem search problem in searchAgents.py (2 points)
  - Implement a non-trivial, consistent heuristic for the CornersProblem in cornersHeuristic (2 points)
- Eating All The Dots
  - Fill in foodHeuristic in searchAgents.py with a consistent heuristic for the FoodSearchProblem (3 points)

Bonus

- Suboptimal Search
  - Implement the function findPathToClosestDot in searchAgents.py (1 points)

Submission

- A 2-3 pages report (either Chinese or English)
  - Compare how these algorithms perform in Pac-Man environment, e.g. state numbers, time, etc
  - Discussion
- Zip the files as the following structure
  - student_id.zip (e.g. 20090112xx.zip)
    - student_id.pdf
    - search.py
    - searchAgents.py

Grading

- Due
  - 2023/4/5 23:59:59
- Correctness of algorithms (80%)
  - Different layouts
  - Check whether your search algorithm returns the right action sequence
- Report (20%)
- Policy
  - Discussion is encouraged , but must be written up individually
  - Do not copy/lend solution from/to others

谢谢！