

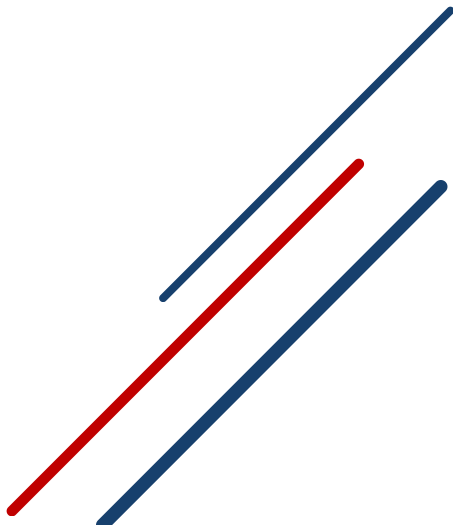
关于在线实验课

- ◆ 助教同理论课
- ◆ 使用腾讯课堂上课，如遇到技术故障将改用腾讯会议
- ◆ 为方便考勤，请同学们将昵称改为“学号-真实姓名”
- ◆ 上课不定时发起签到，请不要迟到早退；
- ◆ 提前下载实验指导书，地址：<https://gitee.com/hitsz-cslab/AI>



人工智能

实验1-搜索策略pacman



实验安排

- ◆ 实验共10学时，2个实验项目(各4学时)，答辩2学时，总成绩30分（30%）

实验内容	分值
答辩	实验1（5分）、实验2（5分）
实验1：搜索	10分（报告、代码、考勤等）
实验2：深度学习	10分（报告、代码、考勤等）

- ◆ 小组形式完成2个实验，每组3或4人
- ◆ 实验课第7~10周，最后一次课5分钟小组答辩；



实验分组

3或4人并登记在共享文档，没有找到队友的联系助教组队；
课程冲突同学换到其他班级，不影响组队

人工智能上课班	授课教师	主平台：腾讯课堂，备用平台：腾讯会议	周次	分组表（按理论课班级安排）
2019级3、4班	房敏、罗文坚	腾讯课堂： https://ke.qq.com/webcourse/4533548/104699832#from=800021724&lite=1&live=1 腾讯会议：454-2613-1303	7，10周 星期一 1-2节 7周 星期六 5-6节	罗文坚老师班级（3、4班）：
2019级1、2班	房敏、徐增林	腾讯课堂： https://ke.qq.com/webcourse/4533552/104699836#from=800021724&lite=1&live=1 腾讯会议：657-3177-4987	7-9周 星期五 3-4节 10周 星期二 11-12节 10周 星期六 7-8节	徐增林老师班级（1、2班）： https://docs.qq.com/sheet/DYUZoaVBibE9XRE5Z?groupUin=eByL8qjczyQdhZhonDqJ3Q%3D%3D&tdsourceta&tab=BB08J2
2019级7、8班	郑海刚、房敏	腾讯课堂： https://ke.qq.com/webcourse/4582389/104750156#from=800021724&lite=1&live=1 腾讯会议：456-5214-3848	7，9周 星期一 9-10节 9周 星期五 1-2节 10周 星期二 5-6节	
2019级9、10班	郑海刚、汤步洲	腾讯课堂： https://ke.qq.com/webcourse/4582421/104750188#from=800021724&lite=1&live=1 腾讯会议：456-5214-3848	7周 星期一 3-4节 8周 星期三 7-8节 9-10周 星期六 3-4节 10周 星期四 9-10节	汤步洲老师班级（8、9、10班）： https://www.kdocs.cn/l/cs20ptGLL13q
2019级5、6班	郑海刚、苏敬勇	腾讯课堂： https://ke.qq.com/webcourse/4582407/104750174#from=800021724&lite=1&live=1 腾讯会议：456-5214-3848	7，10周 星期一 1-2节 7周 星期六 5-6节 9周 星期六 1-2节	苏敬勇老师班级（5、6、7班）： https://docs.qq.com/sheet/DTXh4UmJGd2FaeHVh

华为云代金券

确认在共享文档填写了[华为云账号](#)！！



python环境统计（通过腾讯课堂答题）

- ☐ A 已经安装好python3的环境
- ☐ B 没有安装python3的环境

提交

python学习

- ◆ python基础自主学习，后续深度学习实验继续使用python完成，可以参考“廖雪峰python教程”。
- ◆ 学习重点：python安装、数据类型、函数、模块等基础即可



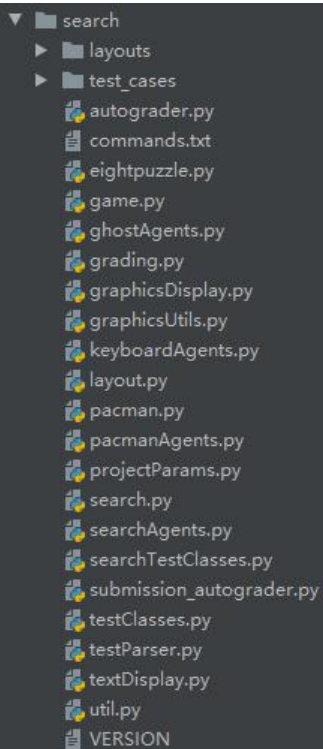
实验内容

要求采用且不限于课程介绍的各种搜索算法，编写一系列程序解决吃豆人（pacman）游戏中定义的问题1-8。



pacman

pacman是加州大学伯克利分校开源的人工智能实验项目



文件说明

需要阅读代码的文件

文件	主要功能
pacman.py	吃豆人游戏的主程序
game.py	吃豆人游戏的运行逻辑
util.py	搜索策略可以用到的数据结构

需要完善代码的文件

文件	主要功能
search.py	待实现的搜索算法
searchAgents.py	待实现的智能体和相关Problem



待解决的8个问题

Q1: 应用dfs找到一个特定的位置的豆

```
python pacman.py -l mediumMaze -p SearchAgent
```

```
python pacman.py -l bigMaze -z .5 -p SearchAgent
```

Q2: 应用bfs找到一个特定的位置的豆

```
python pacman.py -l mediumMaze -p SearchAgent -a fn=bfs
```

```
python pacman.py -l bigMaze -p SearchAgent -a fn=bfs -z .5
```

Q3: 应用代价一致算法找到一个特定的位置的豆

```
python pacman.py -l mediumMaze -p SearchAgent -a fn=ucs
```



待解决的8个问题

Q4: 应用A* 算法找到一个特定的位置的豆

```
python pacman.py -l bigMaze -z .5 -p SearchAgent -a fn=astar,heuristic=manhattanHeuristic
```

Graph Search Pseudo-Code

```
function GRAPH-SEARCH(problem, fringe) return a solution, or failure
  closed ← an empty set
  fringe ← INSERT(MAKE-NODE(INITIAL-STATE[problem]), fringe)
  loop do
    if fringe is empty then return failure
    node ← REMOVE-FRONT(fringe)
    if GOAL-TEST(problem, STATE[node]) then return node
    if STATE[node] is not in closed then
      add STATE[node] to closed
      for child-node in EXPAND(STATE[node], problem) do
        fringe ← INSERT(child-node, fringe)
      end
    end
  end
```



待解决的8个问题

Q5: 找到所有的角落——基于BFS的角落问题 (完成CornersProblem)

```
python pacman.py -l tinyCorners -p SearchAgent -a fn=bfs,prob=CornersProblem
```

```
python pacman.py -l mediumCorners -p SearchAgent -a fn=bfs,prob=CornersProblem
```

Q6: 找到所有的角落——基于A*的角落问题 (构建启发函数)

```
python pacman.py -l mediumCorners -p AStarCornersAgent -z 0.5
```

Number of nodes expanded	Grade
more than 2000	0/3
at most 2000	1/3
at most 1600	2/3
at most 1200	3/3



待解决的8个问题

Q7: 吃掉所有的豆子——食物搜索问题 (A*算法, 构造启发函数)

`python pacman.py -l trickySearch -p AStarFoodSearchAgent` (会stuck, 什么原因?)

Number of nodes expanded	Grade
more than 15000	1/4
at most 15000	2/4
at most 12000	3/4
at most 9000	4/4 (full credit; medium)
at most 7000	5/4 (optional extra credit; hard)

Q8: 次最优搜索——任意食物搜索问题 (优先吃最近的豆子加快搜索)

`python pacman.py -l trickySearch -p ClosestDotSearchAgent` (跟Q7同样的地图测试)

`python pacman.py -l bigSearch -p ClosestDotSearchAgent -z .5`



待解决的8个问题

python autograder.py测评

```
(venv) D:\HITSZ\github\ai_search\search>python2 autograder.py
Starting on 10-16 at 9:12:01

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->E2', '0:E2->F', '0:F->G']
***   expanded_states:    ['A', 'B2', 'C', 'D', 'E2', 'F']
```



实验报告

- ◆ 报告内容及格式要求见指导书。



提交方式

- ◆ 独立完成实验报告，严禁抄袭，只需写自己完成的部分，代码提交小组最终完整版本（只需要提交新增和修改的）。

- ◆ 提交截止时间答辩前一天，具体见作业提交系统

<http://grader.tery.top:8000/#/courses>

