Question one :

Depth first search implemented with path checking can only ensure that state c is not equal to the state reached by any ancestor of c along this path. Each of paths are checked in isolation. However, if we implement DFS by full cycle checking, we keep track of all states added to OPEN during the search. Which mean we can ensure state c is not equal to any previously seen state. The difference is DFS with full cycle checking will expand fewer node but higher space complexity than DFS with path checking only.

Question two:

(a) 4

(b)n^2

(c) $O(3^{n^2})$

Question three:

Since there is no cycle checking. We can assume the maximum number of states produced by the successor function is 4. Define $\boldsymbol{n}$ as the maximum number of node on the open list of A* search. Worst-case space complexity of A* is equal to the worst-case space complexity of uniform cost search, which with an upper bound of $O(b^{\{\frac{c^*}{\varepsilon}+1\}})$. *($c^*$ indicates cost of the optimal solution, $\mathcal{E}$ is a lower bound on any move.)*. Therefore, in worst-case space complexity of A*, we have $\boldsymbol{n} \geq 4^{\{\frac{c^*}{cmin}+1\}}$. Therefore, maximum number of node before A* find a solution is less than $4^{\{\frac{c^*}{cmin}+1\}}$.

Question four:

(a) It is optimal.

(b) It is not-optimal, upper bound ratio is 3.

(c) It is optimal.