Notes:

Notes for independent pacman study:

* agents that plans ahead, vs reflex agents
  + plan ahead agent don’t have to write complex conditions to execute an action, can write generic function.
  + reflex agent, code like if-then statement, don’t plan ahead. act based on current and past memory.

Find about peusuiave and evader topics online:

-my approach for the persuade evader research topic:

First make the single agent able to travel optimally. The research on multiple agents.

-Berkeley open course project: have the code for graphics, but search algorithm I have to write myself.

-completed A star search algorithm with one heuristic method.

-trying to find more efficient heuristic method.

-study how to incorporate multiple agents behavior.

Project performance results

**A star search to find the optimal path to goal state,**

more red the color, earlier the expantion, note:more dark the color, later the expansion happens

command:  -l bigMaze -z .5 -p SearchAgent -a fn=astar,heuristic=manhattanHeuristic

**result for pacman eating all dots at once:**

**with no heuristic**, == uniform cost search

Command Line : -l tinySearch -p AStarFoodSearchAgent

Path found with total cost of 27 in 7.2 seconds

Search nodes expanded: 5366

Pacman emerges victorious! Score: 573

**heuristic 1:1/18/2014,**

description: manhattan distance for each dot to its closest dot, and its closest dot, until all the dot is travelled.

**result:**

Command Line : -l tinySearch -p AStarFoodSearchAgent

Path found with total cost of 27 in 0.8 seconds

Search nodes expanded: 1250

Pacman emerges victorious! Score: 573

Conclusion: much less node expanded, result is much faster execusion.

# Notes: 2/7/2014:

-found a better heuristic that reduce the search time

-and found the original heuristic is not admissible.

Performance result (in a bigger map) below:

**Eat all the dots, trickysearch map:**

No heuristic: uniform cost search:

Command Line : -l trickySearch -p AStarFoodSearchAgent

Path found with total cost of 60 in 49.1 seconds

Search nodes expanded: 17209

Pacman emerges victorious! Score: 570

Heuristic 1: manhattan distance for each dot to its closest dot, and its closest dot, until all the dot is travelled.

Command Line : -l trickySearch -p AStarFoodSearchAgent

Path found with total cost of 68 in 26.6 seconds

Search nodes expanded: 12762

Pacman emerges victorious! Score: 562

result: This is not admissible, Did not find the optimal path.

heuristic 2: prim algorithm:

Command Line : -l trickySearch -p AStarFoodSearchAgent

Path found with total cost of 60 in 16.1 seconds

Search nodes expanded: 7743

Pacman emerges victorious! Score: 570

-Note that with relative big size map, the search already takes quiet a while. This is clearly not scarable to bigger map problems and multi agent problem

-another approach by using reflex agent. Need evaluation function, reaction time can be reduced by limiting the length of the tree.

Next meet up goal: try to implement reflex agent to see if behavior of multiagent can exist.

# Note 2/25/2014:

Multi-agent pacman - reflex agent

Write code for the reflex agent. The demonstration shows that the computation time can be very short and pacman do not have to wait, compare to the plan ahead agent.

The pacman currently does not have an actual plan to eat dot, but more reflex agent. It will show something it is confused on eating dots.

**Pacman against a random ghost on an open map:**

command: -p ReflexAgent -l openClassic -n 100 -q

result:

Average Score: 1233.89

Scores: 1239, 1220, 1207, 1220, 1239, 1234, 1250, 1237, 1201, 1207, 1203, 1210, 1224, 1425, 1231, 1218, 1239, 1217, 1220, 1223, 1247, 1231, 1221, 1233, 1237, 1205, 1237, 1204, 1209, 1222, 1213, 1243, 1220, 1221, 1234, 1233, 1219, 1247, 1215, 1224, 1434, 1242, 1222, 1216, 1245, 1199, 1234, 1236, 1236, 1239, 1231, 1249, 1230, 1221, 1236, 1227, 1250, 1225, 1394, 1237, 1231, 1228, 1222, 1236, 1229, 1241, 1211, 1233, 1235, 1221, 1242, 1233, 1241, 1211, 1232, 1219, 1208, 1217, 1241, 1226, 1225, 1223, 1418, 1228, 1225, 1240, 1212, 1228, 1235, 1191, 1229, 1213, 1242, 1213, 1225, 1243, 1186, 1223, 1222, 1229

**Win Rate: 1.0**

Record: Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win

**against two random ghost in the narrow space map:**

command: --frameTime 0 -p ReflexAgent -k 2 -n 100 -q

result:

Average Score: 1190.81

Scores: 1866, 173, 1373, 1503, 1652, 1521, 1626, 1463, 1213, 1744, 1421, 1545, 1190, 471, 1648, 1346, 1736, 1406, 1435, 1324, 1562, 1941, 1284, 1534, 44, 1382, 1531, 1867, 1841, -304, 1702, 1260, 264, -442, 126, 1353, 1203, 1517, 1495, -107, 1420, 732, 103, 1214, 1420, 1214, 1938, 1061, 1319, 1814, 1329, 1462, -273, -1, 1407, 1239, 497, 1273, 692, 1518, 1538, 1226, 1645, 1289, 1545, 190, 1696, 1627, 1232, 1416, 1160, 1204, 224, 293, 1208, 1751, 1253, 1269, 1663, 1316, 1912, 322, 1309, 1047, 1824, 13, 259, 1504, 1133, -301, 1418, 1716, 1692, 1291, 1551, 1652, 1456, 1662, 1405, 1384

**Win Rate: 0.79**

Record: Win, Loss, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Loss, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Loss, Win, Win, Win, Win, Loss, Win, Win, Loss, Loss, Loss, Win, Win, Win, Win, Loss, Win, Loss, Loss, Win, Win, Win, Win, Win, Win, Win, Win, Win, Loss, Loss, Win, Win, Loss, Win, Loss, Win, Win, Win, Win, Win, Win, Loss, Win, Win, Win, Win, Win, Win, Loss, Loss, Win, Win, Win, Win, Win, Win, Win, Loss, Win, Win, Win, Loss, Loss, Win, Win, Loss, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win

**run in the narrow map with one directional ghost:**

command: -p ReflexAgent -k 1 -g DirectionalGhost -n 100 -q

result:

Average Score: 1190.85

Scores: 227, 1242, 1471, 1503, 1295, 1469, 412, 1486, 1433, 1508, 1085, 154, 1476, 1318, 1345, 1249, 1170, 1298, 1371, 1178, 1293, 1500, 1235, 1304, 1476, 1143, -36, 1265, 1540, 1523, 1239, 1343, 157, 1061, 1268, 1339, 1445, 1278, 1271, 29, 1332, 256, 1218, 1218, 1252, 1217, 1227, 1314, 1360, 1237, 180, 1313, 103, 1540, 1494, 1440, 1136, 1172, 375, 1301, 1487, 1466, 1502, 1507, 1350, 1284, 1173, 1259, 1239, 1673, 1186, 185, 1282, 1741, 1277, 1264, 1254, 1331, 1237, 1237, 1454, 1329, 1479, 1280, 1229, 1267, 1330, 1294, 1511, 1473, 87, 1543, 422, 1529, 1495, 1397, 1247, 1284, 1230, 1223

**Win Rate: 0.87**

Record: Loss, Win, Win, Win, Win, Win, Loss, Win, Win, Win, Win, Loss, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Loss, Win, Win, Win, Win, Win, Loss, Win, Win, Win, Win, Win, Win, Loss, Win, Loss, Win, Win, Win, Win, Win, Win, Win, Win, Loss, Win, Loss, Win, Win, Win, Win, Win, Loss, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Loss, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Win, Loss, Win, Loss, Win, Win, Win, Win, Win, Win, Win

**Narrow map, against two direction ghost:**

command: -p ReflexAgent -k 2 -g DirectionalGhost -n 100 -q

Average Score: 522.96

Scores: -191, 320, 525, 510, 320, -8, 48, 287, 107, 283, -18, 49, -196, -39, 1345, 61, 1529, 272, -37, 1490, -183, 1432, 181, -40, 172, 395, 308, 1335, 166, 1483, 171, 515, 60, 130, 1087, -122, 33, 662, 464, 322, 494, -286, 167, 647, -171, 164, -67, 476, -34, 1330, 1437, 1527, -263, 1513, 406, -290, 1311, 669, 661, 662, 1321, 278, 1418, 474, 1306, 211, 1729, 399, 1491, 130, 379, 429, -33, 1275, 1317, 308, 1322, 237, 37, 88, 59, 1495, 1422, 436, 1276, 185, 207, 1337, 1321, 1320, 52, 198, 442, 1292, -272, 369, 1229, -167, 187, 211

**Win Rate: 0.27**

Record: Loss, Loss, Loss, Loss, Loss, Loss, Loss, Loss, Loss, Loss, Loss, Loss, Loss, Loss, Win, Loss, Win, Loss, Loss, Win, Loss, Win, Loss, Loss, Loss, Loss, Loss, Win, Loss, Win, Loss, Loss, Loss, Loss, Loss, Loss, Loss, Loss, Loss, Loss, Loss, Loss, Loss, Loss, Loss, Loss, Loss, Loss, Loss, Win, Win, Win, Loss, Win, Loss, Loss, Win, Loss, Loss, Loss, Win, Loss, Win, Loss, Win, Loss, Win, Loss, Win, Loss, Loss, Loss, Loss, Win, Win, Loss, Win, Loss, Loss, Loss, Loss, Win, Win, Loss, Win, Loss, Loss, Win, Win, Win, Loss, Loss, Loss, Win, Loss, Loss, Win, Loss, Loss, Loss

Future study actions:

Have en agent that can plan ahead. Goal: maybe a min-max tree implementation. So pacman can evaluate what is best way to go and flow through the plan.