C950 Task-2 WGUPS Write-Up

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

Alireza Barzegar

ID #010057350

WGU Email: abarzeg@wgu.edu

06/05/2024

C950 Data Structures and Algorithms II

A. Hash Table

HashTable.py > HashTable > search

```python
class HashTable:

    def __init__(self, size=7):
        self.size = size
        self.table = [None] * self.size

    def __hash(self, key):
        return key % len(self.table)

    def insert(self, key, item):
        index = self.__hash(key)
        if self.table[index] == None:
            self.table[index] = []
        self.table[index].append([key, item])

    def search(self, key):
        toReturn = None
        index = self.__hash(key)
        if self.table[index] != None:
            pairList = self.table[index]
            for pair in pairList:
                if pair[0] == key:
                    toReturn = pair[1]
                    break
        return toReturn

    def remove(self, key):
        toReturn = False
        index = self.__hash(key)
        if self.table[index] != None:
            for pair in self.table[index]:
                if pair[0] == key:
                    toReturn = True
                    self.table[index].remove(pair)
                    break
            return toReturn
```

B. Look-Up Functions

```python
# lookup function that takes in a package id and returns a list containing each component of that specific package
def lookup(key):
    package = packages.search(key)
    components = [package.address, package.deadline, package.city, package.zipcode, package.weight, package.status]
    return components
```

```python
# findPackage function that finds the package in hashtable and returns the objec
def findPackage(hashTable, key):
    return hashTable.search(key)
```

C. Original Code

Major code blocks screenshots go here showing implementation

C1. Identification Information

main.py × distanceData.csv Package.py

main.py > ...

```python
# student ID: 010057350
import datetime
from re import M
from Truck import Truck
from Package import Package
from LoadDistanceData import loadDistanceData, loadAddressData
from LoadPackageData import loadPackageData
from HashTable import HashTable
from statusEnum import Status

packages = HashTable(20)
distanceData = []
addressData = []

loadDistanceData(distanceData)
loadAddressData(addressData)
loadPackageData(packages)

# function to return distance between 2 addresses
def distanceBetween(address1, address2):
    toReturn = distanceData[addressData.index(address1)][addressData.index(address2)]
    if toReturn == "":
        toReturn = distanceData[addressData.index(address2)][addressData.index(address1)]
    return toReturn

# function using greedy algorithm for delivering packages based on the next closest address
def deliverPackages(truck):
    for i in range(len(truck.packages)):
        minMiles = ''
        nextPackage = truck.packages[i]
        for j in range(len(truck.packages)):
            if truck.packages[j].status != Status.delivered:
                curr = distanceBetween(truck.current_location, truck.packages[j].address)
                curr = float(curr)
                if "wrong address" in truck.packages[j].specialNotes.lower() and j < len(truck.packages) - 1:
                    if truck.current_time < datetime.time(10, 20):
                        continue
                    else:
                        truck.packages[j].address = "410 South State St"
                        truck.packages[j].zipcode = "84111"
                if minMiles == '':
                    minMiles = curr
                    nextPackage = truck.packages[j]
                elif curr < minMiles:
                    minMiles = curr
                    nextPackage = truck.packages[j]
        truck.total_miles += round(float(minMiles), 2)
        truck.current_location = nextPackage.address
        mins = round((float(minMiles) / (18 / 60)))
        currentTimeMins = (truck.current_time.hour * 60) + truck.current_time.minute + mins
        currentTimeHours = currentTimeMins // 60
        currentTimeMins %= 60
        truck.current_time = datetime.time(currentTimeHours, currentTimeMins)
        nextPackage.deliveryTime = truck.current_time
        nextPackage.status = Status.delivered
    truck.current_location = "HUB"
```

C2. Process and Flow Comments

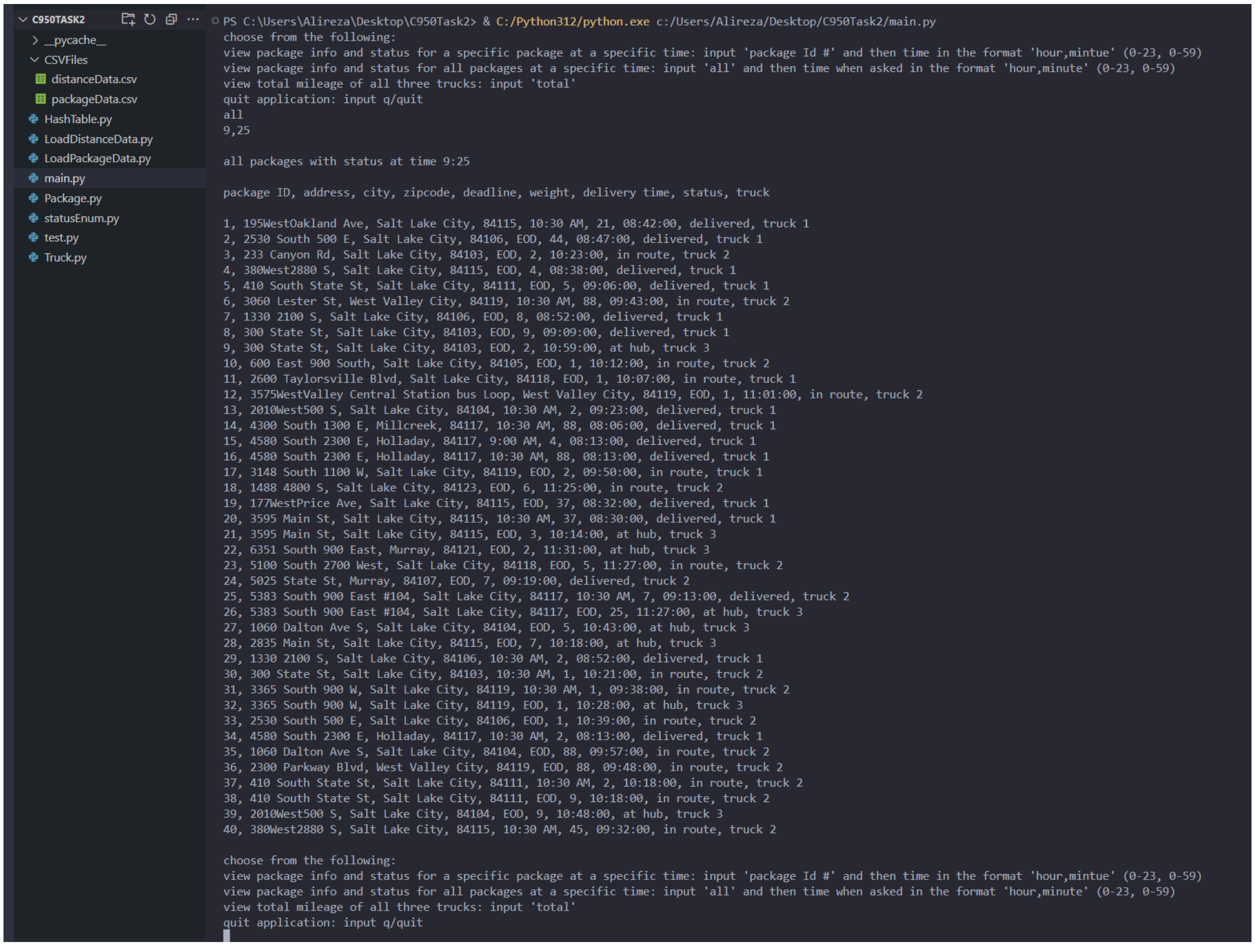
```python

# function for loading the truck
def loadTruck(truck, packages, time):
    truck.packages = packages
    truck.current_time = time
    truck.load_time = time
    for package in truck.packages:
        package.status = Status.inRoute
        package.loadingTime = truck.load_time
# findPackage function that finds the package in hashtable and returns the object
def findPackage(hashTable, key):
    return hashTable.search(key)

# lookup function that takes in a package id and returns a list containing each component of that specific package
def lookup(key):
    package = packages.search(key)
    components = [package.address, package.deadline, package.city, package.zipcode, package.weight, package.status]
    return components

# prints choices for user input in commandline interface when called.
def printOptions():
    print("choose from the following: ")
    print("view package info and status for a specific package at a specific time: input 'package Id #' and then time in the format 'hour,mintue' (0-23, 0-59)")
    print("view package info and status for all packages at a specific time: input 'all' and then time when asked in the format 'hour,minute' (0-23, 0-59)")
    print("view total mileage of all three trucks: input 'total'")
    print("quit application: input q/quit")
```

```python
# initialize the 3 trucks
truck1 = Truck("truck 1")
truck2 = Truck("truck 2")
truck3 = Truck("truck 3")

# load the truck1 and truck 2 with their packages, truck3 waits until one of the 2 drivers is back
loadTruck(truck1, truck1Packages, datetime.time(8,0))
loadTruck(truck2, truck2Packages, datetime.time(8,0))

# deliver truck1 and truck2 packages. have the driver that returns first load and deliver truck3 packages.
for truck in [truck1, truck2]:
    deliverPackages(truck)

if truck1.current_time < truck2.current_time:
    loadTruck(truck3, truck3Packages, truck1.current_time)
else:
    loadTruck(truck3, truck3Packages, truck2.current_time)

deliverPackages(truck3)

```

D. Interface

```
PROBLEMS   OUTPUT   DEBUG CONSOLE   TERMINAL   PORTS

PS C:\Users\Alireza\Documents\C950Task2> & C:/Python312/python.exe c:/Users/Alireza/Documents/C950Task2/main.py
choose from the following:
view package info and status for a specific package at a specific time: input 'package Id #' and then time in the format 'hour,mintue' (0-23, 0-59)
view package info and status for all packages at a specific time: input 'all' and then time when asked in the format 'hour,minute' (0-23, 0-59)
view total mileage of all three trucks: input 'total'
quit application: input q/quit
```

D1. First Status Check



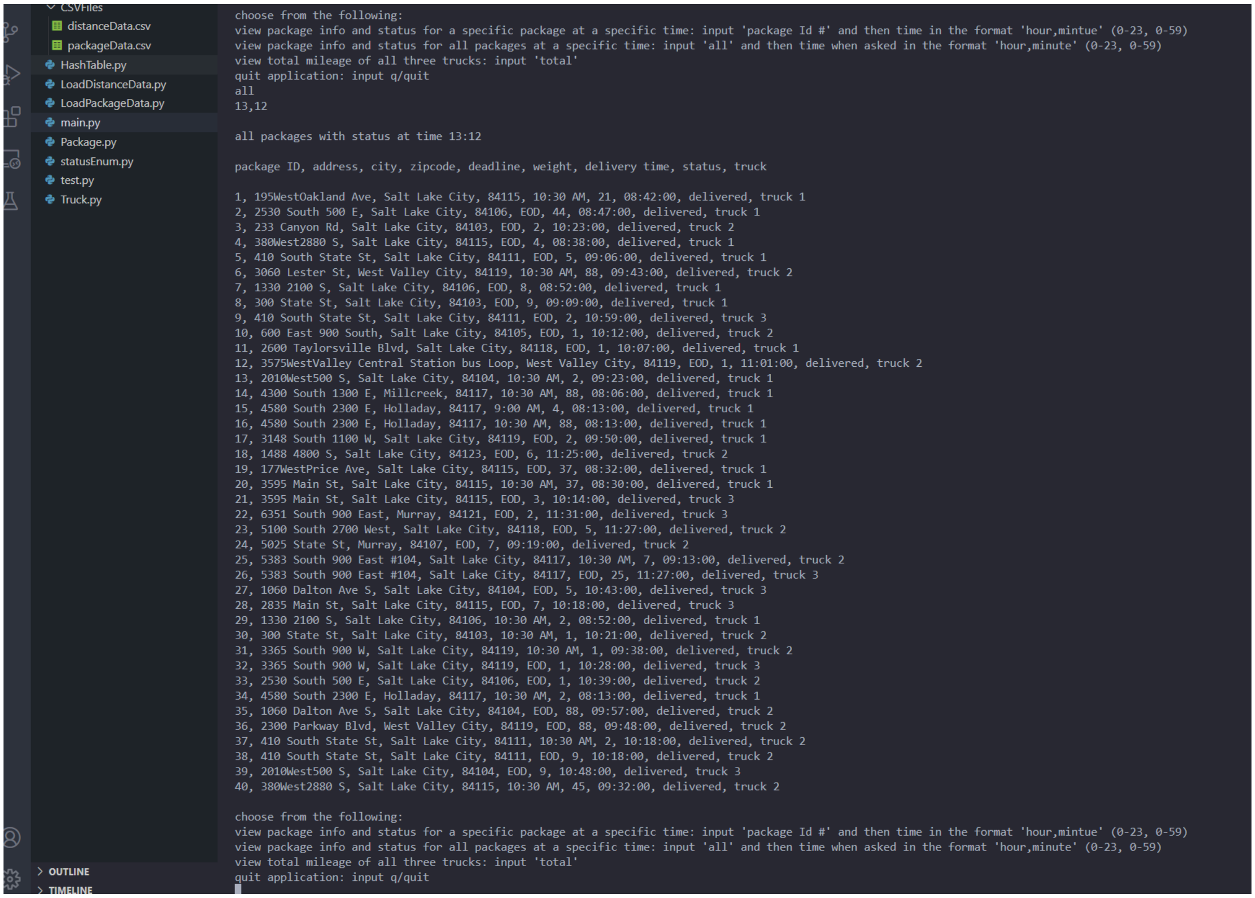

```
PS C:\Users\Alireza\Desktop\C950Task2> & C:/Python312/python.exe c:/Users/Alireza/Desktop/C950Task2/main.py
choose from the following:
view package info and status for a specific package at a specific time: input 'package Id #' and then time in the format 'hour,mintue' (0-23, 0-59)
view package info and status for all packages at a specific time: input 'all' and then time when asked in the format 'hour,minute' (0-23, 0-59)
view total mileage of all three trucks: input 'total'
quit application: input q/quit
all
9,25

all packages with status at time 9:25

package ID, address, city, zipcode, deadline, weight, delivery time, status, truck

1, 195WestOakland Ave, Salt Lake City, 84115, 10:30 AM, 21, 08:42:00, delivered, truck 1
2, 2530 South 500 E, Salt Lake City, 84106, EOD, 44, 08:47:00, delivered, truck 1
3, 233 Canyon Rd, Salt Lake City, 84103, EOD, 2, 10:23:00, in route, truck 2
4, 380West2880 S, Salt Lake City, 84115, EOD, 4, 08:38:00, delivered, truck 1
5, 410 South State St, Salt Lake City, 84111, EOD, 5, 09:06:00, delivered, truck 1
6, 3060 Lester St, West Valley City, 84119, 10:30 AM, 88, 09:43:00, in route, truck 2
7, 1330 2100 S, Salt Lake City, 84106, EOD, 8, 08:52:00, delivered, truck 1
8, 300 State St, Salt Lake City, 84103, EOD, 9, 09:09:00, delivered, truck 1
9, 300 State St, Salt Lake City, 84103, EOD, 2, 10:59:00, at hub, truck 3
10, 600 East 900 South, Salt Lake City, 84105, EOD, 1, 10:12:00, in route, truck 2
11, 2600 Taylorsville Blvd, Salt Lake City, 84118, EOD, 1, 10:07:00, in route, truck 1
12, 3575WestValley Central Station bus Loop, West Valley City, 84119, EOD, 1, 11:01:00, in route, truck 2
13, 2010West500 S, Salt Lake City, 84104, 10:30 AM, 2, 09:23:00, delivered, truck 1
14, 4300 South 1300 E, Millcreek, 84117, 10:30 AM, 88, 08:06:00, delivered, truck 1
15, 4580 South 2300 E, Holladay, 84117, 9:00 AM, 4, 08:13:00, delivered, truck 1
16, 4580 South 2300 E, Holladay, 84117, 10:30 AM, 88, 08:13:00, delivered, truck 1
17, 3148 South 1100 W, Salt Lake City, 84119, EOD, 2, 09:50:00, in route, truck 1
18, 1488 4800 S, Salt Lake City, 84123, EOD, 6, 11:25:00, in route, truck 2
19, 177WestPrice Ave, Salt Lake City, 84115, EOD, 37, 08:32:00, delivered, truck 1
20, 3595 Main St, Salt Lake City, 84115, 10:30 AM, 37, 08:30:00, delivered, truck 1
21, 3595 Main St, Salt Lake City, 84115, EOD, 3, 10:14:00, at hub, truck 3
22, 6351 South 900 East, Murray, 84121, EOD, 2, 11:31:00, at hub, truck 3
23, 5100 South 2700 West, Salt Lake City, 84118, EOD, 5, 11:27:00, in route, truck 2
24, 5025 State St, Murray, 84107, EOD, 7, 09:19:00, delivered, truck 2
25, 5383 South 900 East #104, Salt Lake City, 84117, 10:30 AM, 7, 09:13:00, delivered, truck 2
26, 5383 South 900 East #104, Salt Lake City, 84117, EOD, 25, 11:27:00, at hub, truck 3
27, 1060 Dalton Ave S, Salt Lake City, 84104, EOD, 5, 10:43:00, at hub, truck 3
28, 2835 Main St, Salt Lake City, 84115, EOD, 7, 10:18:00, at hub, truck 3
29, 1330 2100 S, Salt Lake City, 84106, 10:30 AM, 2, 08:52:00, delivered, truck 1
30, 300 State St, Salt Lake City, 84103, 10:30 AM, 1, 10:21:00, in route, truck 2
31, 3365 South 900 W, Salt Lake City, 84119, 10:30 AM, 1, 09:38:00, in route, truck 2
32, 3365 South 900 W, Salt Lake City, 84119, EOD, 1, 10:28:00, at hub, truck 3
33, 2530 South 500 E, Salt Lake City, 84106, EOD, 1, 10:39:00, in route, truck 2
34, 4580 South 2300 E, Holladay, 84117, 10:30 AM, 2, 08:13:00, delivered, truck 1
35, 1060 Dalton Ave S, Salt Lake City, 84104, EOD, 88, 09:57:00, in route, truck 2
36, 2300 Parkway Blvd, West Valley City, 84119, EOD, 88, 09:48:00, in route, truck 2
37, 410 South State St, Salt Lake City, 84111, 10:30 AM, 2, 10:18:00, in route, truck 2
38, 410 South State St, Salt Lake City, 84111, EOD, 9, 10:18:00, in route, truck 2
39, 2010West500 S, Salt Lake City, 84104, EOD, 9, 10:48:00, at hub, truck 3
40, 380West2880 S, Salt Lake City, 84115, 10:30 AM, 45, 09:32:00, in route, truck 2

choose from the following:
view package info and status for a specific package at a specific time: input 'package Id #' and then time in the format 'hour,mintue' (0-23, 0-59)
view package info and status for all packages at a specific time: input 'all' and then time when asked in the format 'hour,minute' (0-23, 0-59)
view total mileage of all three trucks: input 'total'
quit application: input q/quit
```

D2. Second Status Check

```
CSVFiles
  distanceData.csv
  packageData.csv
HashTable.py
LoadDistanceData.py
LoadPackageData.py
main.py
Package.py
statusEnum.py
test.py
Truck.py
OUTLINE
TIMELINE
```

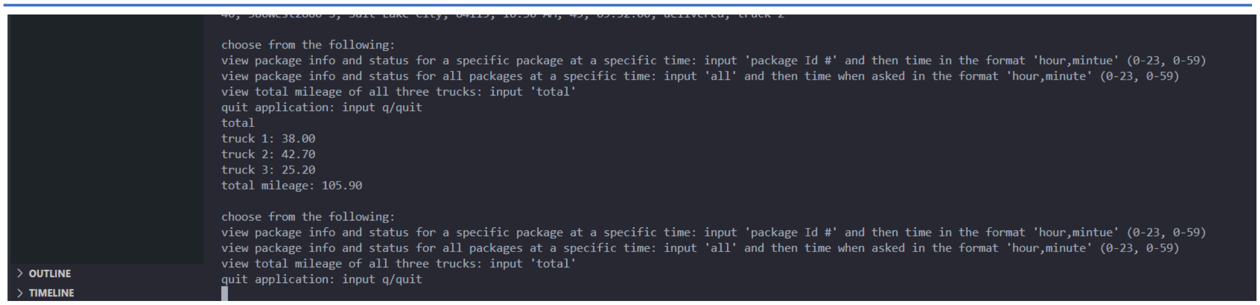
```
choose from the following:
view package info and status for a specific package at a specific time: input 'package Id #' and then time in the format 'hour,mintue' (0-23, 0-59)
view package info and status for all packages at a specific time: input 'all' and then time when asked in the format 'hour,minute' (0-23, 0-59)
view total mileage of all three trucks: input 'total'
quit application: input q/quit
all
10,35

all packages with status at time 10:35

package ID, address, city, zipcode, deadline, weight, delivery time, status, truck

1, 195WestOakland Ave, Salt Lake City, 84115, 10:30 AM, 21, 08:42:00, delivered, truck 1
2, 2530 South 500 E, Salt Lake City, 84106, EOD, 44, 08:47:00, delivered, truck 1
3, 233 Canyon Rd, Salt Lake City, 84103, EOD, 2, 10:23:00, delivered, truck 2
4, 380West2880 S, Salt Lake City, 84115, EOD, 4, 08:38:00, delivered, truck 1
5, 410 South State St, Salt Lake City, 84111, EOD, 5, 09:06:00, delivered, truck 1
6, 3060 Lester St, West Valley City, 84119, 10:30 AM, 88, 09:43:00, delivered, truck 2
7, 1330 2100 S, Salt Lake City, 84106, EOD, 8, 08:52:00, delivered, truck 1
8, 300 State St, Salt Lake City, 84103, EOD, 9, 09:09:00, delivered, truck 1
9, 410 South State St, Salt Lake City, 84111, EOD, 2, 10:59:00, in route, truck 3
10, 600 East 900 South, Salt Lake City, 84105, EOD, 1, 10:12:00, delivered, truck 2
11, 2600 Taylorsville Blvd, Salt Lake City, 84118, EOD, 1, 10:07:00, delivered, truck 1
12, 3575WestValley Central Station bus Loop, West Valley City, 84119, EOD, 1, 11:01:00, in route, truck 2
13, 2010West500 S, Salt Lake City, 84104, 10:30 AM, 2, 09:23:00, delivered, truck 1
14, 4300 South 1300 E, Millcreek, 84117, 10:30 AM, 88, 08:06:00, delivered, truck 1
15, 4580 South 2300 E, Holladay, 84117, 9:00 AM, 4, 08:13:00, delivered, truck 1
16, 4580 South 2300 E, Holladay, 84117, 10:30 AM, 88, 08:13:00, delivered, truck 1
17, 3148 South 1100 W, Salt Lake City, 84119, EOD, 2, 09:50:00, delivered, truck 1
18, 1488 4800 S, Salt Lake City, 84123, EOD, 6, 11:25:00, in route, truck 2
19, 177WestPrice Ave, Salt Lake City, 84115, EOD, 37, 08:32:00, delivered, truck 1
20, 3595 Main St, Salt Lake City, 84115, 10:30 AM, 37, 08:30:00, delivered, truck 1
21, 3595 Main St, Salt Lake City, 84115, EOD, 3, 10:14:00, delivered, truck 3
22, 6351 South 900 East, Murray, 84121, EOD, 2, 11:31:00, in route, truck 3
23, 5100 South 2700 West, Salt Lake City, 84118, EOD, 5, 11:27:00, in route, truck 2
24, 5025 State St, Murray, 84107, EOD, 7, 09:19:00, delivered, truck 2
25, 5383 South 900 East #104, Salt Lake City, 84117, 10:30 AM, 7, 09:13:00, delivered, truck 2
26, 5383 South 900 East #104, Salt Lake City, 84117, EOD, 25, 11:27:00, in route, truck 3
27, 1060 Dalton Ave S, Salt Lake City, 84104, EOD, 5, 10:43:00, in route, truck 3
28, 2835 Main St, Salt Lake City, 84115, EOD, 7, 10:18:00, delivered, truck 3
29, 1330 2100 S, Salt Lake City, 84106, 10:30 AM, 2, 08:52:00, delivered, truck 1
30, 300 State St, Salt Lake City, 84103, 10:30 AM, 1, 10:21:00, delivered, truck 2
31, 3365 South 900 W, Salt Lake City, 84119, 10:30 AM, 1, 09:38:00, delivered, truck 2
32, 3365 South 900 W, Salt Lake City, 84119, EOD, 1, 10:28:00, delivered, truck 3
33, 2530 South 500 E, Salt Lake City, 84106, EOD, 1, 10:39:00, in route, truck 2
34, 4580 South 2300 E, Holladay, 84117, 10:30 AM, 2, 08:13:00, delivered, truck 1
35, 1060 Dalton Ave S, Salt Lake City, 84104, EOD, 88, 09:57:00, delivered, truck 2
36, 2300 Parkway Blvd, West Valley City, 84119, EOD, 88, 09:48:00, delivered, truck 2
37, 410 South State St, Salt Lake City, 84111, 10:30 AM, 2, 10:18:00, delivered, truck 2
38, 410 South State St, Salt Lake City, 84111, EOD, 9, 10:18:00, delivered, truck 2
39, 2010West500 S, Salt Lake City, 84104, EOD, 9, 10:48:00, in route, truck 3
40, 380West2880 S, Salt Lake City, 84115, 10:30 AM, 45, 09:32:00, delivered, truck 2

choose from the following:
view package info and status for a specific package at a specific time: input 'package Id #' and then time in the format 'hour,mintue' (0-23, 0-59)
view package info and status for all packages at a specific time: input 'all' and then time when asked in the format 'hour,minute' (0-23, 0-59)
view total mileage of all three trucks: input 'total'
quit application: input q/quit
```

D3. Third Status Check




```
choose from the following:
view package info and status for a specific package at a specific time: input 'package Id #' and then time in the format 'hour,mintue' (0-23, 0-59)
view package info and status for all packages at a specific time: input 'all' and then time when asked in the format 'hour,minute' (0-23, 0-59)
view total mileage of all three trucks: input 'total'
quit application: input q/quit
all
13,12

all packages with status at time 13:12

package ID, address, city, zipcode, deadline, weight, delivery time, status, truck

1, 195WestOakland Ave, Salt Lake City, 84115, 10:30 AM, 21, 08:42:00, delivered, truck 1
2, 2530 South 500 E, Salt Lake City, 84106, EOD, 44, 08:47:00, delivered, truck 1
3, 233 Canyon Rd, Salt Lake City, 84103, EOD, 2, 10:23:00, delivered, truck 2
4, 380West2880 S, Salt Lake City, 84115, EOD, 4, 08:38:00, delivered, truck 1
5, 410 South State St, Salt Lake City, 84111, EOD, 5, 09:06:00, delivered, truck 1
6, 3060 Lester St, West Valley City, 84119, 10:30 AM, 88, 09:43:00, delivered, truck 2
7, 1330 2100 S, Salt Lake City, 84106, EOD, 8, 08:52:00, delivered, truck 1
8, 300 State St, Salt Lake City, 84103, EOD, 9, 09:09:00, delivered, truck 1
9, 410 South State St, Salt Lake City, 84111, EOD, 2, 10:59:00, delivered, truck 3
10, 600 East 900 South, Salt Lake City, 84105, EOD, 1, 10:12:00, delivered, truck 2
11, 2600 Taylorsville Blvd, Salt Lake City, 84118, EOD, 1, 10:07:00, delivered, truck 1
12, 3575WestValley Central Station bus Loop, West Valley City, 84119, EOD, 1, 11:01:00, delivered, truck 2
13, 2010West500 S, Salt Lake City, 84104, 10:30 AM, 2, 09:23:00, delivered, truck 1
14, 4300 South 1300 E, Millcreek, 84117, 10:30 AM, 88, 08:06:00, delivered, truck 1
15, 4580 South 2300 E, Holladay, 84117, 9:00 AM, 4, 08:13:00, delivered, truck 1
16, 4580 South 2300 E, Holladay, 84117, 10:30 AM, 88, 08:13:00, delivered, truck 1
17, 3148 South 1100 W, Salt Lake City, 84119, EOD, 2, 09:50:00, delivered, truck 1
18, 1488 4800 S, Salt Lake City, 84123, EOD, 6, 11:25:00, delivered, truck 2
19, 177WestPrice Ave, Salt Lake City, 84115, EOD, 37, 08:32:00, delivered, truck 1
20, 3595 Main St, Salt Lake City, 84115, 10:30 AM, 37, 08:30:00, delivered, truck 1
21, 3595 Main St, Salt Lake City, 84115, EOD, 3, 10:14:00, delivered, truck 3
22, 6351 South 900 East, Murray, 84121, EOD, 2, 11:31:00, delivered, truck 3
23, 5100 South 2700 West, Salt Lake City, 84118, EOD, 5, 11:27:00, delivered, truck 2
24, 5025 State St, Murray, 84107, EOD, 7, 09:19:00, delivered, truck 2
25, 5383 South 900 East #104, Salt Lake City, 84117, 10:30 AM, 7, 09:13:00, delivered, truck 2
26, 5383 South 900 East #104, Salt Lake City, 84117, EOD, 25, 11:27:00, delivered, truck 3
27, 1060 Dalton Ave S, Salt Lake City, 84104, EOD, 5, 10:43:00, delivered, truck 3
28, 2835 Main St, Salt Lake City, 84115, EOD, 7, 10:18:00, delivered, truck 3
29, 1330 2100 S, Salt Lake City, 84106, 10:30 AM, 2, 08:52:00, delivered, truck 1
30, 300 State St, Salt Lake City, 84103, 10:30 AM, 1, 10:21:00, delivered, truck 2
31, 3365 South 900 W, Salt Lake City, 84119, 10:30 AM, 1, 09:38:00, delivered, truck 2
32, 3365 South 900 W, Salt Lake City, 84119, EOD, 1, 10:28:00, delivered, truck 3
33, 2530 South 500 E, Salt Lake City, 84106, EOD, 1, 10:39:00, delivered, truck 2
34, 4580 South 2300 E, Holladay, 84117, 10:30 AM, 2, 08:13:00, delivered, truck 1
35, 1060 Dalton Ave S, Salt Lake City, 84104, EOD, 88, 09:57:00, delivered, truck 2
36, 2300 Parkway Blvd, West Valley City, 84119, EOD, 88, 09:48:00, delivered, truck 2
37, 410 South State St, Salt Lake City, 84111, 10:30 AM, 2, 10:18:00, delivered, truck 2
38, 410 South State St, Salt Lake City, 84111, EOD, 9, 10:18:00, delivered, truck 2
39, 2010West500 S, Salt Lake City, 84104, EOD, 9, 10:48:00, delivered, truck 3
40, 380West2880 S, Salt Lake City, 84115, 10:30 AM, 45, 09:32:00, delivered, truck 2

choose from the following:
view package info and status for a specific package at a specific time: input 'package Id #' and then time in the format 'hour,mintue' (0-23, 0-59)
view package info and status for all packages at a specific time: input 'all' and then time when asked in the format 'hour,minute' (0-23, 0-59)
view total mileage of all three trucks: input 'total'
quit application: input q/quit
```

E. Screenshot of Code Execution




```
choose from the following:
view package info and status for a specific package at a specific time: input 'package Id #' and then time in the format 'hour,mintue' (0-23, 0-59)
view package info and status for all packages at a specific time: input 'all' and then time when asked in the format 'hour,minute' (0-23, 0-59)
view total mileage of all three trucks: input 'total'
quit application: input q/quit
total
truck 1: 38.00
truck 2: 42.70
truck 3: 25.20
total mileage: 105.90

choose from the following:
view package info and status for a specific package at a specific time: input 'package Id #' and then time in the format 'hour,mintue' (0-23, 0-59)
view package info and status for all packages at a specific time: input 'all' and then time when asked in the format 'hour,minute' (0-23, 0-59)
view total mileage of all three trucks: input 'total'
quit application: input q/quit
```

F1. Strengths of the Chosen Algorithm

The greedy algorithm keeps going through the list of packages and picks the next closest destination based on the current location, which makes its implementation easy. Also, since the algorithm iterates through the entire list to find the next destination, adding new deliveries to the list can be handled without backtracking and wasting any time.

F2. Verification of Algorithm

the algorithm delivers all packages before their deadlines while keeping total mileage under 140 miles and meeting all other requirements, such as delayed package arrivals and later address correction.

F3. Other Possible Algorithms

Two other possible algorithms that could have been used instead to meet all the requirements are Ant Colony Optimization (ACO) and Tabu search algorithms.

F3a. Algorithm Differences

Both algorithms require more complex computations, making them harder to implement than a simple greedy algorithm. ACO and Tabu search would also require a weighted graph for storing distance data instead of the 2D array used with the greedy algorithm. Both algorithms consider long-term choices to provide more optimal solutions, compared to the greedy algorithm that just picks the next best immediate choice. Tabu search would also need an additional data structure to store tabu paths, which means it will require more memory than a greedy algorithm would.

G. Different Approach

One major thing that I would do differently if I were to do this project again, is to assign truck packages using an algorithm instead of doing it manually. 3 lists would be initialized for truck1, truck2, and truck3. The algorithm would search all packages in order based on their ID, and check their special notes for keywords such as “delayed”, “truck 2”, “wrong address”, etc. Then it would determine which list to add the package ID to. Finally, each list would be iterated to add the package with the corresponding ID to the correct truck.

H. Verification of Data Structure

The Hashtable used for storing packages allows for proper storage of package objects with all their required components. It also allows searching for each package’s data using its ID as the key.

H1. Other Data Structures

Linked lists and binary search trees are two other data structures that could have replaced a hash table for storing package data.

H1a. Data Structure Differences

A linked list and binary search tree (BST) would have been simpler to implement than a hash table. However, they both provide less efficient average search speeds of O(n) and O(logn), respectively, compared to the average search speed of a hash table which is O(1). A time complexity of O(n) and below is still an efficient search speed. Each node’s value on the linked list would be a package object with all the required components plus an ID component to allow searching the list based on a package ID. BST would have nodes ordered ascending based on their keys, which would be package IDs, allowing search by ID, and the values would be package objects with the required components. Hash tables on the other hand are not ordered. Also, while both BST and hash table have a worst-case space complexity of O(n), a hash table tends to keep a bigger array than it needs, making it less memory efficient than a BST.

I. Sources

GeeksforGeeks. (2020, May 17). *Introduction to ant colony optimization.* retrieved June 5, 2024, from https://www.geeksforgeeks.org/introduction-to-ant-colony-optimization

GeeksforGeeks. (2024, April 29). *What is tabu search?* retrieved June 5, 2024, from https://www.geeksforgeeks.org/what-is-tabu-search

GeeksforGeeks. (2023, April 26). *Advantages of BST over hash table*. retrieved June 5, 2024, from https://www.geeksforgeeks.org/advantages-of-bst-over-hash-table