Introduction to Artificial Intelligence

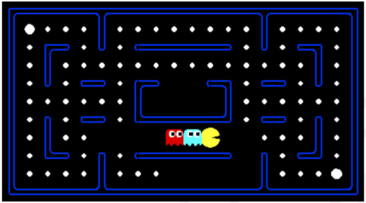
Project 2 – Multi-Agent PacMan

Jianmin Li

Department of Computer Science and Technology
Tsinghua University

Spring, 2017

Multi-Agent PacMan



- Berkeley Multi-Agent Pacman Project
  http://ai.berkeley.edu/multiagent.html
  https://s3-us-west-2.amazonaws.com/cs188websitecontent/projects/release/multiagent/v1/002/multiagent.zip

Berkeley Multi-Agent Pacman Project

- Demo
  - python pacman.py -p ReflexAgent
  - python pacman.py -p ReflexAgent -l testClassic

Basic Tasks (1)

- ReflexAgent
  - Improve the ReflexAgent in multiAgents.py
  - python pacman.py -p ReflexAgent -l openClassic -n 10 -q
  - 4 points
- MinimaxAgent
  - Implement minimax algorithm
  - In the provided MinimaxAgent class stub in multiAgents.py
  - Any number of ghosts
  - python pacman.py -p MinimaxAgent -l minimaxClassic -a depth=4
  - 5 points

Basic Tasks (2)

- AlphaBetaAgent
  - Implement alpha-beta pruning algorithm
  - In the provided AlphaBetaAgent class stub in multiAgents.py
  - python pacman.py -p AlphaBetaAgent -a depth=3 -l smallClassic
  - 5 points
- ExpectimaxAgent
  - Random ghosts
  - python pacman.py -p AlphaBetaAgent -l trappedClassic -a depth=3 -q -n 10
  - python pacman.py -p ExpectimaxAgent -l trappedClassic -a depth=3 -q -n 10
  - 6 points

Bonus

- Better Evaluation Function
  - Write a better evaluation function for pacman
  - In the provided function betterEvaluationFunction
  -
    ```
    python pacman.py -l smallClassic -p ExpectimaxAgent -a evalFn=better -q -n 10
    ```
  - 5 points

Submission

- A 2-3 pages report (either Chinese or English)
  - Compare how these agents perform, e.g. state numbers, time, win rate, etc
  - Discussion
- Zip the files as the following structure
  - student_id.zip (e.g. 20090112xx.zip)
    - ★ student_id.pdf
    - ★ multiAgents.py

Grading

- Due
  - 2017/4/18 23:59:59
- Correctness & performance of agents (80%)
  - Run multiple games for each agent
  - Grading rules
- Report (20%)