# COMP237 Assignment 2 – Written Response

# Student: Yuen Kwan LI (301228849)

## Question 1

The connections of the graph data

A screenshot of a computer

Description automatically generated with medium confidence

### 1st run: Dolly wishes to get introduced to YuenKwan



Text

Description automatically generated

Diagram

Description automatically generated

Diagram

Description automatically generated

Diagram

Description automatically generated

Diagram

Description automatically generated

Diagram

Description automatically generated

### 2nd run: George wishes to get introduced to Bob



Text

Description automatically generated

Diagram

Description automatically generated

Diagram, schematic

Description automatically generated

Diagram

Description automatically generated

Diagram

Description automatically generated

Diagram

Description automatically generated

Diagram

Description automatically generated

### For Req. 6: Negative test case for relationship that cannot be established.



Graphical user interface, text

Description automatically generated

### For Req. 7: Negative test case for invalid names in arguments



A screenshot of a computer

Description automatically generated with medium confidence

### For Req. 8: Negative test case for names not found in the graph data



Graphical user interface, text, application

Description automatically generated

## Question 2

Text

Description automatically generated

### Greedy Search

Graphical user interface, text

Description automatically generated

Diagram

Description automatically generated

### A\* Search

Text

Description automatically generated

Diagram

Description automatically generated

### UCS

Text

Description automatically generated

Diagram

Description automatically generated

### Comparison between Greedy Search, A\* Search and UCS

|  |  |  |
| --- | --- | --- |
|  | Level Required | No. of Nodes Expanded |
| Greedy | 6 | 13 |
| A\* | 6 | 15 |
| UCS | 6 | 24 |

In terms of the level required, A\* Search and UCS are guaranteed for the optimal solution; Greedy does not, but in this scenario, it can also reach the optimal solution in the same performance as the other two algorithms.

For the no. of nodes expanded, because of the use of the heuristic function by Greedy Search and A\* Search, leading to way less nodes expanded than UCS. Yet, optimal solution is always guaranteed by UCS because it explores the nodes in all directions.

## Question 3

1. The tree search will reach the Goal State at depth 4.

Diagram, schematic

Description automatically generated

1. “The number of misplaced tiles” is NOT a good heuristic function for the 8-puzzle problem. A good heuristic function should fulfill both features: Admissible and Consistent.

First, “Admissibility” refers to the heuristic function should not overestimate the true cost to goal state. The heuristic function only considers the number of tiles that are not in their goal position, but not the distance of each tile from its goal position. This means that the estimate may be too low, making the algorithm inefficient, as it may choose a path that leads to a state with more tiles out of place.

Second, “Consistency” refers to the heuristic function should be non-decreasing and conform to Triangle Inequality, which is h(n) <= step cost + h(n’).

The middle node at depth 1 is h(n) = 3 and at depth 2 is h(n’) = 3.

A picture containing text, crossword puzzle

Description automatically generated

We have a step cost at 1 for every depth moved on. There fore, inconsistency occurs in the estimated cost at depth 2, where its step cost + h(n’) = 4 that is greater than h(n) at depth 1.

1. Modifying the heuristic to be “The number of misplaced tiles with total Manhattan Distance”.

h(n) = Sum up for all misplaced tiles (

Absolute value (tile’s current position in X axis – tile’s goal position in X axis) +

Absolute value (tile’s current position in Y axis – tile’s goal position in Y axis)

)

Referring to the Inconsistency in (b),

the Manhattan Distance at depth 1:

Tile “2”: |0 – 1| + |0 - 0| = 1

Tile “8”: |0 – 0| + |1 –0| = 1

Tile “1”: |0 – 0| + |1 – 0| = 1

h(n) = 1+1+1 = 3

the Manhattan Distance at depth 2:

Tile “2”: |0 – 1| + |0 - 0| = 1

Tile “8”: |0 – 0| + |1 –0| = 1

Tile “1”: |1 – 0| + |1 – 0| = 2

h(n) = 1+1+2 = 4

Therefore, the inconsistency is resolved.

For admissibility, the modified heuristic function only sums up the number of misplaced tiles and the sum of the Manhattan Distances of each of these tiles from their goal position. Since this calculation is done for each tile, it ensures that the true cost will not be overestimated.

As a result, it is a good heuristic function.