**Full Name (as shown in attendance sheet): KAR CHAUDHURI ANIRBAN**

**[Optional] LumiNUS Account:** e0508035@u.nus.edu

**[Optional] NRIC / Passport / NUS Matriculation No.: A0108517H**

(Select and provide the applicable)

**Institute of Systems Science**

**National University of Singapore**

**GRADUATE CERTIFICATE**

**INTELLIGENT REASONING SYSTEMS**

**Assessment**

**Subject: *Reasoning Systems***

SECTION A

|  |  |
| --- | --- |
| **Question** | **Marks** |
| **1** | **/1** |
| **TOTAL** | **/1** |

**Instructions for Paper**

Duration: Fifteen minutes exam

This is an *OPEN BOOK* examination. This examination paper consists of *one* Section and *one* Question. You are to answer *ALL* questions. There are a total of *1 Mark* for this paper.

1. Read **ALL** instructions before answering any of the examination questions.
2. Write your Student ID number on the **front page** of this examination paper in the box provided.
3. This is an **Open Book** examination. If you wish, you may use reference materials to answer a question. Reference materials can be *books, manuals, handouts* or *notes*.
4. Answers are to be written **only** in this **examination paper** and any **attachments** provided and will be considered for credit. Answers written in any appendices will **NOT** be marked.
5. Use a pen for writing your answers. Pencil may only be used for drawing diagrams and writing program code.
6. Non-programmable calculators may be used if required. **However, computers of any form (laptops, tablets, smart watches etc.) are not permitted to be brought into the examination hall.**
7. State clearly any assumptions you make in answering any question where you feel the requirement is not sufficiently clear.
8. At the end of the examination:
9. Hand-in the examination paper for **each** section **separately**, any appendices and attachments.
10. You are **not** allowed to remove the examination paper, appendices or attachments from the examination hall.

***REMEMBER:***

***This is an OPEN BOOK exam.***

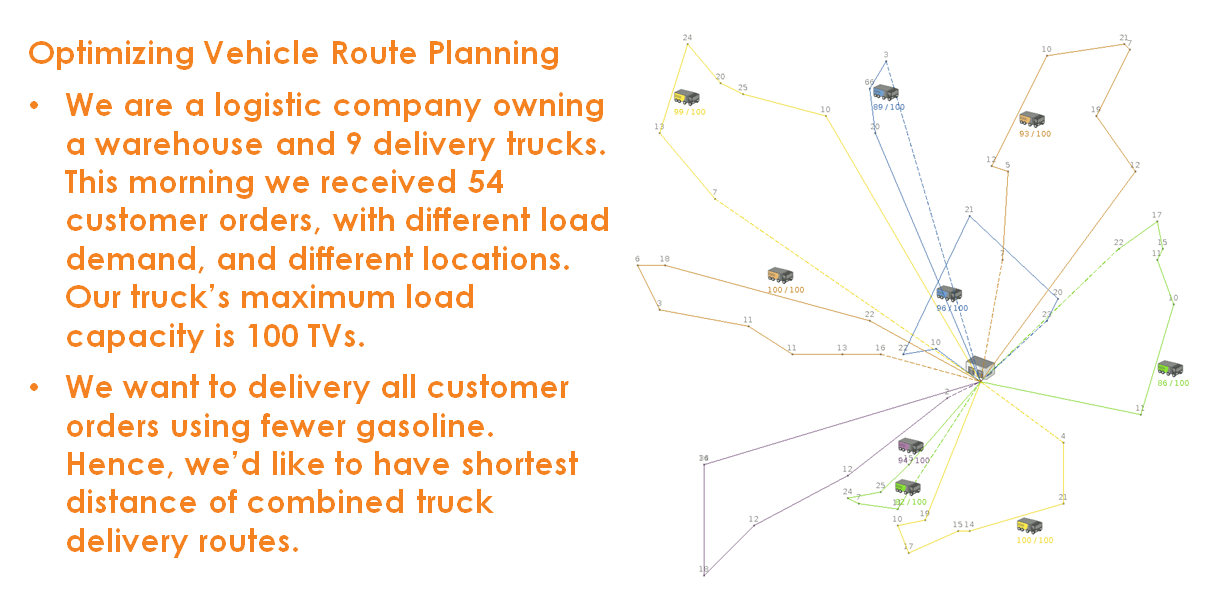
***There are a total of 1 Mark for this paper.***

***You are required to answer ALL questions.***

***State clearly any assumptions you make in answering any question where you feel the requirement is not sufficiently clear.***

**SECTION A**

**Question 1** *(Total: 1 Mark)*

****

References:

***VRP*** can be solved using ***State Space Search***, which is a process used in the field of computer science, including artificial intelligence (AI), in which successive configurations or states of an instance are considered, with the intention of finding a goal state with a desired property. Depth-first search and Breadth-first search are forms of state space search.

Problems are often modelled as a state space, a set of ***states*** that a problem can be in. The set of states forms a graph where two states are connected if there is an ***operation*** that can be performed to transform the first state into the second.

<https://en.wikipedia.org/wiki/State_space_search>

**Answer the following questions:**

* 1. Propose a *vector or graph or tree* representation of **state (VRP solution representation)** to carry out the **state space search**, and suggest an initial state. (Hint: There are 9 trucks, thus 9 delivery routes; for 54 unique customers/locations)

*(0.5 Mark)*

[Answer]

(Heuristics hill climbing search)

(truck capacity)

RL=[rl1, rl2, rl3, rl4, rl5, rl6, rl7, rl8, rl9]

=[99, 89, 93, 86, 100, 112, 94, 100, 96 ]

(initial distances of each corresponding trucks from warehouse)

[r1, r2, r3, r4, r5, r6, r7, r8, r9]

= [40, 40, 40, 42, 42, 42, 44, 44, 46, 46]

* 1. Define *one or more* possible **search operator/action** to permutate above designed state/solution of VRP.

*(0.5 Mark)*

[Answer]

The search operator can be defined by adding or subtracting a vector value representing distance an amount shown in the graph above depending on decision on next route.

RL=[rl1, rl2, rl3, rl4, rl5, rl6, rl7, rl8, rl9]

=[99, 89, 93, 86, 100, 112, 94, 100, 96 ]

(initial)

[r1, r2, r3, r4, r5, r6, r7, r8, r9]

= [40, 40, 40, 42, 42, 42, 44, 44, 46, 46]

1st iteration = [34, 36, 32, 30, 30,30, 28, 32, 30, 28 ]

2nd iteration = [20, 18, 18, 16, 15, 16, 15, 16, 12]

3rd iteration = [3, 2, 2, 2, 2, 2, 2, 2, 1]

4th iteration = [0, 0, 0, 0, 0, 0, 0, 0, 0]

**END OF ASSESSMENT PAPER**