



Coding Challenge

Drone Delivery Service

2023

Summary	3
Drone Delivery Service	3
Solution Provided	3
Given Input	3
Expected Output	4
Algorithm	4
Approach	4
Best Practices	4
Technical Dependencies and Libraries	4
Example Input and Output Files	5

Summary

During the interview process, Veloziert clients like to see a coding example so they can understand software development styles and skills. This coding challenge will provide that example and should take approximately 2-4 hours for you to complete.

Please read the instructions carefully and make sure you submit your written strategy in addition to the coded solution. Be sure to provide the input file as well as ensure your solution runs and produces output!

If you have questions about the assignment, please ask; don't assume. We are happy to clarify anything that is unclear.

Drone Delivery Service

A squad of drones is tasked with delivering packages for a major online retailer in a world where time and distance do not matter.

Each drone can carry a specific weight and can make multiple deliveries before returning to home base to pick up additional packages; however, the goal is to make the fewest number of trips, as each time the drone returns to home base, it is extremely costly to refuel and reload the drone.

The software shall accept input which will include: the name of each drone, the maximum weight it can carry, along with a series of locations and the total weight needed to be delivered to that specific location. The software should highlight the most efficient deliveries for each drone to make on each trip.

Assume that time and distance to each location do not matter, and that the size of each package is irrelevant. It is also assumed that the cost to refuel and restock each drone is a constant and does not vary between drones.

The maximum number of drones in a squad is 100, and there is no maximum number of required deliveries.

Solution Provided

Please supply an input data file. The client should be able to run the project and display results.

Given Input

Line 1: [Drone #1 Name], [#1 Maximum Weight], [Drone #2 Name], [#2 Maximum Weight], etc.

Line 2: [Location #1 Name], [Location #1 Package Weight]

Line 3: [Location #2 Name], [Location #2 Package Weight] Line 4:

[Location #3 Name], [Location #3 Package Weight] Etc.

Expected Output

[Drone #1 Name]

Trip #1

[Location #2 Name], [Location #3 Name]

Trip #2

[Location #1 Name]

[Drone #2 Name]

Trip #1

[Location #4 Name], [Location #7 Name]

Trip #2

[Location #5 Name], [Location #6 Name]

Algorithm

Please provide a written summary describing any Algorithm you chose to utilize in your Solution.

Approach

Please describe in a few sentences the approach you took in developing your solution and any assumptions you made.

Best Practices

Be sure that your solution is well documented and follows Coding Best Practices. Be sure that your code is readable and easy to follow, and that it optimizes space and time.

1. Write as few lines as possible of legible code.
2. Segment blocks of code in the same section.
3. Use indentation to mark the beginning and end of control structures, methods, and/or functions used and developed throughout the project.
4. Write portable code. That way it will work on any environment.

Technical Dependencies and Libraries

Please add or explain any dependencies and/or libraries that you are using in your project solution. Also, do not forget to add the framework versions and applications you are using to write, compile, and execute your code (i.e., Visual Code or Microsoft Visual Studio).

Example Input and Output Files

These input and output files are shown to provide you with an example of what a given input file should produce in terms of output. They are here to show you examples to help you better understand the assignment. And you should run the input files and verify you get the outputs provided here to validate your solution.

Please provide your own, unique, input file and output files with your solution.

INPUT/OUTPUT example #1:

BEGIN INPUT FILE #1

[DroneA], [200], [DroneB], [250], [DroneC], [100]
[LocationA], [200] [LocationB],
[150]
[LocationC], [50]
[LocationD], [150]
[LocationE], [100] [LocationF],
[200]
[LocationG], [50] [LocationH],
[80]
[LocationI], [70]
[LocationJ], [50]
[LocationK], [30]
[LocationL], [20]
[LocationM], [50]
[LocationN], [30] [LocationO],
[20]
[LocationP], [90]

END INPUT FILE #1

BEGIN OUTPUT FILE #1

[DroneA]
Trip #1
[LocationI], [LocationJ], [LocationM], [LocationN]

[DroneB]
Trip #1
[LocationA]
Trip #2
[LocationB], [LocationC] Trip
#3

[LocationD], [LocationE]

Trip #4

[LocationF], [LocationG]

Trip #5

[LocationH], [LocationK], [LocationL], [LocationO], [LocationP]

[DroneC]

END OUTPUT FILE #1

BEGIN INPUT FILE #1

[DroneA], [300], [DroneB], [350], [DroneC], [200]

[LocationA], [200]

[LocationB], [150]

[LocationC], [50]

[LocationD], [150]

[LocationE], [100] [LocationF],
[200]

[LocationG], [50] [LocationH],
[80]

[LocationI], [70]

[LocationJ], [50]

END INPUT FILE #2

BEGIN OUTPUT FILE #2

[DroneA]

Trip #1

[LocationF], [LocationI]

[DroneB]

Trip #1

[LocationA], [LocationD]

Trip #2

[LocationE], [LocationG], [LocationH], [LocationJ]

[DroneC]

Trip #1

[LocationB], [LocationC]

END OUTPUT FILE #2