

Assignment 1

As the two solution representations I will use for both task A and B are nearly identical, I will first detail a more general version of the multimodal truck and drone delivery system solution representation then specify any needed changes for each task.

The solution representation

I propose the solution representation of a permutable array, where the first nodes before a barrier node (here called **x**) or end of array is the mother/truck route. The first node in the array will be called the home node. After each subsequently barrier node we have a daughter/drone route. The first and last node in the daughter route must be equal to an node in the mother route. All routes are in arrival order.

To have a valid daughter route we need some integrity checks:

- The first and last node must be a node that the mother route contains (ie a number that specified before the first barrier)
- Length of child route must contain at least one ordinary node (ie length is equal to or greater than three)
- The last node must be after or the same as the initial daughter route node.

To disallow mother/daughter routes delivering to the same port multiple times do the following to verify:

- We do that by stripping the first and last nodes from each daughter route
- remove all barriers then
- check that each remaining nodes is a unique, valid node.

Task A

In task A we use the given solution representation with only one modification. The mother route does not contain the start node **s**. This means that no daughter route can start at **s**, as is required by this task.

Example

In all examples we have the nodes 1..8 and the starting node **s**.

Explanation of highlights:

- Where **bold x** is the barrier node
- Where *italic* numbers are daughter start/end nodes.
- Where no solution representationing means what nodes the route visits.

Valid (equal to the provided example)

6, 4, **x**, 6, 7, 8, 6, **x**, 6, 5, 6, **x**, 4, 3, 2, 1, 4

Invalid, node 3 visited twice

6, 4, **x**, 6, 3, 8, 6, **x**, 6, 5, 6, **x**, 4, 3, 2, 1, 4

Invalid, mother ship never visit node 5

6, 4, **x**, 5, 7, 8, 5, **x**, 6, 5, 6, **x**, 4, 3, 2, 1, 4

Invalid, daughter route contains no nodes

6, 4, **x**, 6, 7, 8, 6, **x**, 6, 5, 6, **x**, 4

Invalid, contains starting node

s, 6, 4, **x**, 6, 7, 8, 6, **x**, 6, 5, 6, **x**, 4, 3, 2, 1, 4

Task B

In this example we only need to strengthen the last daughter route requirement:

- Length of child route must contain exactly one ordinary node (ie length is always equal 3, ex)

Example

In all examples we have the nodes 0..12. With the barrier node equal **x**.

Explanation of highlights:

- Where **bold x** is the barrier node
- Where *italic* numbers are daughter start/end nodes.
- Where no solution representation means what nodes the route visits.

Valid (same example assignment 1-a provided example)

6, 4, **x**, 6, 7, 8, 6, **x**, 6, 5, 6, **x**, 4, 3, 2, 1, 4, 4

Valid (example example)

(node **x** is used as barrier, home node is the 0 node)

0, 10, 9, 8, 7, 3, 5, 6, 0, **x**, 0, 11, 10, **x**, 10, 12, 9, **x**, 7, 4, 3, **x**, 3, 2, 5, **x**, 3, 1, 5