Algorithm Specification

(Pseudo and Code)

Sample data needed to generate

* The route for delivery of the goods
* A list of grid co-ordinates, goods etc that would act as a backup if their IT failed

Specification for Algorithm Pseudo Code:

The algorithm will need to use dijkras solution for the shortest path.

Working Order of the algorithm:

1. Give each van a delivery ID (each delivery ID is for each van path)
2. Find the shortest path using dijkras algorithim.
3. Store each node of the path. (Use array or which method you think is the best of storing this. Please let me know what you used and why)

-After shortest path is decided.

(**See output example down below)**

1. Output delivery ID
2. Output number of deliveries
3. 5.Output each co-ordinate of each node of the path. (delivery path)
4. Output goods that were on the van
5. Store these in a document (notepad or whatever is easiest)
6. Output number of deliveries
7. Output each co-ordinate of each node of the path. (delivery path)
8. Output goods that where on the van.
9. Store these in a document (notepad or whatever is easiest)

Paths to program:

Each square is a node for the algorithm.

Green = Path 1

Orange = Path 2

Red = HQ

Chart, scatter chart

Description automatically generated

Goods on Van:

1. Protein Meal
2. Weight Loss
3. Health Meal

**Output example:**

**Delivery Green nodes**

Delivery ID 1

Number of Deliveries: 13

**\*Delivery number 1 co-ordinates is an example\***

|  |  |  |
| --- | --- | --- |
|  | Delivery Co-Ordinates | |
| Delivery number(node) | X | Y |
| 1 | 0 | -5 |
| 2 |  |  |
| 3 |  |  |
| 4 |  |  |

**\*This is an example this would be 13 deliveries.**

**This also would be different for orange nodes\***

Boxes on van: 39 ( 1 Box per delivery)

Each box is 2 meals inside.

|  |  |  |  |
| --- | --- | --- | --- |
| Goods on Van | Delivery ID | Boxes on Van | Product QTY |
| Protein Meals | 1 | 13 | 26 |
| Weight Loss Meals | 1 | 13 | 26 |
| Health Meal | 1 | 13 | 26 |
|  | Total | 39 | 78 |