**Uninformed and Informed Search**

**Artificial Intelligence(CS5100) - HW2**

Nephi Calvin Bandela

1. Search Techniques Implemented:
2. Uninformed Search:
3. Depth First Search:
   1. Algorithm:

*1. Initialize Stack*

*2. Push Start Node into Stack*

*3. Mark Start Node as Visited*

*loop*

*4. Get Top*

*5. Get Children of Top*

*6. If Child Node Unvisited*

*7. Mark Unvisited child as visited*

*8. Push Unvisited child onto Stack*

*9. If no child nodes - Retract Move*

*end loop*

* 1. Complexity: b-branching factor, m: Max depth from start, d: Goal depth or solution depth

|  |  |
| --- | --- |
| Time Complexity | Space Complexity |
| O(b^m) – May visit all nodes-b+b^2+…….+b^m | O(bm) – m(b-1) -Entire Space Storage |

* 1. Environment: Partially Observable

1. Informed Search:
2. Greedy Best First Search
   1. Algorithm:

*1. Retrieve all Node Indices of Targets*

*2. Initialize Visited, HashMap to store heuristics and their corresponding Node Indices*

*3. The heuristics- Manhattan Distance are sorted in ascending order.*

*4. The HashMap consists of successors of current Node arranged in increasing orders*

*5. Return the Node with the best heuristic as the target node*

* 1. Complexity: b-branching factor, m: Max depth from start, d: Goal depth or solution depth. h(n)- Heuristic Function

|  |  |
| --- | --- |
| Time Complexity | Space Complexity |
| O(b^m) – May visit all nodes-b+b^2+…….+b^m but due to the heuristic function h(n) -Manhattan Distance – It might be almost O(b^d) | O(bm) – m(b-1) -Entire Space Storage |

* 1. Environment: Fully Observable

1. References:
   1. <http://www.codeproject.com/Articles/32212/Introduction-to-Graph-with-Breadth-First-Search-BF>
   2. Stuart Russell and Peter Norvig. Artificial Intelligence: A Modern Approach.