C950 Task-2 WGUPS Write-Up

(Task-2: The implementation phase of the WGUPS Routing Program).

(Zip your source code and upload it with this file)

Brandon Nguyen

ID #003964281

WGU Email: bnguy80@wgu.edu

Date: 8/11/23

C950 Data Structures and Algorithms II

**Stated Problem:**

The objective of this program is to establish efficient routes and distribution plans for package deliveries within the Western Governor University Parcel Service (WGUPS), utilizing Python 3.11. The task involves handling 40 packages scheduled for delivery by the end of the day, each with unique delivery deadlines. Challenges include limit of three trucks and two available drivers for delivery, accounting for delays in package arrival, adhering to specific truck loading constraints, and accommodating special requirements like grouped package deliveries and designated trucks for certain packages. To address these challenges a systematic approach is proposed.

**Proposed Solution and Approach:**

Our approach consists of several strategic steps aimed at optimizing package deliveries for the Western Governor University Parcel Service (WGUPS):

1. **Truck Prioritization:** The three available trucks will be prioritized based on package delivery deadlines. High priority(truck1) for packages with a delivery deadline of 9:00 AM, medium priority(truck2) with delivery deadline of 10:30 AM, and low priority(truck3) for packages with deliver deadline by EOD which will be defined as 5: 00 PM. This prioritization will for the foundation for efficient route planning and distribution.

2**. Nearest Neighbor Greedy Algorithm:** We will incorporate the nearest neighbor greedy algorithm to sort the order of packages loaded on trucks, this step aims to reduce the distance covered when calling the function that implements dijkstra's shortest path greedy algorithm.

3. **Dijkstra's Shortest Path Greedy Algorithm:** The core of our approach lies in the implementation of the dijkstra's shortest path algorithm. This algorithm calculates the most optimal route to for each truck. It ensures the the chosen paths are the shortest possible, contributing to efficient and timely deliveries.

4. **Two-Opt Greedy Algorithm:** Building upon the results of the dijkstra's algorithm, we introduce the two-opt greedy algorithm. This final step aims to further enhance route efficiency by iteratively swapping pairs of edges in the optimized routes obtained from the Dijkstra's algorithm. This process continues until no further improvements can be made, resulting in a route with minimized total distance traveled.

A. Hash Table

Hash Table Screenshot goes here

B. Look-Up Functions

Look-up function screenshot goes here

C. Original Code

Major code blocks screenshots go here showing implementation

C1. Identification Information

main.py screenshot goes here showing Student ID

C2. Process and Flow Comments

Some code blocks screenshots go here showing comments

D. Interface

Interface screenshot goes here

D1. First Status Check

Screen shot goes here

D2. Second Status Check

Screenshot goes here

D3. Third Status Check

Screenshot goes here

E. Screenshot of Code Execution

Screenshot goes here

F1. Strengths of the Chosen Algorithm

Text goes here

F2. Verification of Algorithm

Text goes here

F3. Other Possible Algorithms

Text goes here

F3a. Algorithm Differences

Text goes here

G. Different Approach

Text goes here

H. Verification of Data Structure

Text goes here

H1. Other Data Structures

Text goes here

H1a. Data Structure Differences

Text goes here

I. Sources

Text goes here

An example:

Lysecky, R., & Vahid, F. (2018, June). *C950: Data Structures and Algorithms II*. zyBooks.

Retrieved March 22, 2021, from <https://learn.zybooks.com/zybook/WGUC950AY20182019/>

J. Professional Communication