Question 1:

The problem is to find a schedule that uses the fewest number of students and meets all daily workforce requirements and the objective is to minimize the number of students needed.

The main motivation for this problem is to find a schedule that uses the fewest number of students and meets all daily workforce requirements and to find a schedule that uses the fewest number of students and meets all daily workforce requirements.

The data given is a table that has the number of students available for each day of the week.

The model we have created is an LP model that relates five-day shift schedules to daily numbers of students available and implements a solution to find a schedule that uses the fewest number of students and meets all daily workforce requirements.

We used pulp library to create the model and solve the model because it is a linear programming problem and we have to use the pulp library to solve the model to find the optimal solution.

We will use the data given and the LP model to solve the problem and in that model we have used the function create\_model to create the model, here we have used the function solve\_model to solve the model and in that function we have used the function get\_solution to get the solution from the model.

Question 2:

This is an Linear Programming problem to find the optimal solution. The objective is to minimize the total cost of shipping the automobiles from the plants to the regions that stays within the plant’s capacities while also meeting regional demands.

The model should use binary variables to represent the assignment of automobiles to regions.

The main motivation for this problem is to show that the problem can be solved using a simple LP model and to obtain the minimum total cost of shipping the automobiles from the plants to the regions that stays within the plant’s capacities while also meeting regional demands.

We have used the data tables given to find the desired results.

We have used the pulp library to solve the problem because it is a linear programming problem and we have to obtain the solution using this library.

We have created a LP opimization model and added the objective function to it.

The model should use binary variables to represent the assignment of automobiles to regions in which the variables are named x\_i\_j where i is the plant number and j is the region number.

The constraints are entered in the model.

In second part of the problem, Elon Must wants to extend the model to include the unit production costs and the initial baseline model to include the unit selling prices.

we will add the following constraints to the model:

The objective function is added to 'prob' first.

The constraints are entered.

The problem data is written to an .lp file.

The problem is solved using PuLP's choice of Solver.

The status of the solution is printed to the screen.

Each of the variables is printed with it's resolved optimum value.

Question 3:

write a detailed description of the problem and the solution algorithm

This is a Linear Programming problem. In this problem, we are trying to find the minimum optimal solution.

The main objective function is the sum of the distances between the service centres and the customers.

The constraints are that each service centre must be in one city, and that each city must have one service centre.

The variables are binary variables that represent the location of the service centres.

The problem is solved using PuLP's choice of Solver and the LpMinimize objective function to find the minimum total distance travelled.

We have created an LP model and added the objective function and constraints to the model so that it can find the minimum total distance travelled.

The model depends on the data of the customers and the service centres being located in different cities so that it can fetch the distance between the two cities.

The model also depends on the assumption that each customer is assigned to one service centre.

If we decide to locate3 the different service centres in the cities, then we can assign the customers in each city to one of the service centres.

APPENDIX:

1. For first problem, the Linear Programming model with the help of Pulp library:

model = pulp.LpProblem("Enactus", pulp.LpMinimize)

# Create variables

x = pulp.LpVariable.dicts("x", (data.index, data.columns), 0, 1, pulp.LpInteger)

# Create objective function

model += pulp.lpSum([x[i][j] \* data.loc[i, j] for i in data.index for j in data.columns])

# Create constraints

for i in data.index:

for j in data.columns:

model += x[i][j] <= data.loc[i, j]

return model

Here, we can see the variables used clearly and the functionality of the variables performing according to the data prvided.

1. For second problem, the Linear Programming model can be described as:

prob = pulp.LpProblem("Tesla Automobile Assignment", pulp.LpMinimize)

# Create a dictionary of the variables and set the lower and upper bounds

x = {}

for i in range(1, 11):

for j in range(1, 11):

x[i, j] = pulp.LpVariable("x\_" + str(i) + "\_" + str(j), cat='Binary')

# The objective function is added to 'prob' first

prob += pulp.lpSum([x[i, j] \* distance[i, j] for i in range(1, 11) for j in range(1, 11)])

# The constraints are entered

for i in range(1, 11):

for j in range(1, 11):

prob += x[i, j] <= capacity[i]

prob += x[i, j] <= demand[j]

prob += x[i, j] >= 0

The model performing the function of minimizing the cost of shipping the automobiles so that it can find the cheapest way to shipping them. Here we assigned variables to the Pulp library constraints according to the capacity and demand so that they can perform in an optimal manner.

1. For the third problem, Linear Programming model can be interpreted as:

prob = pulp.LpProblem("Service Centre Assignment", pulp.LpMinimize)

# Create a dictionary of the variables and set the lower and upper bounds

x = {}

for i in range(1, 11):

for j in range(1, 11):

x[i, j] = pulp.LpVariable("x\_" + str(i) + "\_" + str(j), cat='Binary')

# The objective function is added to 'prob' first

prob += pulp.lpSum([x[i, j] \* distance[i, j] for i in range(1, 11) for j in range(1, 11)])

# The constraints are entered

for i in range(1, 11):

prob += pulp.lpSum([x[i, j] for j in range(1, 11)]) == 1

prob += pulp.lpSum([x[j, i] for j in range(1, 11)]) == 1

This model implements the variables that determines the locations of service centres and then assigns customers to these service centres to minimize the total annual distance travelled by using the variables through the Pulp library and after it can perform the operation to minimize the total annual distance in an optimal way.