**Homework 4 - VRPPD**

Name1 Surname1\*, Name2 Surname2, Name3 Surname3  
*\**Group representative

# Problem description

GlobalEat is a prominent food delivery platform, its work consists in picking up orders from the most famous restaurants of a city and delivering them to the final customers who requested them. Now it faces the challenge of optimizing its sequence of customers visits that each vehicle should perform to minimize their total travel time, as well as the number of vehicles necessary to obtain the optimal solution. This problem is distinguished as a VRPPD problem in which a fleet of vehicles must fulfill a set of transportation requests. Each transportation request includes a pickup node, the corresponding delivery one and the demand to be transported between these two nodes. VRPPDs can be divided in three categories, which are one-to-one, many-to-many and one-to-many-to-one, the problem we’re solving is a one-to-one problem as each request has a single pickup and a single delivery. The input of the problem includes the geographic coordinates for both the depot, the restaurants and the customers, the serving time of each node, and the demand for each request. Furthermore, we have constraints on the capacity of vehicles and maximum duration of their journey.

# Mathematical model

*The mathematical model of the VRPPD problem can be represented as a objective function and a list of parameters, variables and constraints as follows.*

*2.1 Parameters*

*(1) V: Set of vertices*

*(2) P: Set of pickup nodes i = 1,…,n*

*(3) D: Set of delivery nodes i = n+1,…, 2n*

*(4) N: P∪D*

*(5) A: Set of arcs (i,j) with i,j ϵ V*

*(6) K: Set of vehicles k adopted to visit all the nodes*

*(7) : Demand at node i ϵ P (= 0 at the depot, -qi+n at corresponding delivery node)*

*(8) : Travel time of the arc (i,j)*

*(9): Service time at node i (to pickup or dropoff products)*

*(10) L: Maximum duration of a route*

*(11) Q: Vehicle capacity*

*2.2 Variables*

*(1) is a binary variable. It’s equal to 1 if arc (i,j) is traversed by vehicle k, otherwise 0.*

*(2) : load of the vehicle k after visiting node i.*

*(3) : time at which vehicle k begins service at node i.*

*2.3 Objective Function*

*The aim of the objective function is to Minimize the total travel time of the arcs traversed by all the vehicles.*

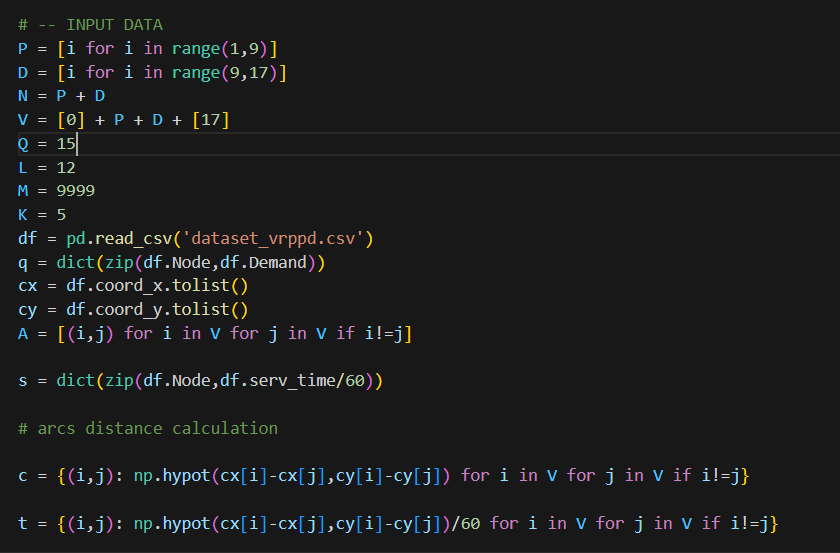
*2.4 Constraints*

*(1) (2) (3) (4) (5)*

*(6)(7) (8) (9) (10)(11) (12) (13)*

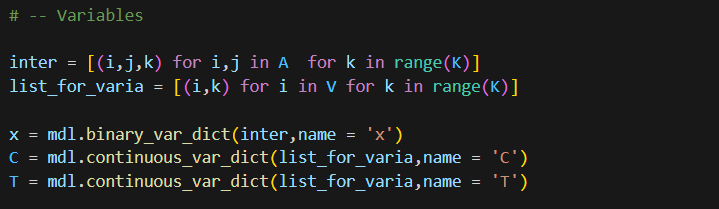
# Main code components

*3.1 Input*

**

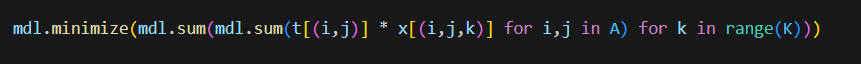
*This part is to implement the input data needed for the Problem. The definition of the parameters are the same with 2.1. M is a large number which will be used for a certain constraint. K is the number of available vehicles. The distances between nodes are calculated as euclidean distance and are stored in a dictionary c. Known the value for speed of vehicles to be 60km/h, the dictionary t is for the travel time between every couple of nodes, represented in the unit of hours.*

*3.2 Variables*

**

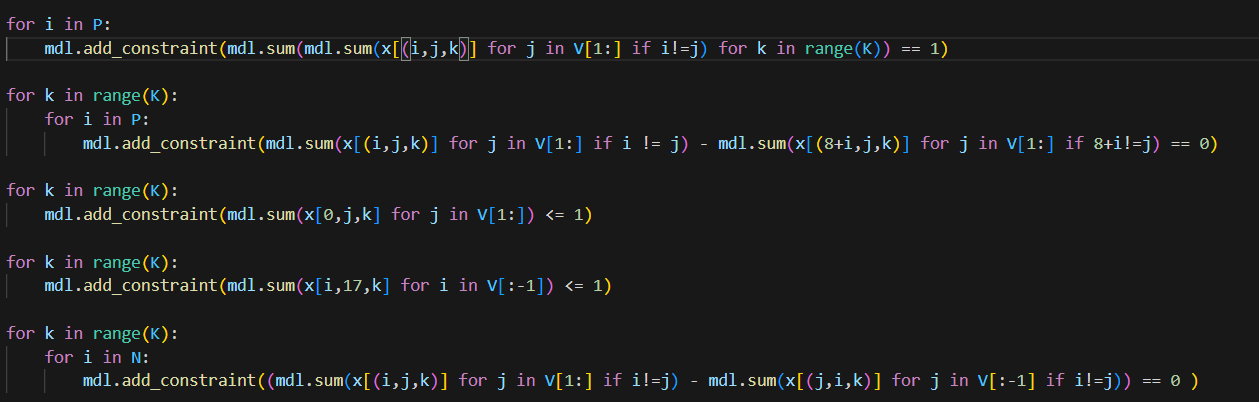
*At first we define two list to define our variables. Inter stands for (i,j,k) which is the key for x. And list\_for\_varia is a list for all the combinations of (i,k) which is the key for C and T. The values in x are binary while continuous valur for C and T since they reoresent load and time.*

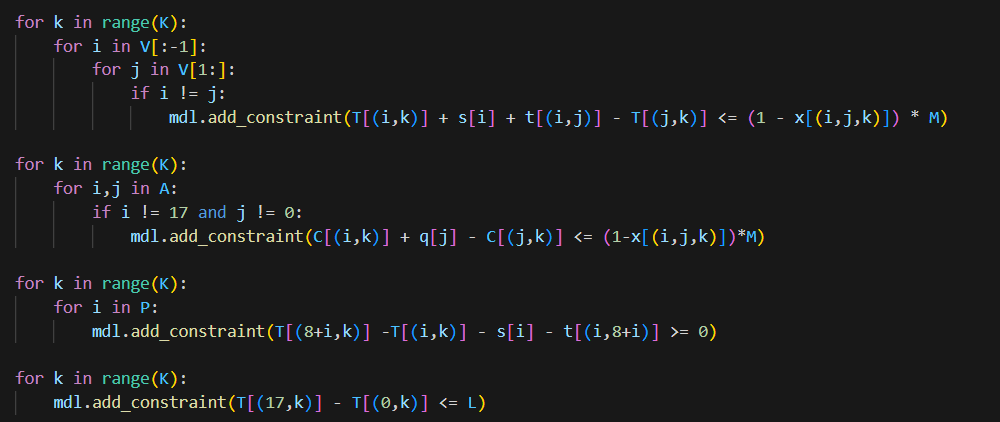
*3.3 Objective function*

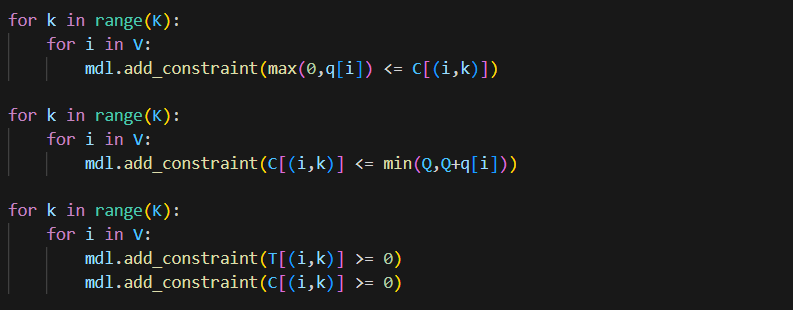
**

*The objective function is to minimize the the total travel time of the arcs traversed by all the vehicles.*

*3.4 Constraints*

**

**

**

*In the code of constraints, i widely use V[1:] and V[:-1]. V[1:] means all the nodes except for the node 0 which is the entry depot. V[:-1] means all the nodes expect for the node 17 which is the exit depot.*

*The interpretation of the constraints are as follows in order:*

*constraint 1: Each request must be served exactly once.*

*Constraint 2: Pickup and delivery nodes of same request must be assigned to the same vehicle.*

*Constraint 3 and 4: Each vehicle travels at most one route starting and ending at the depot.*

*Constraint 5: Flow conservation constraint: number of arcs entering and exiting each node must be the same.*

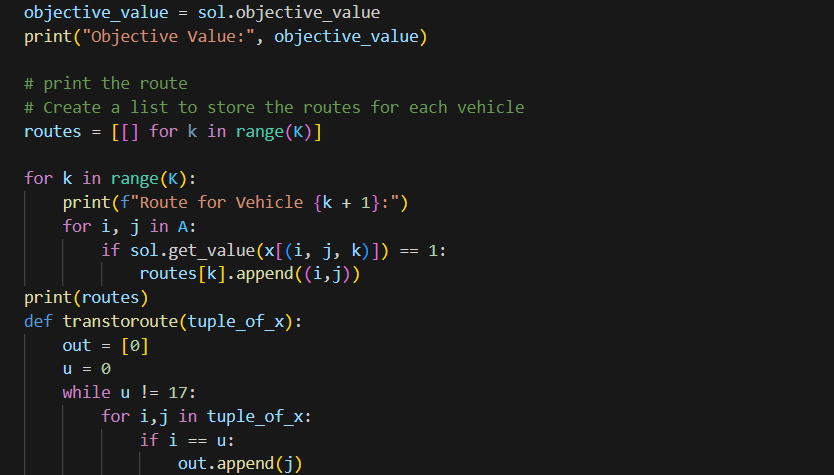
*Constraint 6 and 7: Consistency of time and load variables.*

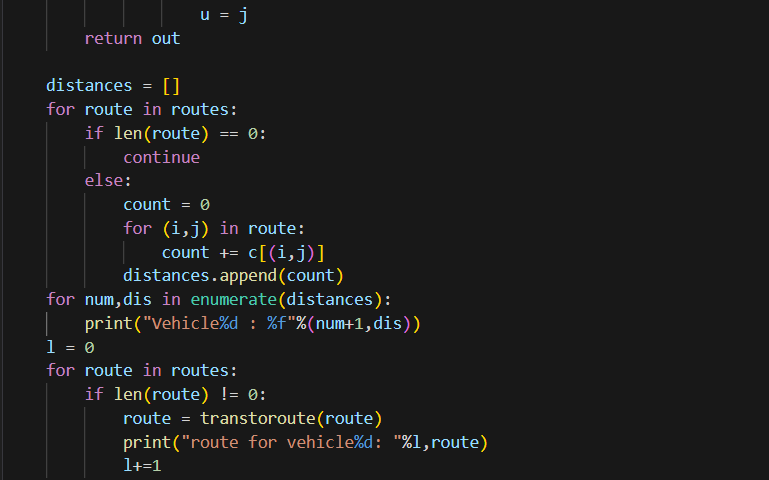
*Constraint 8: Precedence constraint: each pickup point must be visited before the corresponding 𝑇 delivery one.*

*Constraint 9: Route duration constraint.*

*Constraint 10: Vehicle capacity constraint.*

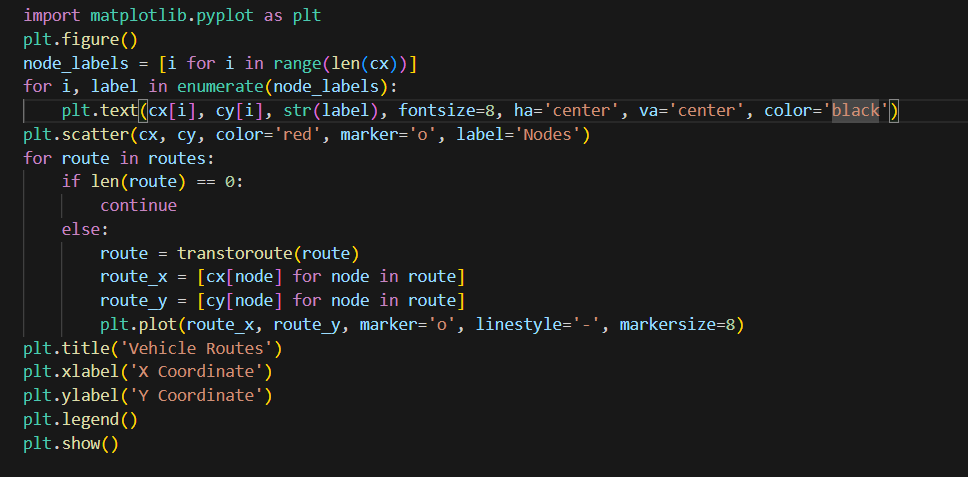
*3.5 Visualize the results*

**

**

*Routes is a list of list, each list contains the arcs the specific vehicle has traversed. The input of the function transtoroute is list of arcs traversed by a vehicle, the output is a list of nodes which represents the route of a vehicle. Then print the route. Also, we compute the total travel distance for each used vehicle and store it in a list ‘distances’. Then print the total distance for each vehicle.*

*3.6 Draw the map and routes*

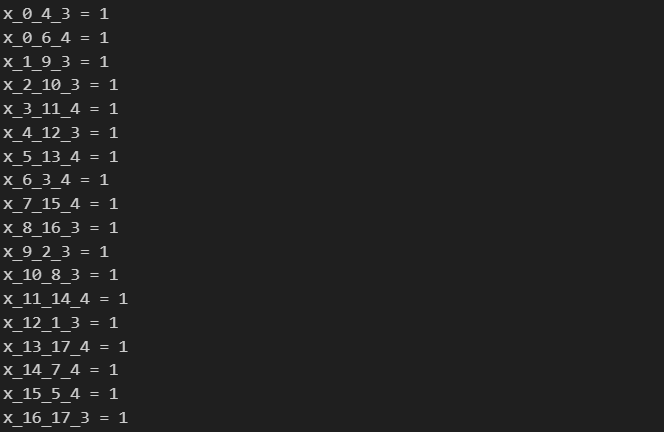
**

*This part is to put every node on a map and draw the route for each vehicle.*

# Results and insights

*The results we obtain are as follows.*

*At first, the solution for x is:*

**

*We set K equal to 5, which means 5 vehicles are available. But from the solution of x, only 2 vehicles are used to obtain the optimal solution. The route for each vehicle is:*

*route for vehicle1: [0, 4, 12, 1, 9, 2, 10, 8, 16, 17]*

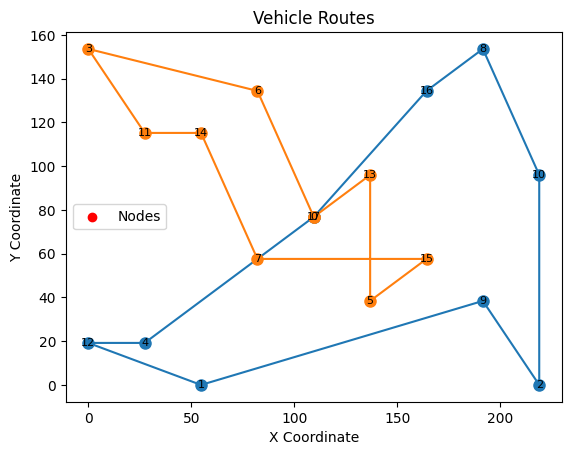
*route for vehicle2: [0, 6, 3, 11, 14, 7, 15, 5, 13, 17]*

*The solution is feasible as each route begins from 0 and end at 17. And all the requests are finished. Furthermore, for each request, each pickup point is visited before the corresponding delivery one. The travel distance for each vehicle is:*

*Vehicle1 : 647.502906km*

*Vehicle2 : 492.870712km*

*After all, the total travel time which is to minimize is 19.006 hours. To visualize the route of the vehicles, the graph is as follows:*

**

*There are two routes on the map since 2 vehicles are used to obtain the optimal solution.*