CS 150: Project III

Due: 11:55pm, Sunday May 7, 2017

Introduction

Project Description

The same popular coffee shop chain from Project I and II was really impressed with your work in the last two projects. They are hiring you to manage the company's logistics in the city of Coffeetown. The roads in Coffeetown are on a 100x100 perfect square grid plan, where the grid sizes are equal throughout the city. The horizontal roads in Coffeetown are labeled *1st St* to *100th St* from South to North and the vertical roads are labeled *1st Ave* to *100th Ave* from East to West.

Your company's coffee shops are always located at intersections of the roads. Each shop has a unique ID and its location is given by a tuple (x, y) meaning the intersection of xth Ave and yth St. Every night, each coffee shop of your company sends an order to you, listing the weights of the cargos to be delivered to the shop the next day $(w1, w2, \ldots, w_k)$. The cargos are allowed to arrive in different shipments, as long as they are all delivered the next day.

Your company's warehouses are also located at intersections of the roads. Similarly, each warehouse has a unique ID and its location is given by a tuple (x, y). Each warehouse has a limited supply of trucks, except the main warehouse which is located at $(1, 1)$ and has more than enough supply of trucks.

Your job as a manager is to schedule the daily deliveries from the warehouses to fulfill the orders from the shops. You need to decide how many trucks are needed, and the route and the delivery schedule for each truck.

Your goal is to minimize the total distance traveled by all the trucks under the following assumptions:

1. All trucks have the same capacity $C = 500$ that cannot be exceeded.
2. All trucks originate from one of the company's warehouses and return to the same warehouse at the end of the trip.
3. All trucks make at most one trip in a day.
4. All scheduled deliveries can be made within a day.
5. All warehouses have unlimited supply of the required cargos.

Note: Optimally solving this problem in general case is hard (NP-hard to be specific), but simple "greedy" approaches can find a practical solution for this specific problem.

Assignments

Your program will take the following input:

1. A file `shops.txt`, in which the first line is n, the number of shops in the city, followed by n lines of the format $shopId\ \ (x, y)\ :\ w_1, w_2, \ldots, w_k$, indicating the ID of the shop, its location, and the weights of the cargos in its order.

2. A file `warehouses.txt`, in which the first line is m, the number of warehouses in the city, followed by m lines of format $warehouseId\ \ (x, y)\ :\ t$, indicating the ID of the warehouse, its location, and the number of trucks at this warehouse. The last warehouse is the main warehouse.

The output is a list of trucks, their initial warehouses, their route and their delivery schedules at each shop in their route. You will also output the total distance traveled by the all the trucks.

Guidelines

1. You should use the appropriate data structures and efficient algorithms. Make appropriate use of object-oriented design concepts such as abstract class, encapsulation, modularization, etc.

2. Your analysis and conclusions should be supported by mathematical reasoning and experimental data.

Report

The guideline for wiring the project report is the same as that of Project 1 and 2.

Grading

The project is to be completed either individually or by a team of two. **However, each individual member must design and conduct their own experiments, collect and analyze their own data, and write their own reports individually. In addition, each team must submit a common statement signed by both members detailing the contributions of each member along with your final submission. If you choose to work in a team, you need to let me know in advance**. Your project will be graded on the following criteria (assuming the program compiles and runs):

1. soundness of the analysis

2. correctness of the program

3. documentation (methods and classes) including javadoc

4. unit testing and the ability to allow more testing

5. object oriented design

6. quality of the analysis and simulation

7. quality of the project report