Question 1.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Algorithm | Time Complexity | Space Complexity | Complete? | Optimal? |
| BFS (Breadth First Search) | O(V + E) where V is the number of nodes and E is the number of edges. | O(V) where V is the number of nodes. | Yes. | Yes, when the path cost is equal. |
| UCS (Uniform Cost Search) | O(b^(1+C/ε)) where b is the branching factor. | O(b^(1+C/ε)), where b is the branching factor. | Yes, when b is finite. | Yes. |
| DFS (Depth First Search) | O(V + E) where V is the number of nodes and E is the number of edges. | O(V) where V is the number of nodes. | No. | No |
| DLS (Depth Limited Search) | O(V^L) where V is the number of nodes and L is the level limit. | O(bL) where b is the branching factor and L is the level limit. | Yes if the goal node is not below the level limit. | No. |
| IDS (Iterated Deepening Search) | O(b^L) where b is the branching factor and L is the level limit. | O(bL) where b is the branching factor and L is the level limit. | Yes, when b is finite. | Yes. |
| A\* | O(b^d) where b is the branching factor and d is the number of nodes on the path. | O(b^d) where b is the branching factor and d is the number of nodes on the path. | Yes. | Yes. |

Question 2.

mission\_complete:

Breadth First Search: 38

Depth First Search: 23

Depth Limited Search: 24

moveToSample:

Breadth First Search: 4

Depth First Search: 53

Depth Limited Search: 4

removeSample:

Breadth First Search: 15

Depth First Search: 30

Depth Limited Search: 8

returnToCharger:

Breadth First Search: 7

Depth First Search: 12

Depth Limited Search: 12

The total states for the decomposition for breadth first search was 26 vs the total one which was 38 so less states were generated. The total states for the decomposition for depth first search was 95 vs the total one which was 23 so significantly more states were generated. The total states for the decomposition for depth limited search was 24 vs the total one which was 24 so those stayed the same.

Question 3.

A\* Search: 31 states

Uniform Cost Search: 31 states

Question 4.

1. One engineering advance that led to Deep Blue’s success was enhancing the chess chip. They improved it so that it had a redesigned evaluation function which went from 6400 features to over 8000 and they also added hardware repetition detection which generates all the moves that attack the opponents’ pieces. Another improvement was that they more than doubled the number of chess chips in the system and used the newer generation of SP computer to support the higher processing demands. They also developed a set of software tools which helped to debug and prepare for the match.

The improvement of the evaluation function is specific to chess as it computes a score for a chess position and chess behavior is specific to chess such that no other game follows similar rules/restrictions. Similarly, the chess chip is chess specific as it consists of three main parts: the move generator, the evaluation function, and the search control. The move generator checks both evasion and attacking moves, the evaluation function computes a score for a specific chess position in a single clock cycle, and the search control was in charge of searching the tree to select a move to play. I couldn’t quite find more information on the software tools developed to help debug but I’d assume that they’re more universal so they’d be able to be transferred to other problems.

1. StockFish and Deep Blue use a brute force search method which is enhanced by alpha-beta pruning and evaluates positions using evaluation functions programmed with specific heuristics. This approach ends up evaluating an enormous number of nodes to find the best move as it relies on sheer computation power to explore a wide range of possibilities. On the other hand, AlphaZero uses Monte Carlo Tree Search and a deep neural network to evaluate move/position. With the use of a deep neural network, it reduced the need for extensive search as it evaluates positions more accurately and efficiently so it explores fewer nodes but with a stronger evaluation. AlphaZero searches 80 thousand positions per second in Chess compared to 70 million for StockFish. AlphaZero also had a more effectively thinking time than StockFish.