COMP341 Introduction to Artificial Intelligence
HW1

This programming homework will test your knowledge and your implementation abilities of what you have learned in the uninformed and informed search parts of the class. This homework has two parts; a programming part and a written report part. The written report will require you to interpret the output of the programming part.

The programming part of the homeworks will usually follow the Berkeley CS188 Spring 2014 pacman projects at `http://ai.berkeley.edu/` closely. This homework includes P1:Search.

This homework must be completed individually. Discussion about algorithms and data structures are allowed but group work is not.

However, you might find yourself having trouble implementing the algorithms. I am going allow you to copy or look at someone else's code as long as you credit the person or the website! In this case I will grade you differently, as described in the next part.

Any academic dishonesty, which includes you taking someone else's code but not properly crediting the source, will not be tolerated.

Grading

The code part and the report will have 2:1 weight ratio in your submission (programming 2/3, report 1/3). In order for the report to be graded, corresponding programming parts need to be submitted.

In case you have copied/used someone else's code as stated above, you will only be graded for your report. In this case you will be eligible for up to half the credit, i.e., the report will have 1/2 weight. You should not submit your code in this case **AND** properly credit the source in your report. Reports without proper credit will not be graded.

Finally, we are going to compare your code to previous year's submissions, each other's and other online sources known to us. If your code's similarity level is above a certain threshold, your code will be scrutinized and you might end up getting a 0.

Part 1: Programming

You are going to do the 8 programming questions about search given here: `http://ai.berkeley.edu/search.html`. Feel free to use the forum in the blackboard site for general discussion however remember there is no group work.

You are only required to change *search.py* and *searchAgents.py.* If you have any issues with other parts of the code let your instructor or TA know ASAP, even if you manage to solve your problem. Use the data structures in *util.py* for the autograder to work properly. If the you think you have the right answer but the autograder is not giving you any points, try to run it on individual questions (look at P0 for details on how to use the *autograder.py*). The file *autograder.py* is missing an "import pprint", which might cause issues in some setups.

Part 2: Report

This part includes answering the following questions based on your programs' output on the given pacman tests. Look specfically at *bigMaze* and the *openMaze* for stark comparisons.

You are expected to answer the questions concisely. This means maximum five sentences per question. Answers that are longer than 150 words will only be graded for the first 150 words or so. It is okay if you over-generalize, as long as your direction is clear and correct. Note that some answers are already in the provided link to the Berkeley site!

Written Q1:

What are some differences between DFS and BFS in terms of path cost and number of expanded nodes? Which one would you pick when and why?

Written Q2:

What are some differences between UCS and A* in terms of path cost and number of expanded nodes? Which one would you pick when and why?

Written Q3:

Comment on your choice of state in the four corners problem. Why does it allow you to solve the problem?

Written Q4:

Comment on your choice of heuristic in the four corners problem. Why did you settle on that heuristic? Why is it admissible and consistent?

Written Q5:

Comment on your choice of heuristic in the eating all the dots problem. Why did you settle on that heuristic? Why is it admissible and consistent?

Written Q6:

What are some practical differences between a consistent and an inadmissible heuristic, in terms of path cost and number of expanded nodes? Which one would you pick when and why?

Submission

You are going to submit a compressed archive through the blackboard site. The file can have *zip*, *tar* or *tar.gz* format. This should extract to a folder with your student ID which includes *search.py*, *searchAgents.py* and *report.pdf.* **DO NOT ARCHIVE THE ENTIRE FOLDER.** Let us know if you have any problems.