Mark Demore II

CSCE523

Assignment #1

1. A) As far as intelligence is concerned, I believe Qualia is the best framework to view it. Everyone has their own perception of events and conditions, whether they me be skewed in one way or another, like having the gene that makes cilantro taste like soap, for example. Human Cognitive Architecture and Physicalism are limited because they are rooted in very matter of fact views of the human mind. Intelligent beings can handle concepts beyond the merely physical, and merely reducing the way that our brain is wired to learn does not encompass all the aspects of intelligence.

B) Based on this definition, artificial ‘intelligence’ is not truly possible. A very close artificial intelligence could be created by adding randomization and plenty of other variables, however it could never obtain the level of abstraction that the human mind is capable of. Yet, I don’t think this level of ‘intelligence’ is necessary for most problems.

1. A) While all of Turing’s concerns were valid in his time, I think the only one that still stands is the argument from extra-sensory perception. Turing doesn’t really refute this argument either. However, I think some sort of “extra-sensory perception” could be implemented by accepting atmospheric noise as an input.

B) I think new arguments could be proposed, all dependent upon the definition of intelligence. For example, the argument I made in response to question 1 on the basis of Qualia. The Chinese Room argument is certainly valid, but again hinges on the debate of what the definition of intelligence is.

C) I cannot think of a better test that would address the issues that arise with the Chinese Room argument. However, I do not think these philosophical questions are terribly important. Artificial Intelligence and the continued development by striving to make things more perfect continue to create new ways of addressing problems.

1. This does not mean that strong AI is impossible. Either new ways of representing these problems will be developed so that they could be computable, or this set of problems will simply be excluded. All other characteristics of strong AI are possible and achievable.
2. A) Yes, a reflex agent can still perform in an unknown environment, as long as it is equipped to sense obstacles, boundaries, and dirt. By responding appropriately when an obstacle is detected and ordering movement, the agent will eventually explore the whole space. Likely, a level of randomization would be needed to prevent the agent from moving redundantly.

B) This agent would perform poorly in a large environment and also one with many obstacles, making it difficult to reach the entire space in a reasonable number of moves.

C) A reflex agent with state can outperform a simple reflex agent because it has more information with which it can make decisions, by keeping track of what it has already done and how the environment changed with it.

1. A) OPEN: CLOSED:

A, B, C S

B, C, D, E S, A

C, D, E, I, F S, A, B

D, E, I, F, G S, A, B, C

E, I, F, G, H S, A, B, C, D

I, F, G, H S, A, B, C, D, E

F, G, H, J S, A, B, C, D, E, I

G, H, J S, A, B, C, D, E, I, F

H, J S, A, B, C, D, E, I, F, G

J S, A, B, C, D, E, I, F, G, H

X S, A, B, C, D, E, I, F, G, H, J

B) OPEN: CLOSED:

A, B, C S

D, E, B, C S, A

H, E, B, C S, A, D

E, B, C S, A, D, H

B, C S, A, D, H, E

I, F, C S, A, D, H, E, B

J, F, C S, A, D, H, E, B, I

X, F, C S, A, D, H, E, B, I, J

C) OPEN: CLOSED:

A, B, C S

A, B, G S, C

D, B, E, G S, C, A

H, B, E, G S, C, A, D

B, E, G S, C, A, D, H

F, E, G, I S, C, A, D, H, B

E, G, I S, C, A, D, H, B, F

G, I S, C, A, D, H, B, F, E

I S, C, A, D, H, B, F, E, G

J S, C, A, D, H, B, F, E, G, I

X S, C, A, D, H, B, F, E, G, I, J

1. Compile Instructions: Add the files in /src to an Eclipse project and click run. (I believe the only file I altered was Board.java and I added MySearch.java).

Description: My solution implements the A\* Search, using a goal and blocking heuristic, as well as the node\_count as a ‘cost-so-far’ value. I added a few variables to the Board object to help keep track of these and used a PriorityQueue and HashMap to keep track of the open and visited lists. The following are the results of my implementation on the Data and Killers files.

Data:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Board 1** | **Board 2** | **Board 3** | **Board 4** | **Board 5** |
| 0.002 seconds | 0.008 seconds | 0.014 seconds | 0.135 seconds | 0.131 seconds |
| 1 node | 6 nodes | 55 nodes | 1156 nodes | 2535 nodes |

Killers:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Board 1** | **Board 2** | **Board 3** | **Board 4** | **Board 5** |
| 0.205 seconds | 0.102 seconds | 0.183 seconds | 0.248 seconds | 0.596 seconds |
| 3535 nodes | 2832 nodes | 3895 nodes | 7260 nodes | 13447 nodes |