

Pacman Project Milestone I

Rule-based & Search



【人工智慧概論】

授課教師／孫春在

助教／呂冠霆 吳晨晨 柯維然 鍾幸芸 黃慧瑜 黃奕齊 劉長謬 林怡秀

日期／2017.03.30

-
1. Environment
 2. Rule-based System
 3. Search Algorithms

Environment

- 1.ANACONDA
- 2.PYTORCH

環境配置

Anaconda(python2.7) for Pacman

- OpenSource
- Numpy , scipy , matplotlib and scikit-learn
- Jupyter notebook
- Spyder(IDE)

PyTorch for Deep Learning

- OS : OSX , Linux
- GPU not required

Anaconda



ANACONDA®

Log In **Get Support** **Search** **Contact**

PRODUCTS **COMMUNITY** **CONSULTING** **TRAINING** **ABOUT** **RESOURCES**

DOWNLOAD ANACONDA NOW

Download for

GET SUPERPOWERS WITH ANACONDA

Anaconda is the leading open data science platform powered by Python. The open source version of Anaconda is a high performance distribution of Python and R and includes over 100 of the most popular Python, R and Scala packages for data science.

Additionally, you'll have access to over 720 packages that can easily be installed with conda, our renowned package, dependency and environment manager, that is included in Anaconda. See the [packages](#) included with Anaconda and the Anaconda [changelog](#)

Which version should I download and install?
With Anaconda you can run multiple versions of Python in isolated environments, so choose the download with the Python version that you use more often, as that will be your default Python version.

If you don't have time or disk space for the entire distribution, try [Miniconda](#) which contains only conda and Python. Then install just the individual packages you want through the conda command.

Anaconda



ANACONDA®

Download for Windows Download for macOS Download for Linux

Anaconda 4.3.1

For Windows

Anaconda is BSD licensed which gives you permission to use Anaconda commercially and for redistribution.

[Changelog](#)

1. Download the installer
2. Optional: Verify data integrity with [MD5 or SHA-256 More info](#)

Double-click the .exe file to install Anaconda and follow the instructions on the screen

Behind a firewall? Use these [zipped Windows installers](#)

Python 3.6 version
[64-BIT INSTALLER \(422M\)](#)

32-BIT INSTALLER (348M)

Python 2.7 version
[64-BIT INSTALLER \(414M\)](#)

32-BIT INSTALLER (339M)

[GET ANACONDA SUPPORT](#)

Download for Windows Download for macOS Download for Linux

Anaconda 4.3.1

For Linux

Anaconda is BSD licensed which gives you permission to use Anaconda commercially and for redistribution.

[Changelog](#)

1. Download the installer
2. Optional: Verify data integrity with [MD5 or SHA-256 More info](#)

In your terminal window type one of the below and follow the instructions:
Python 3.6 version

```
bash Anaconda3-4.3.1-Linux-x86_64.sh
```

Python 2.7 version

```
bash Anaconda2-4.3.1-Linux-x86_64.sh
```

NOTE: Include the "bash" command even if you are not using the bash shell.

Python 3.6 version
[64-BIT INSTALLER \(474M\)](#)

32-BIT INSTALLER (399M)

Python 2.7 version
[64-BIT INSTALLER \(462M\)](#)

32-BIT INSTALLER (388M)

[GET ANACONDA SUPPORT](#)

Download for Windows Download for macOS Download for Linux

Anaconda 4.3.1

For macOS

macOS 10.12.2 users: To prevent permissions problems, we recommend that you upgrade to macOS 10.12.3 or later before installing Anaconda.

Anaconda is BSD licensed which gives you permission to use Anaconda commercially and for redistribution.

[Changelog](#)

Graphical Installer

1. Download the graphical installer
2. Double-click the downloaded .pkg file and follow the instructions

Command Line Installer

1. Download the command-line installer
2. Optional: Verify data integrity with [MD5 or SHA-256 More info](#)
3. In your terminal window type one of the below and follow the instructions:
Python 3.6 version

```
bash Anaconda3-4.3.1-MacOSX-x86_64.sh
```

Python 3.6 version
[GRAPHICAL INSTALLER \(424M\)](#)

[COMMAND-LINE INSTALLER \(363M\)](#)

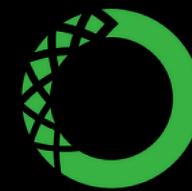
64-Bit

Python 2.7 version
[GRAPHICAL INSTALLER \(419M\)](#)

[COMMAND-LINE INSTALLER \(358M\)](#)

[GET ANACONDA SUPPORT](#)

Spyder



ANACONDA®

The screenshot shows the Spyder IDE interface. On the left, the Spyder Editor displays a script named 'temp.py' with the following content:

```
1 # -*- coding: utf-8 -*-
2 """
3 Spyder Editor
4
5 This is a temporary script file.
6 """
7
8
```

On the right, the Variable explorer shows a table with one row:

| Name | Type | Size | Value |
|------|------|------|-------|
| | | | |

Below the Variable explorer is the IPython console, which shows the following output:

```
Variable explorer File explorer Help
IPython console
Console I/A
Python 2.7.13 |Anaconda 4.3.0 (64-bit)| (default, Dec 19 2016, 13:29:36) [MSC v.1500 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 5.1.0 -- An enhanced Interactive Python.
?           -> Introduction and overview of IPython's features.
%quickref  -> Quick reference.
help       -> Python's own help system.
object?    -> Details about 'object', use 'object??' for extra details.

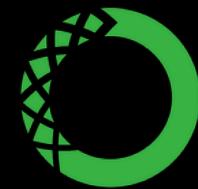
Restarting kernel...

Kernel died, restarting

In [1]:
In [1]:
```

The bottom status bar indicates the following information:

Python console History log IPython console
Permissions: RW End-of-lines: CRLF Encoding: UTF-8 Line: 8 Column: 1 Memory: 85%



ANACONDA®

Jupyter notebook

jupyter spectrogram Last Checkpoint: an hour ago (autosaved)

File Edit View Insert Cell Kernel Help

Cell Toolbar: None

```
In [1]: from scipy.io import wavfile  
rate, x = wavfile.read('test_mono.wav')  
  
In [2]: import matplotlib.pyplot as plt  
fig, (ax1, ax2) = plt.subplots(1, 2, figsize=(12, 4))  
ax1.plot(x); ax1.set_title('Raw audio signal')  
ax2.specgram(x); ax2.set_title('Spectrogram')  
plt.show()
```

Raw audio signal

Spectrogram

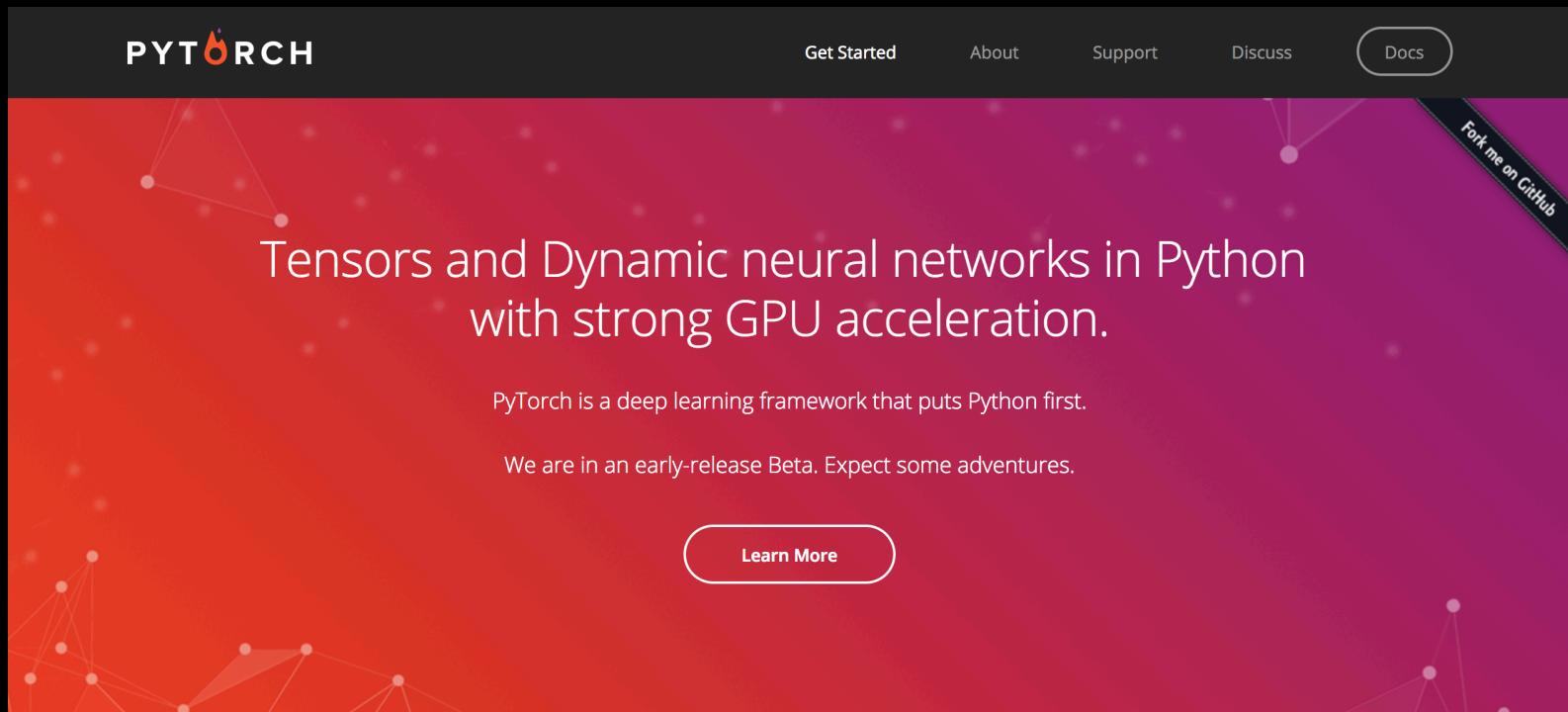
Python Tutorial from CS231n

Python & Numpy

Ipython notebook



PyTorch



The image shows a screenshot of the PyTorch website's landing page. The header features the word "PYTORCH" in white capital letters next to the orange flame logo. Below the header is a dark red banner containing white text. The banner reads: "Tensors and Dynamic neural networks in Python with strong GPU acceleration." To the right of the banner is a "Docs" button. Below the banner, there are two paragraphs of text: "PyTorch is a deep learning framework that puts Python first." and "We are in an early-release Beta. Expect some adventures." A "Learn More" button is located in the center of the page. The background of the page has a subtle network graph pattern.

PYTORCH

Get Started About Support Discuss Docs

Fork me on GitHub

Tensors and Dynamic neural networks in Python with strong GPU acceleration.

PyTorch is a deep learning framework that puts Python first.

We are in an early-release Beta. Expect some adventures.

Learn More



PyTorch

- Torch
- 符合直覺、好理解、易用
- GPU 加速
- 動態神經網絡：基於 **tape** 的 **Autograd**
- 和 **Numpy** 交互使用



PyTorch

Companies & Universities developing PyTorch





PyTorch

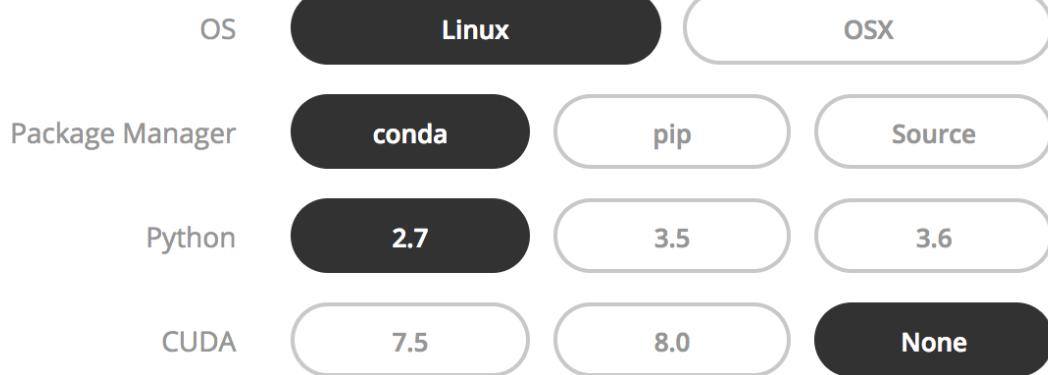
Only on OSX and Linux
Version : CPU or GPU

Get Started.

Select your preferences, then run the PyTorch install command.

Please ensure that you are on the latest pip and numpy packages.

Anaconda is our recommended package manager



Run this command:

```
conda install pytorch torchvision -c soumith
```



PyTorch

[GitHub/PyTorch](#)

[Tutorial](#)

[PyTorch documents](#)

[Torch](#)

Others

[TensorFlow](#)

[Theano](#)

[Keras \(中文\)](#)

[TFLearn](#)

[TensorLayer \(中文\)](#)

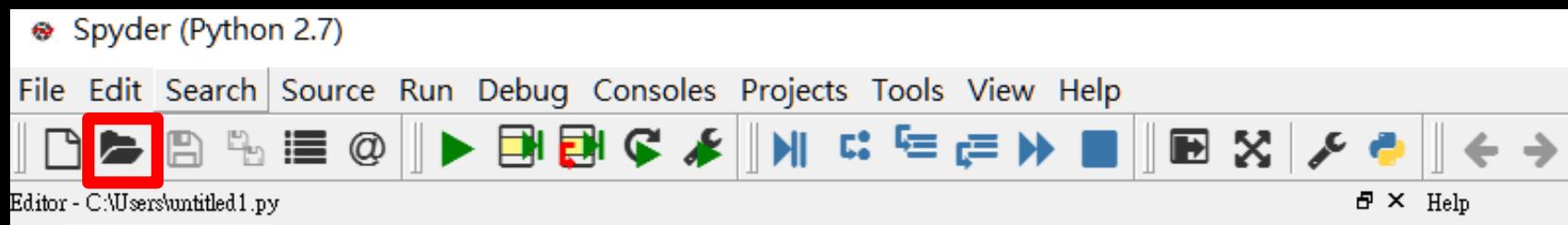
[Docker](#)



Spyder(1/4)

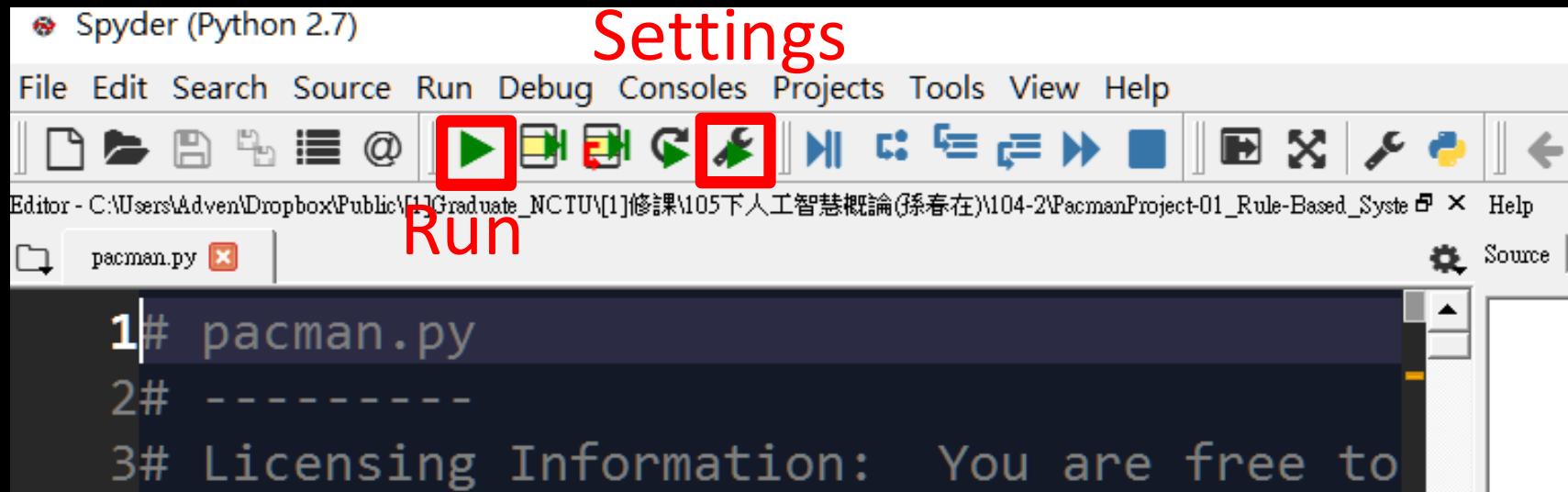
Step 1: Open Spyder

Step 2: Open the file “Pacman/pacman.py”



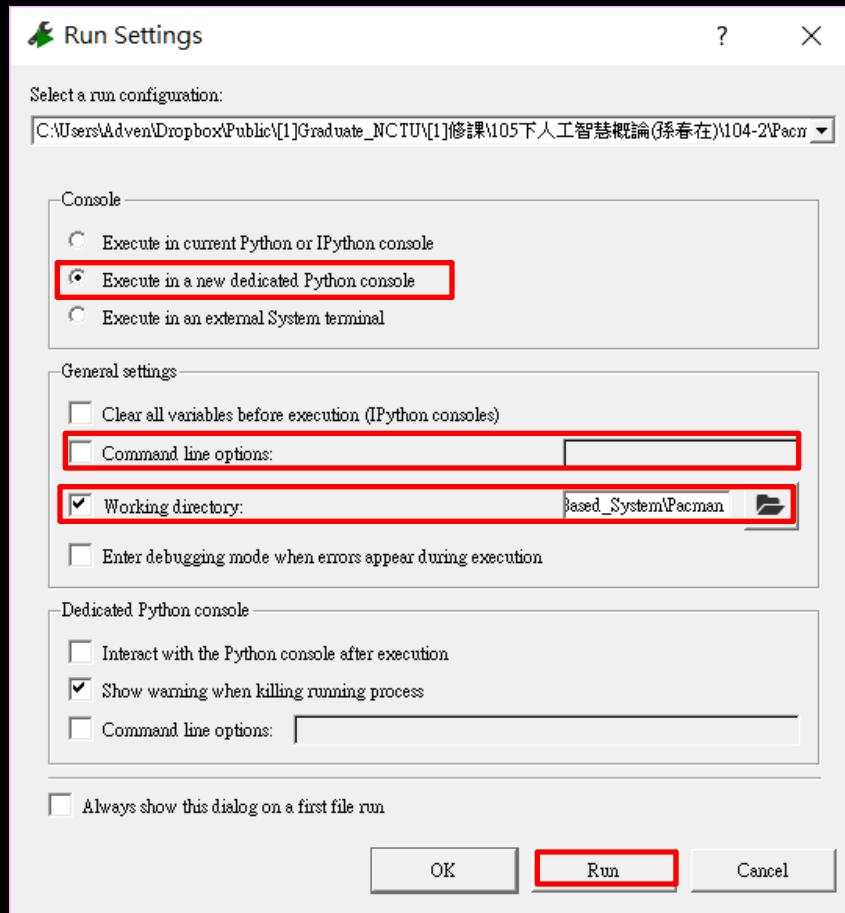
Spyder(2/4)

Step 3: Click “Settings”

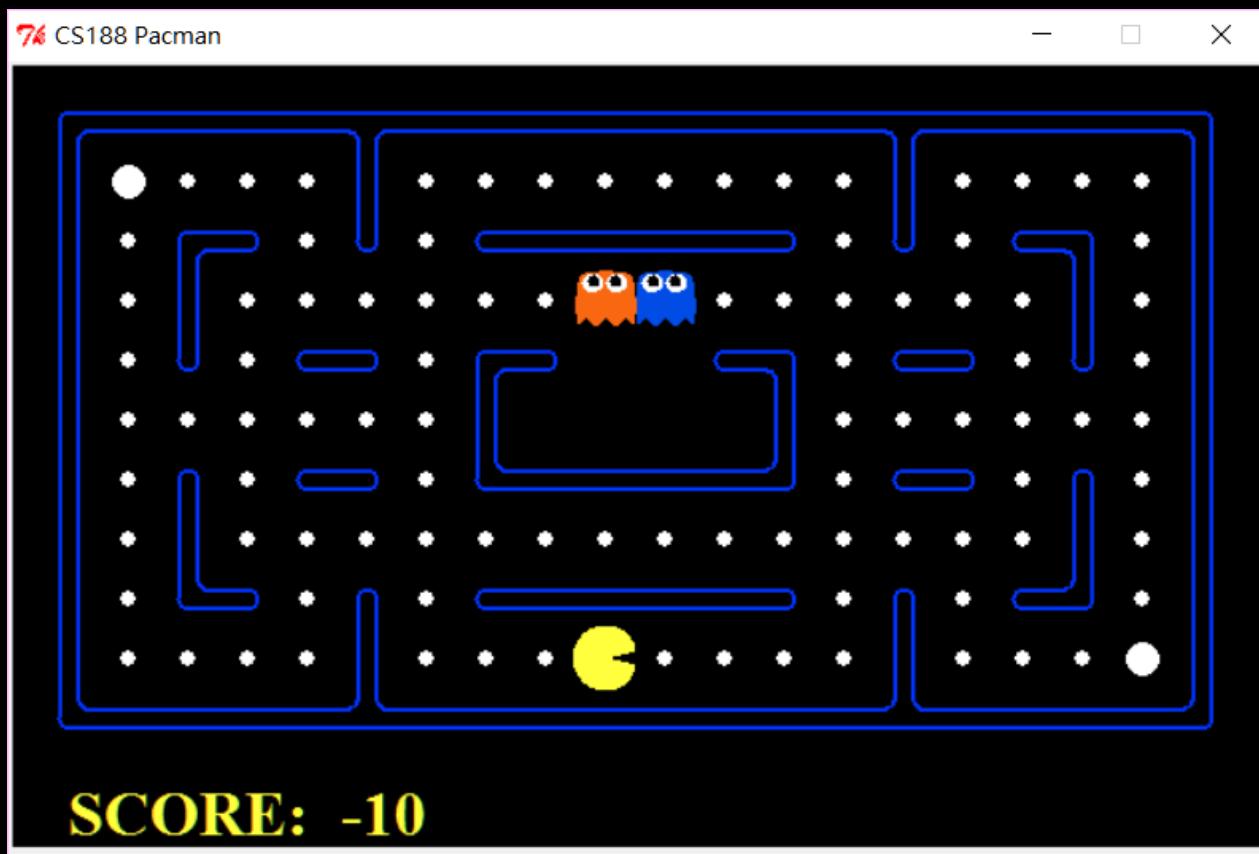


Spyder(3/4)

Step 4



Spyder(4/4)



SCORE: -10

Rule-based

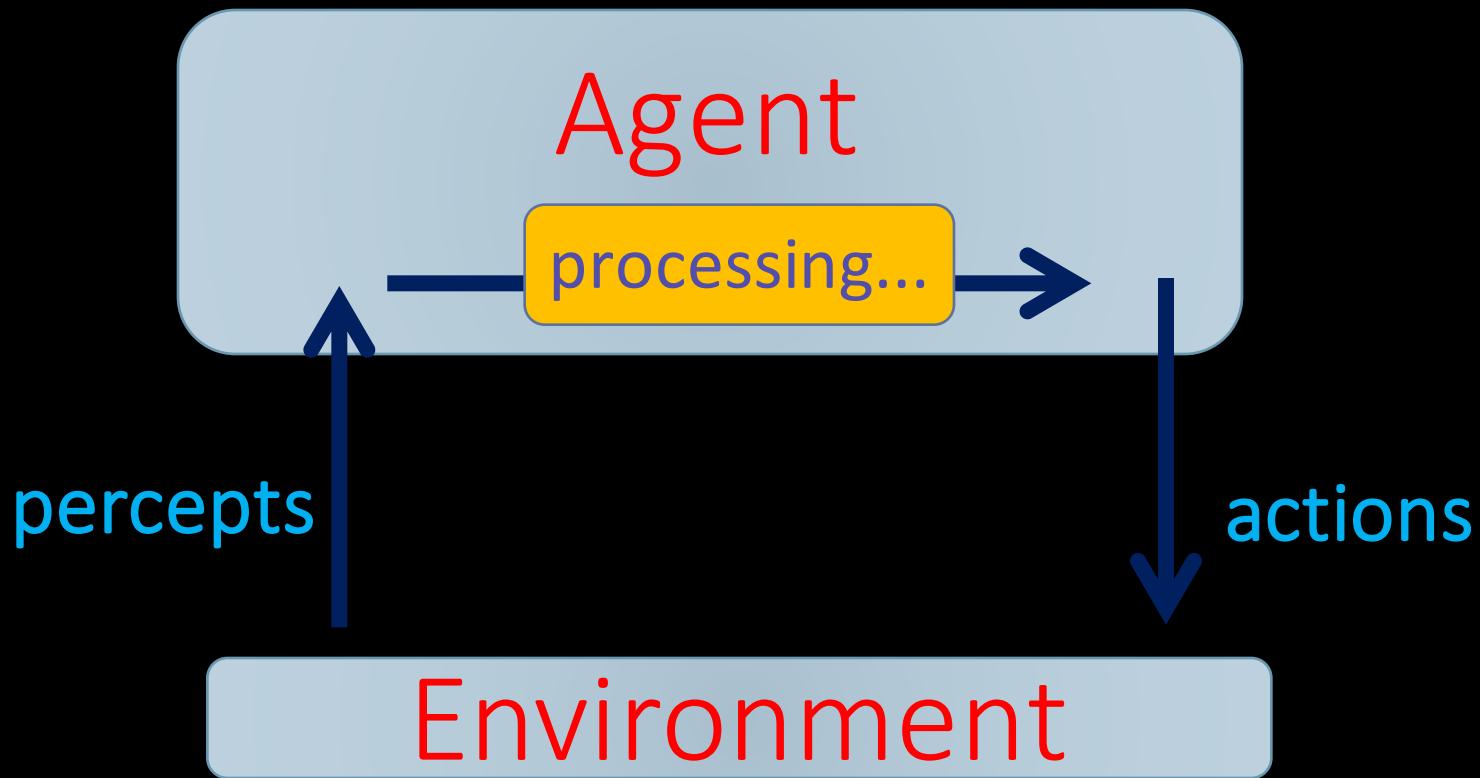
P1-1 CleanerAgent

P1-2 FroggerAgent

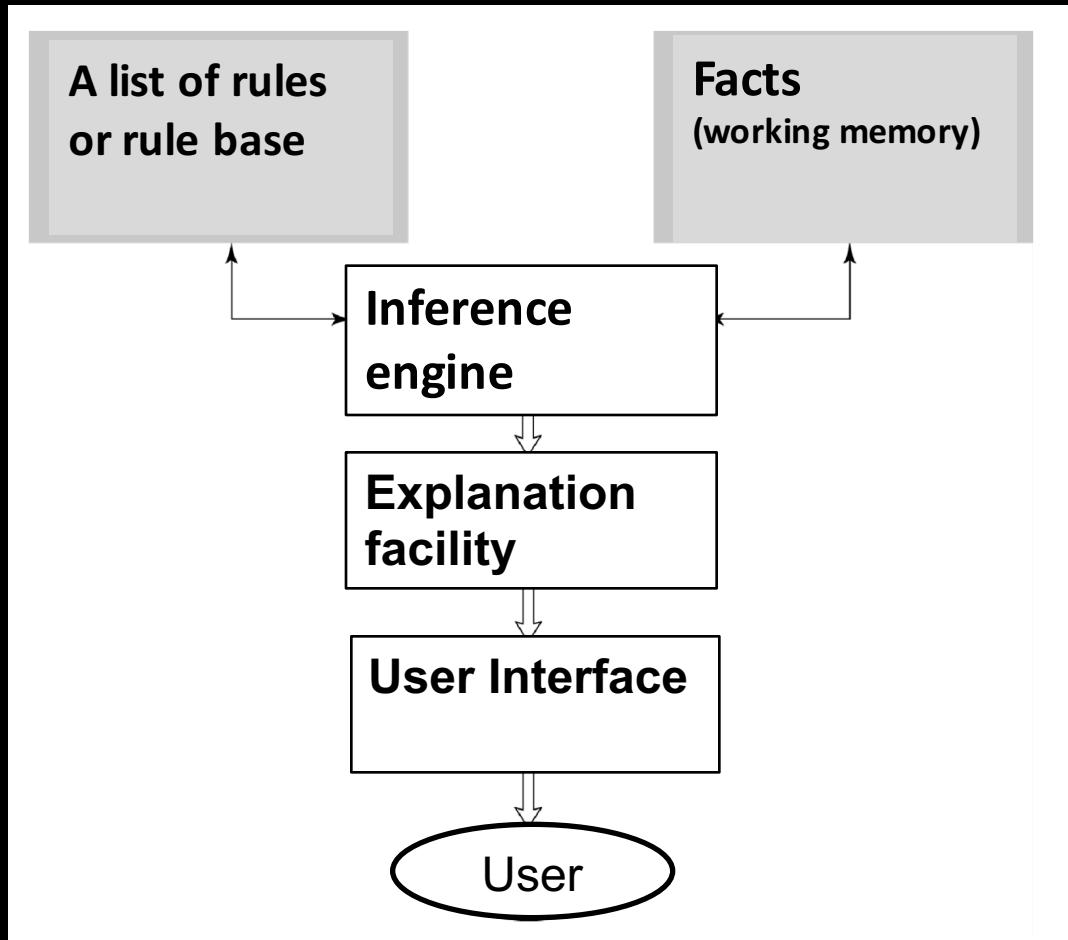
P1-3 SnakeAgent

P1-4 DodgeAgent

Agent



Rule-Based System



Example

If Health < 25% Then

 Drink potion

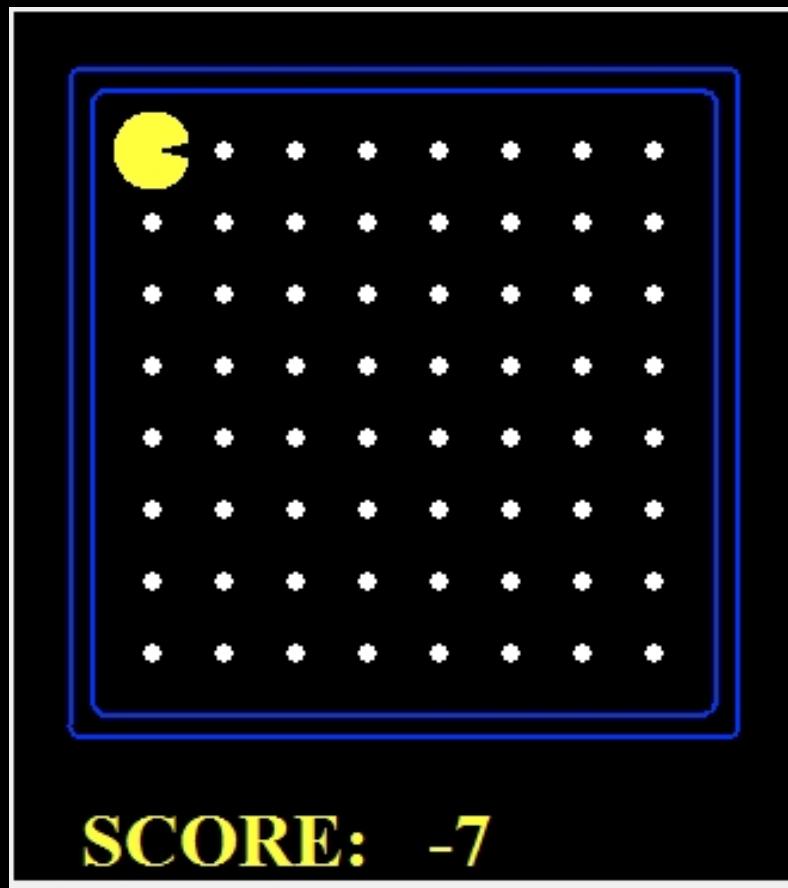
Else If Health < 50% Then

 Run

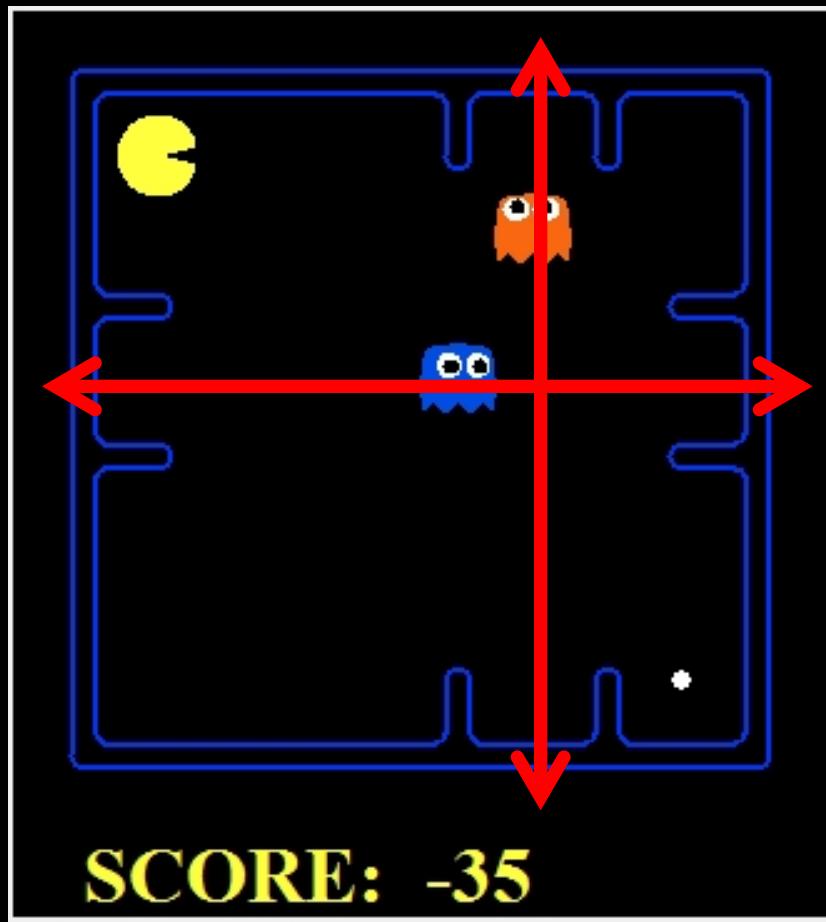
Else

 Attack

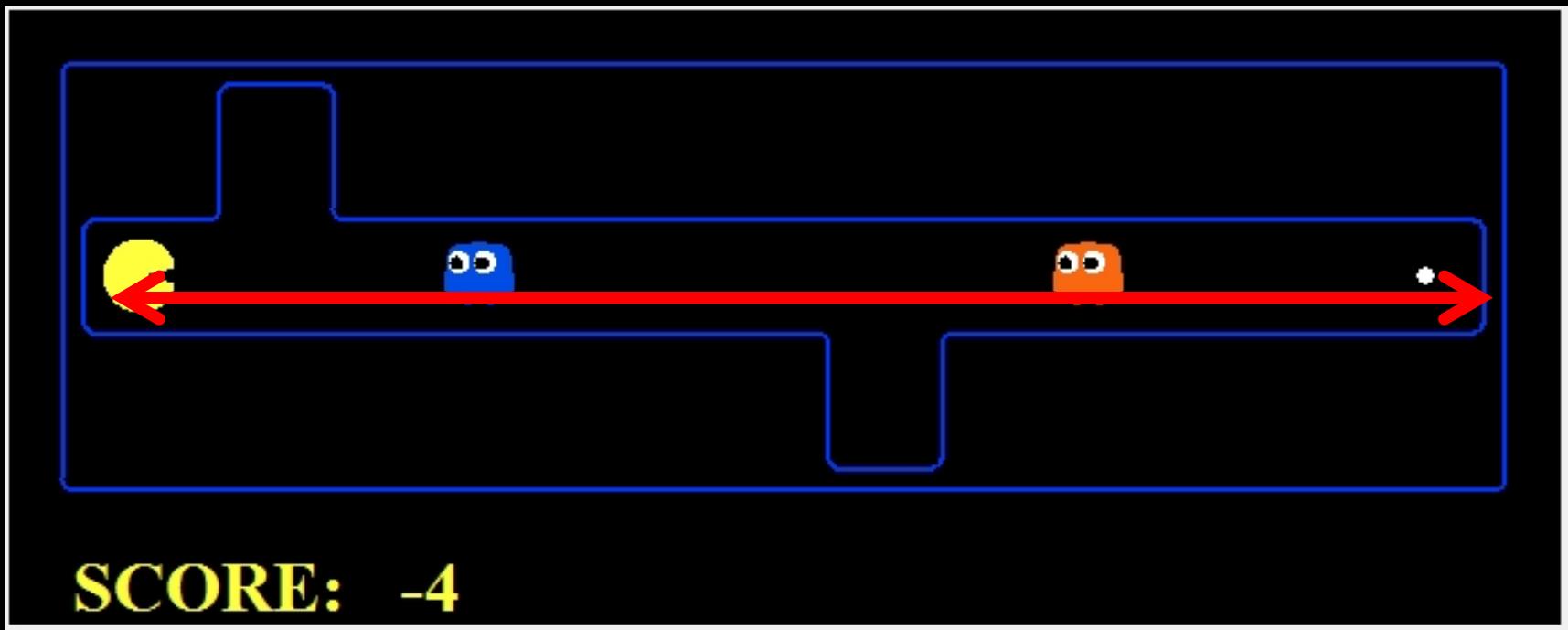
P1-1 CleanerAgent



P1-2 FroggerAgent



P1-3 SnakeAgent



P1-4 DodgeAgent



Hints

1. class GameState in **pacman.py**
getXX() should be useful
2. **game.py**
AgentState, Agent,etc.
3. class Agents in **searchAgents.py**
how agents work



Objectives

P1-1 CleanerAgent (10%)

◦-p CleanerAgent -l P1-1

P1-2 FroggerAgent (10%)

◦-p FroggerAgent -l P1-2 -g StraightRandomGhost

P1-3 SnakeAgent (10%)

◦-p SnakeAgent -l P1-3 -g StraightRandomGhost

P1-4 DodgeAgent (10%)

◦-p DodgeAgent -l P1-4

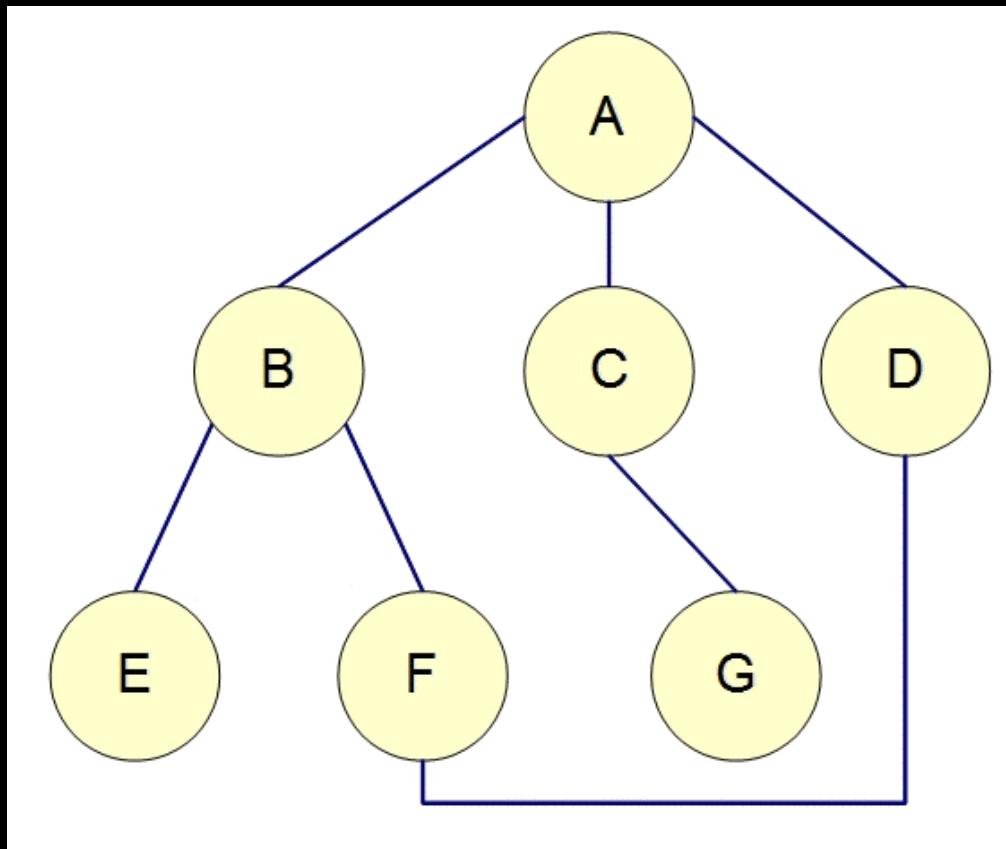
Search

P2-1 DFS

P2-2 BFS

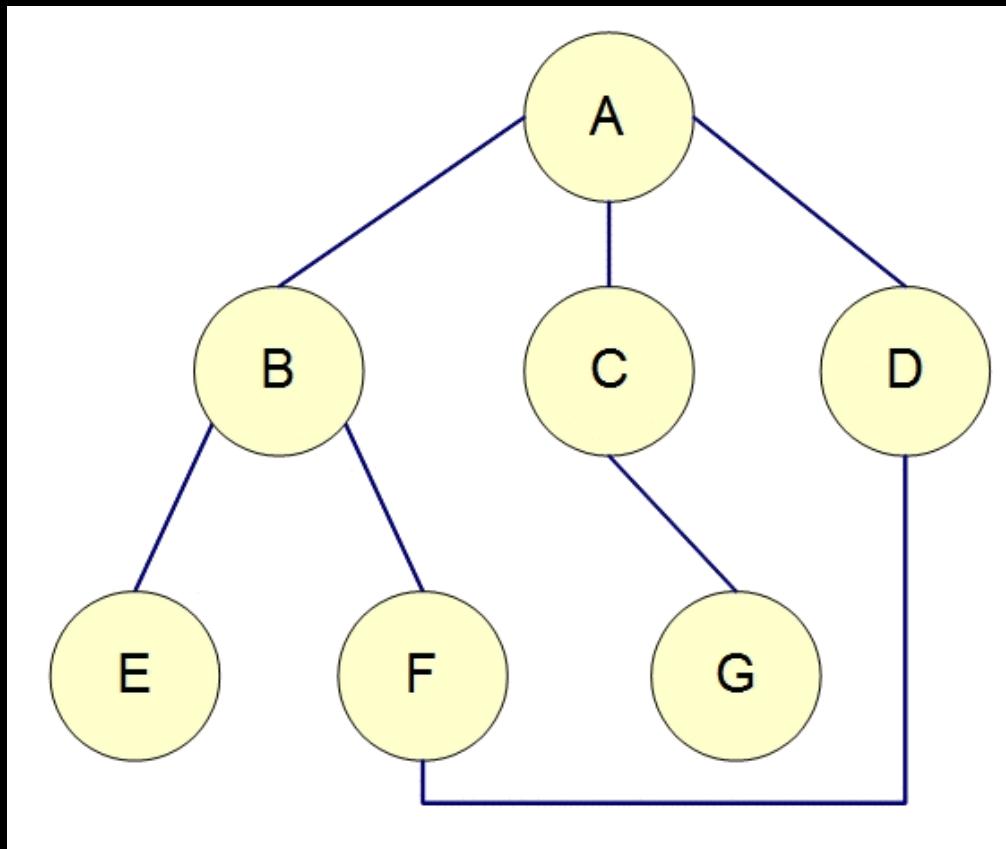
P2-3 A*

P2-1 Depth-First Search



[\(Source\)](#)

P2-2 Breadth-First Search



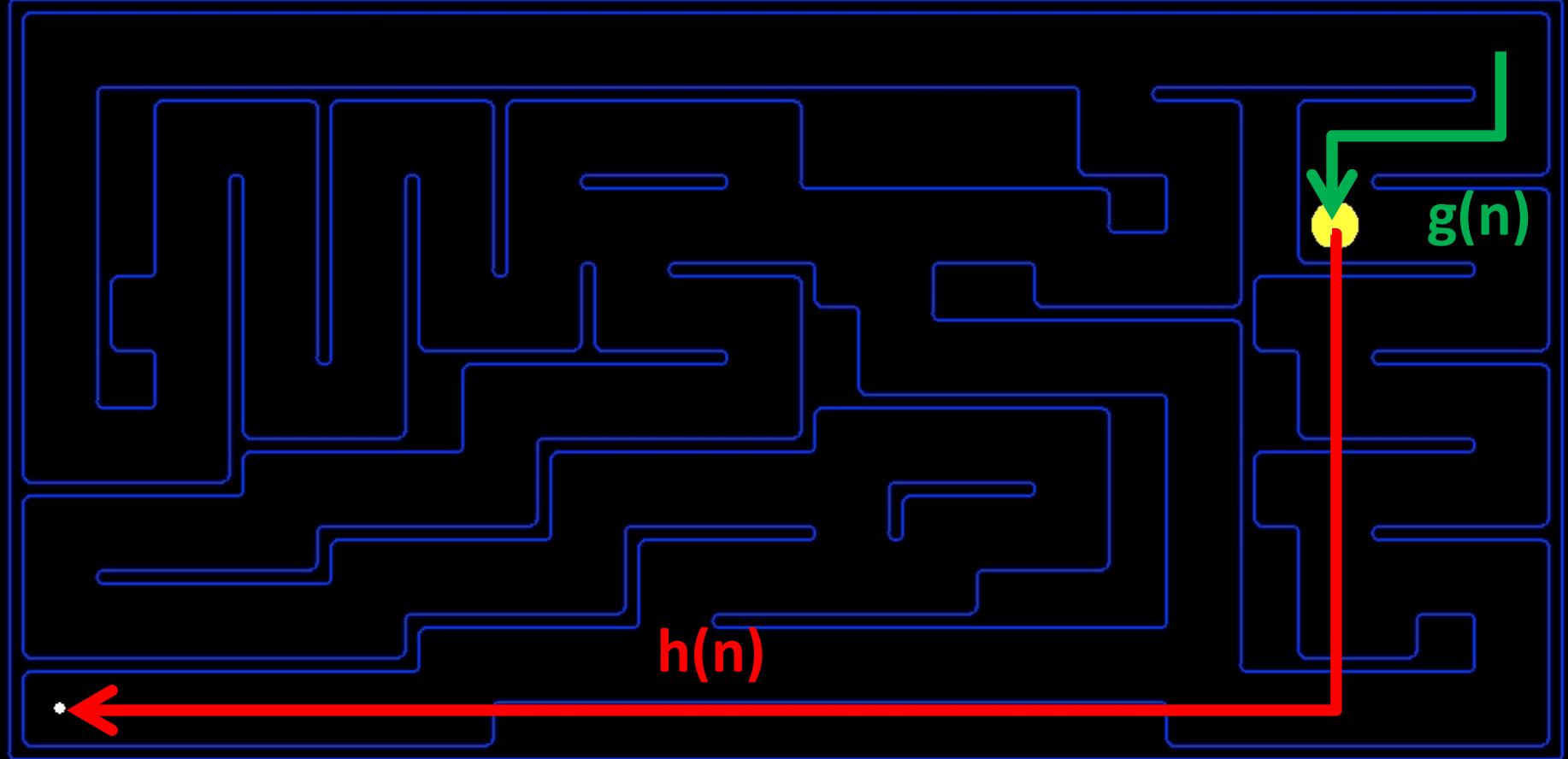
[\(Source\)](#)

P2-3 A* Search

$$f(n) = g(n) + h(n)$$

- $g(n)$: the cost from the start
- $h(n)$: heuristic; it estimates the cost to the goal
 - admissible: $h(n) \leq h^*(n)$ (real cost)
 - For Pacman, we use **Manhattan distance**

P2-3 A* Search



SCORE: -81

Hints

1. util.Stack, util.Queue,
util.PriorityQueue
2. A* algorithm:
getCostOfActions(), heuristic()



Objectives

P2-1 Depth-First Search (20%)

- l mediumMaze -p SearchAgent -a fn=dfs

P2-2 Breadth-First Search (20%)

- l mediumMaze -p SearchAgent -a fn=bfs

P2-3 A* Search (20%)

- l mediumMaze -p SearchAgent -a
fn=astar,heuristic=manhattanHeuristic

Layouts

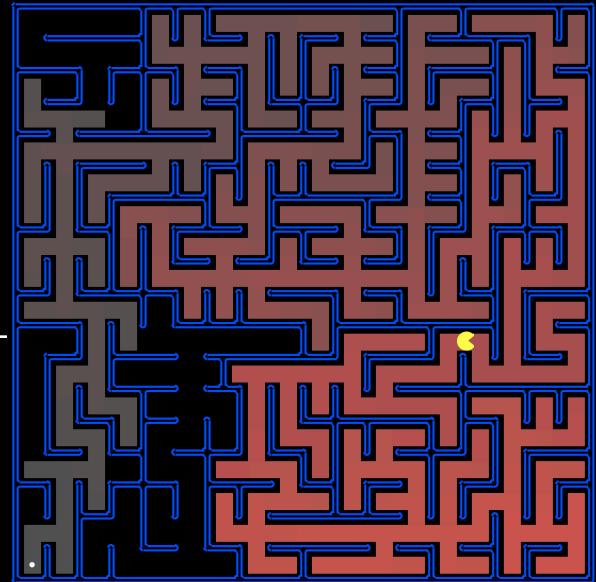
tinyMaze

smallMaze

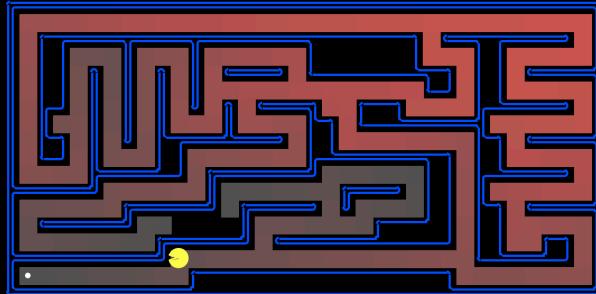
mediumMaze

bigMaze

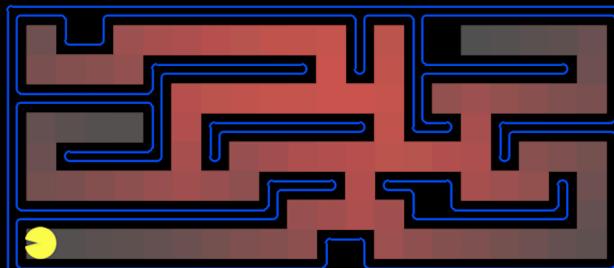
-z 0.5



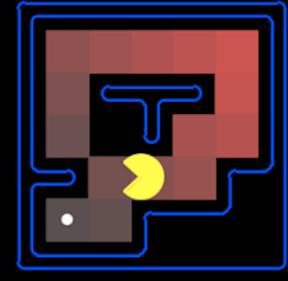
SCORE: -59



SCORE: -57



SCORE: 491



SCORE: -5

Autograder

For debugging your code.

- P2-1: autograder.py -q q1
- P2-2: autograder.py -q q2
- P2-3: autograder.py -q q4

Autograder

Pass

```
Starting on 3-15 at 22:56:32

Question q1
=====

*** PASS: test_cases\q1\graph_backtrack.test
***      solution:          ['1:A->C', '0:C->G']
***      expanded_states:  ['A', 'D', 'C']
*** PASS: test_cases\q1\graph_bfs_vs_dfs.test
***      solution:          ['2:A->D', '0:D->G']
***      expanded_states:  ['A', 'D']
*** PASS: test_cases\q1\graph_infinite.test
***      solution:          ['0:A->B', '1:B->C', '1:C->G']
***      expanded_states:  ['A', 'B', 'C']
*** PASS: test_cases\q1\graph_manypaths.test
***      solution:          ['2:A->B2', '0:B2->C', '0:C->D', '2:D->E']
***      expanded_states:  ['A', 'B2', 'C', 'D', 'E2', 'F']
*** PASS: test_cases\q1\pacman_1.test
***      pacman layout:           mediumMaze
***      solution length: 130
***      nodes expanded:          146

### Question q1: 3/3 ###

Finished at 22:56:32

Provisional grades
=====
Question q1: 3/3
-----
Total: 3/3

Your grades are NOT yet registered. To register your grades, make sure
to follow your instructor's guidelines to receive credit on your project.
```

Submit

Edit and upload files to e3

- **searchAgents.py (Rule-based)**
- **search.py (Search Algorithms)**

Search for “[Project 1] YOUR CODE HERE”
“[Project 2] YOUR CODE HERE”

Group Homework

Deadline: **4/20** 23:59 (3 weeks)

Late policy: 80%

No plagiarism