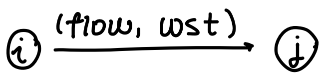
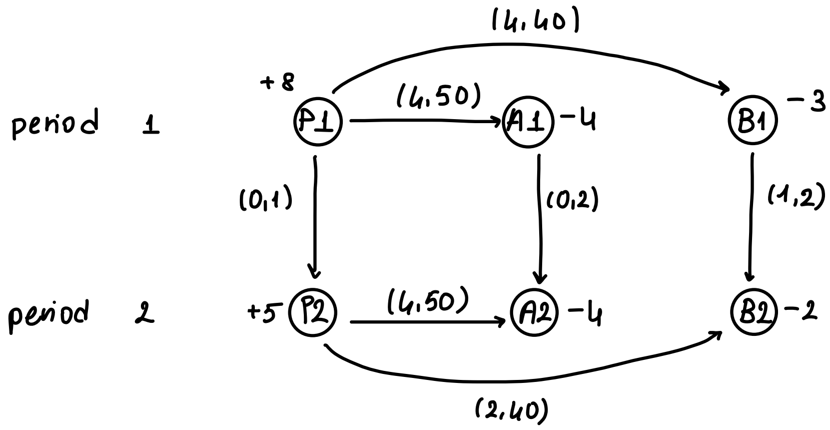
IE 479 - Study Set 2 Solutions

Question 1

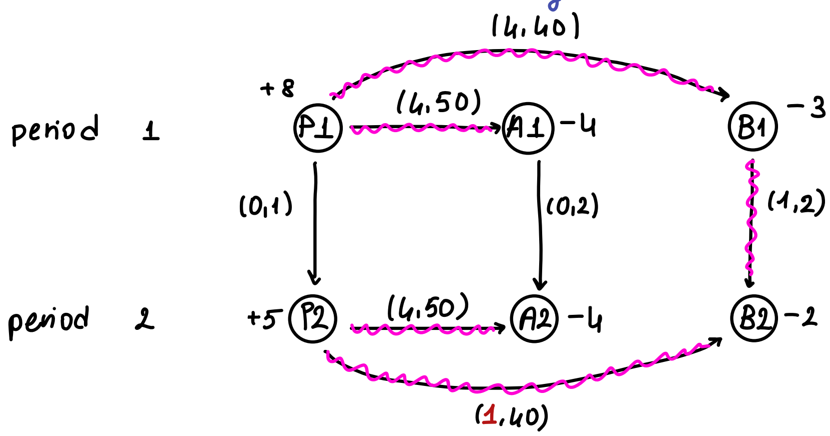




sending 2 units to B from P at period 2 does not satisfy the flow balance.

$$\text{cost} = (8 \times 25) + (5 \times 24) + (4 \times 50) + (4 \times 40) + (4 \times 50) + (2 \times 40) + 2 = 962$$

⇓ fix feasibility



$$\text{cost} = (8\times25) + (5\times24) + (4\times40) + (4\times50) + 2 + (4\times50) + 40 = 922$$

arcs in the tree :

$(P1, A1) \rightarrow \pi_{P1} - \pi_{A1} = 50$
$(P1, B1) \rightarrow \pi_{P1} - \pi_{B1} = 40$
$(B1, B2) \rightarrow \pi_{B1} - \pi_{B2} = 2$
$(P2, A2) \rightarrow \pi_{P2} - \pi_{A2} = 50$
$(P2, B2) \rightarrow \pi_{P2} - \pi_{B2} = 40$

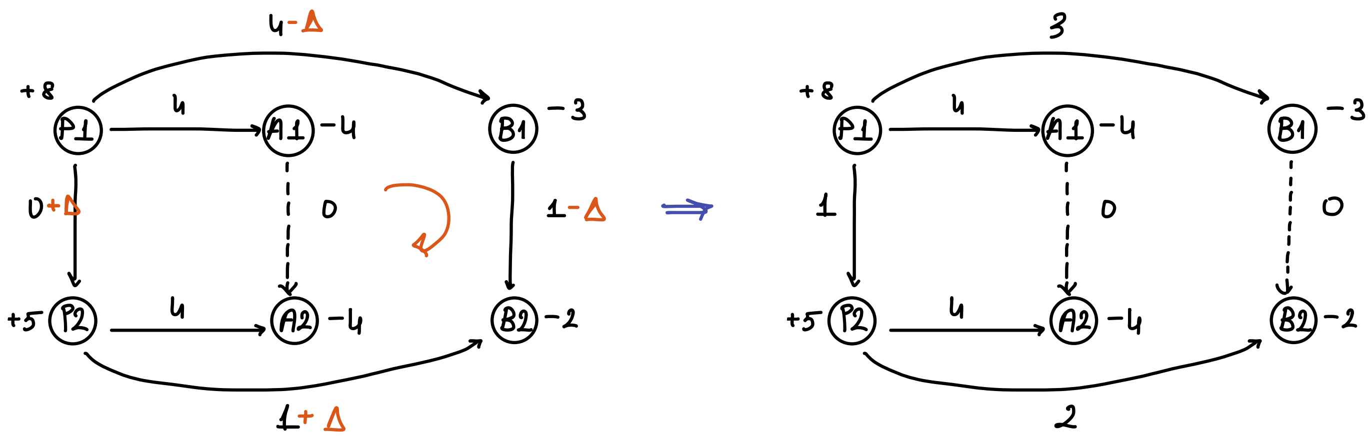
6 unknowns, 5 equations $\Rightarrow$ let $\pi_{A2} = 0$ $\Rightarrow$

$\pi_{P1} = 52$
$\pi_{P2} = 50$
$\pi_{A1} = 2$
$\pi_{A2} = 0$
$\pi_{B1} = 12$
$\pi_{B2} = 10$

arcs not on the tree :

$(P1, P2) \rightarrow X_{P1P2} = 0 \rightarrow$ check for $\pi_{P1} - \pi_{P2} \le C_{P1P2} \Rightarrow 52 - 50 \le 1$ ✗
$(A1, A2) \rightarrow X_{A1A2} = 0 \rightarrow$ " " $\pi_{A1} - \pi_{A2} \le C_{A1A2} \Rightarrow 2 - 0 \le 2$ ✓

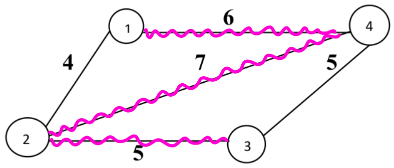
enter the arc (P1, P2) :



new cost : $(8 \times 25) + (5 \times 24) + (3 \times 40) + (4 \times 50) + 1 + (4 \times 50) + (2 \times 40) = 921$

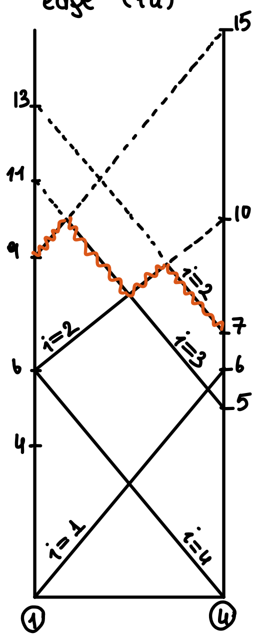
$\Rightarrow$ cost reduction : $922 - 921 = 1$ \$

Question 2

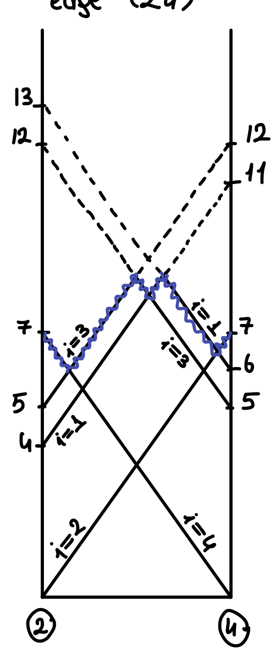


Hakimi p-center algorithm

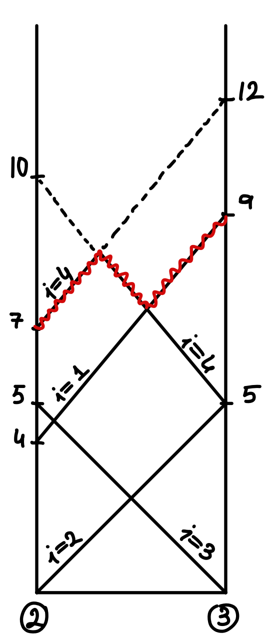
edge (14)



edge (24)

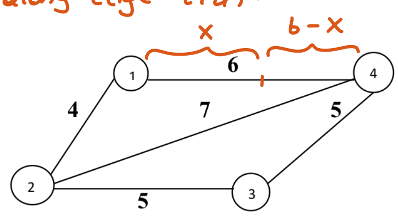


edge (23)



$i = 2,3$

along edge (14) :



distance from ② : $4+x$

" " ③ : $11-x$

$4+x = 11-x$

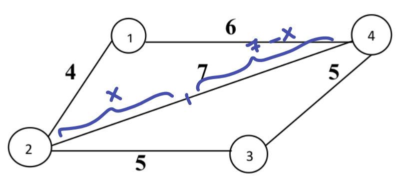
$2x = 7$

$x = 3{,}5$

$\Rightarrow$ worst dist. = 7,5

$i = 1,2,3,4$

along edge (2,4):



distance from ① : $4+x$ OR $13-x$

" " ② : x

" " ③ : $5+x$ OR $12-x$

" " ④ : $7-x$

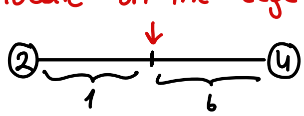
$5+x = 7-x$

$2x = 2$

$x = 1$

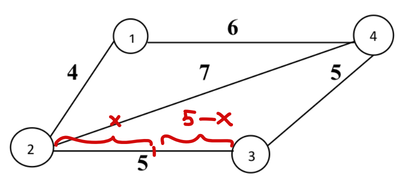
$\Rightarrow$ worst dist. = 6 ★

$\Rightarrow$ locate on the edge (24)



$i = 1,4$

along edge (23)



distance from ① : $4+x$

" " ④ : $10-x$

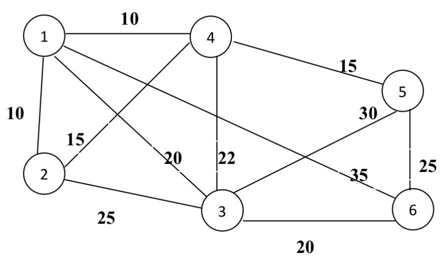
$4+x = 10-x$

$2x = 6$

$x = 3$

$\Rightarrow$ worst dist. = 7

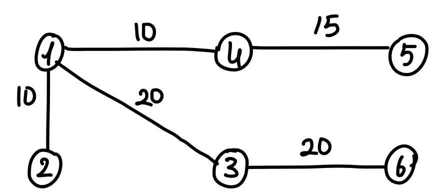
Question 3



$$
\begin{array}{c}
 \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6
\end{array}
\begin{array}{c}
\begin{array}{cccccc} 1 & 2 & 3 & 4 & 5 & 6 \end{array} \\
\begin{bmatrix}
0 & 10 & 20 & 10 & 25 & 35 \\
 & 0 & 25 & 15 & 30 & 45 \\
 & & 0 & 22 & 30 & 20 \\
 & & & 0 & 15 & 40 \\
 & & & & 0 & 25 \\
 & & & & & 0
\end{bmatrix}
\end{array}
$$

Christofides Algorithm

1. min cost spanning tree $\Rightarrow$



$E'_T = \{(12), (13), (14), (36), (45)\}$

↳ arcs on the tree

$V'_D = \{1, 2, 5, 6\}$

↳ odd edge vertices

2. least cost perfect match among V'_D

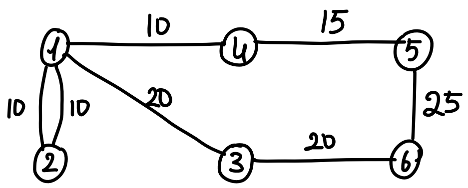
matches	cost
1-2 5-6	10+25 = 35 ★
1-5 2-6	25+45 = 70
1-6 2-5	35+30 = 65

$\Rightarrow E'_D = \{(12), (56)\}$ → edges in the match

$E'_H = E'_T \cup E'_D = \{(12), (13), (14), (36), (45), (56)\}$

↳ double

$H = (V', E'_H)$

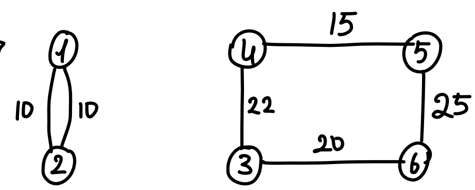


cost = 110

3. Vertices with degree > 2 $\Rightarrow$ 1 $\Rightarrow$ eliminate 2 nodes, add 1

a) eliminate (14) and (13), add (34) $\Rightarrow$

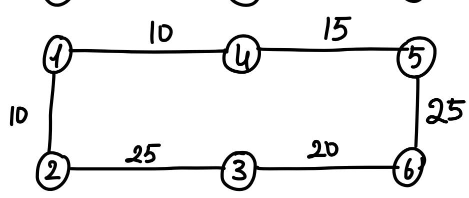
$\Delta = -10 - 20 + 22 = -8$



No tour, cannot be choosen !

b) eliminate (13) and (12), add (23) $\Rightarrow$

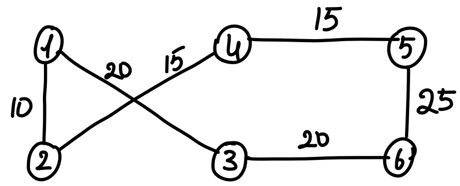
$\Delta = -20 - 10 + 25 = -5$



cost = 105

c) eliminate (12) and (14), add (24) $\Rightarrow$

$\Delta = -10 - 10 + 15 = -5$

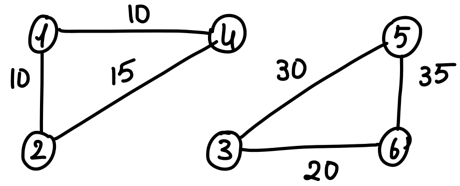


cost = 105

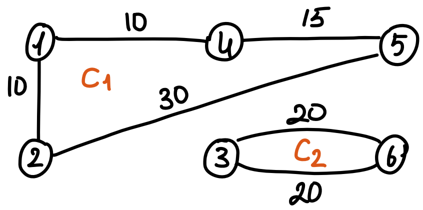
$\Rightarrow$ choose b) OR c)

Patching Heuristic

relaxed solutions :



cost = 120



cost = 105 ⇒ choose this one for the starting point

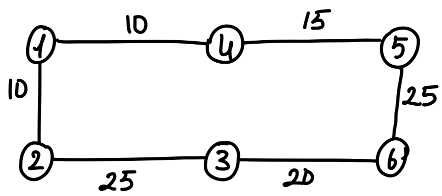
$N = \{1,2,3,4,5,6\}$
$C = \{C_1, C_2\}$

subtour number → P=2

⇒ merge C_1 and C_2 by removing arcs (25) and (36), adding (23) and (56)

most costly ones

[min cost perfect match : 2-3 5-6 ⇒ cost = 25 + 25 = 50 *
2-6 3-5 ⇒ cost = 45 + 30 = 75]



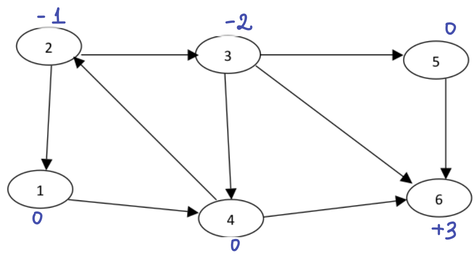
$\Delta = -30 - 20 + 25 + 25 = 0$

cost = 105   P=1 , stop !

Question 4

out > in ⇒ deficit , in > out ⇒ surplus

a)



S = set of surplus vertices
D = " " deficit "
C_{ij} = cost of shortest path from i to j
S_i = surplus of node $i \in S$
d_j = deficit " " $j \in D$

X_{ij} = # of chains added btw i and j

S (surplus) = {6} → $S_6 = 3$ (surplus of node 6)
D (deficit) = {2,3} → $d_2 = 2 - 1 = 1$ (deficit of node 2)
$d_3 = 3 - 1 = 2$ (deficit of node 3)

$C_{62} = 2$, $C_{63} = 1$

solve min $\sum_i \sum_j C_{ij} X_{ij}$

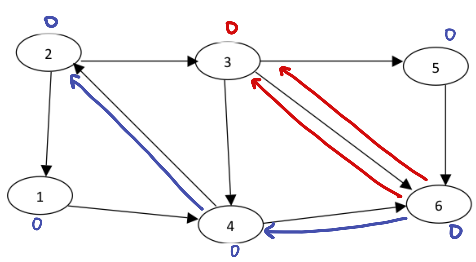
s.t. $\sum_{j \in D} X_{ij} = S_i \quad \forall i \in S$

$\sum_{i \in S} X_{ij} = d_j \quad \forall j \in D$

$X_{ij} \geq 0$ and integer

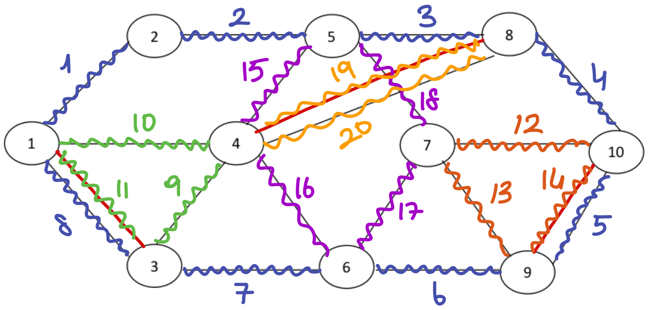
for i = 6 ⇒ $X_{62} + X_{63} = 3$
for j = 2 ⇒ $X_{62} = 1$
for j = 3 ⇒ $X_{63} = 2$

solution : $X_{62} = 1$, $X_{63} = 2$
OFV $= C_{62} \cdot X_{62} + C_{63} \cdot X_{63}$
$= 2 + 2 = 4$ //



⇒ all nodes have even degrees !

b)



* Graph is connected ✓
* Not all vertices have even edges ✗

$\Rightarrow$ odd degree vertices $= \{1, 3, 4, 8, 9, 10\}$

$\Rightarrow$ add chains to make all vertices even

↳ new chains : (13), (48), (910)

Now apply End Pairing Algorithm with additional edges :

1st cycle : 1 − 2 − 3 − 4 − 5 − 6 − 7 − 8

2nd cycle : 9 − 10 − 11

↳ merge $\Rightarrow$ 1 − 2 − 3 − 4 − 5 − 6 − 7 − 9 − 10 − 11 − 8

3rd cycle : 12 − 13 − 14

↳ merge $\Rightarrow$ 1 − 2 − 3 − 4 − 12 − 13 − 14 − 5 − 6 − 7 − 9 − 10 − 11 − 8

4th cycle : 15 − 16 − 17 − 18

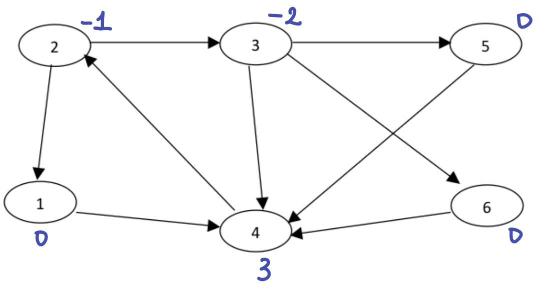
↳ merge $\Rightarrow$ 1 − 2 − 15 − 16 − 17 − 18 − 3 − 4 − 12 − 13 − 14 − 5 − 6 − 7 − 9 − 10 − 11 − 8

5th cycle : 19 − 20

↳ merge $\Rightarrow$ 1 − 2 − 15 − 19 − 20 − 16 − 17 − 18 − 3 − 4 − 12 − 13 − 14 − 5 − 6 − 7 − 9 − 10 − 11 − 8

* all edges are visited, stop !

c)



$S = \{4\} \Rightarrow s_4 = 3$

$D = \{2, 3\} \Rightarrow d_2 = 1$

$d_3 = 2$

$C_{42} = 1$

$C_{43} = 2$

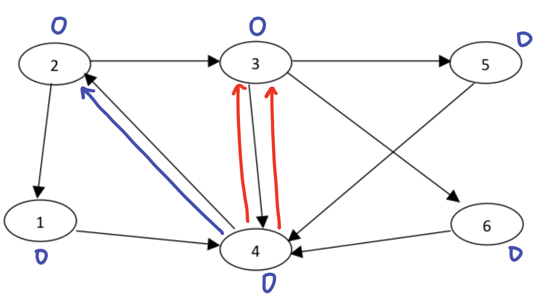
for $i = 4 \Rightarrow X_{42} + X_{43} = 3$

for $j = 2 \Rightarrow X_{42} = 1$

for $j = 3 \Rightarrow X_{43} = 2$

} solution : $X_{42} = 1$, $X_{43} = 2$

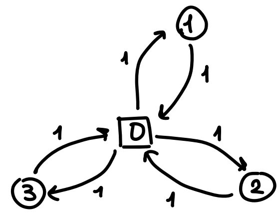
OFV $= 1 + 4 = 5$



$\Rightarrow$ all nodes have even edges and graph is strongly connected (Eulerian cycle)

Question 5

15 nodes (1 depot, 14 customers)
$Q = 20$ → vehicle capacity
$q_i = 5$ → customers' demands
$m = 4$ → # of vehicles
all arc costs are 1
$T = 5$ → max route length

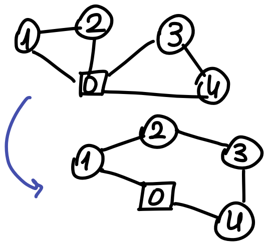


a) Savings Heuristic :

- Initially assign a route for each customer ⇒ total cost = 14x2 = 28

- Merge a pair of routes (lets say 1 and 2) ⇒ saving : $S_{ij} = C_{i0} + C_{0j} - C_{ij}$ (0 is depot)
$S_{12} = C_{10} + C_{02} - C_{12} = 1+1-1 = 1$

- Merge another pair (lets say 3 and 4) ⇒ $S_{34} = 1$

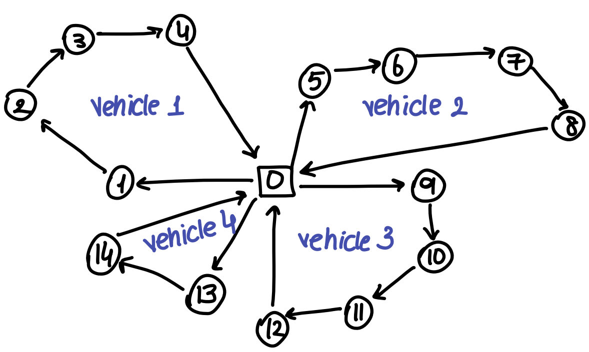


⇒ merge these two routes, saving is 3+3−5 = 1

⇒ total traveled distance = 1 x 5 = 5 = T ✓
total satisfied demand = 4 x 5 = 20 = Q ✓
} no more expending the route

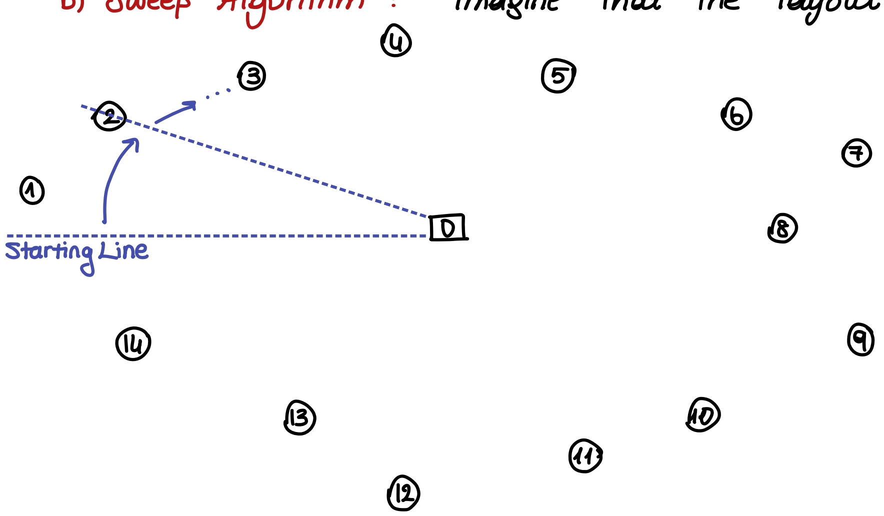
★ With this logic, we can end up with 3 routes with 4 nodes each and a route with 2 nodes, as follows :

↳ merge 5, 6, 7, and 8
↳ " 9, 10, 11, and 12
↳ " 13, and 14 (we have capacity, but no other customer is left)



4 routes :
total cost = 5+5+5+3 = 18
total saving = 28 − 18 = 10

b) Sweep Algorithm : Imagine that the layout of nodes are as follows :



⇒ Sorted customer list according to their angular position around the depot :
1-2-3-4-5-6-7-8-9-10-11-12-13-14

⇒ Add customers in order while satisfying capacity constraints :

Route 1 : 0→1→2→3→4→0
Route 2 : 0→5→6→7→8→0
Route 3 : 0→9→10→11→12→0
Route 4 : 0→13→14→0

Total cost = 18

c) Cluster First - Route Second :

Cluster 1 = {1,2,3,4}
Cluster 2 = {5,6,7,8}
Cluster 3 = {9,10,11,12}
Cluster 4 = {13,14}

total demand of each cluster $\leq$ Q
route length of each cluster $\leq$ T

total cost = 18

Solving TSP for each cluster $\Rightarrow$ same routes as before (since all arc costs are equal)

Which one is better? Since the problem is symmetric (all arc costs are equal), we don't see a difference in terms of cost and route assignments.

d) Inter route improvement? → since all routes gives the less costly option, there is no improvement

Intra route improvement? → 3 routes has achieved full capacity, adding customers to these routes is infeasible.

→ customer exchange between routes makes no improvement, since arc costs are equal.

Question 6 → Hungarian Method

$$\begin{array}{cc} & \begin{matrix}1 & 2 & 3 & 4 & 5\end{matrix} \\ \begin{matrix}1\\2\\3\\4\\5\end{matrix} & \begin{bmatrix}8 & 3 & 5 & 2 & 4\\4 & 5 & 3 & 6 & 7\\6 & 2 & 1 & 3 & 5\\3 & 7 & 2 & 4 & 3\\5 & 2 & 6 & 3 & 4\end{bmatrix} \end{array} \begin{matrix}\text{ROW MIN}\\ \rightarrow 2\\ \rightarrow 3\\ \rightarrow 1\\ \rightarrow 2\\ \rightarrow 2\end{matrix}$$

subtract ROW MIN from each row :

$$\begin{array}{cc} & \begin{matrix}1 & 2 & 3 & 4 & 5\end{matrix} \\ \begin{matrix}1\\2\\3\\4\\5\end{matrix} & \begin{bmatrix}6 & 1 & 3 & 0 & 2\\1 & 2 & 0 & 3 & 4\\5 & 1 & 0 & 2 & 4\\1 & 5 & 0 & 2 & 1\\3 & 0 & 4 & 1 & 2\end{bmatrix} \end{array}$$

COL MIN $\Rightarrow$ 1 0 0 0 1

subtract COL MIN from each column :

$$C' = \begin{array}{cc} & \begin{matrix}1 & 2 & 3 & 4 & 5\end{matrix} \\ \begin{matrix}1\\2\\3\\4\\5\end{matrix} & \begin{bmatrix}5 & 1 & 3 & 0 & 1\\0 & 2 & 0 & 3 & 3\\4 & 1 & 0 & 2 & 3\\0 & 5 & 0 & 2 & 0\\2 & 0 & 4 & 1 & 1\end{bmatrix} \end{array}$$

cover zero's with min # of lines $\Rightarrow$

$$C' = \begin{array}{cc} & \begin{matrix}1 & 2 & 3 & 4 & 5\end{matrix} \\ \begin{matrix}1\\2\\3\\4\\5\end{matrix} & \begin{bmatrix}5 & 1 & 3 & 0 & 1\\0 & 2 & 0 & 3 & 3\\4 & 1 & 0 & 2 & 3\\0 & 5 & 0 & 2 & 0\\2 & 0 & 4 & 1 & 1\end{bmatrix} \end{array}$$

5 Lines, n = 5
STOP !
Optimum sol. is achieved

$$\Rightarrow \begin{array}{cc} & \begin{matrix}1 & 2 & 3 & 4 & 5\end{matrix} \\ \begin{matrix}1\\2\\3\\4\\5\end{matrix} & \begin{bmatrix}5 & 1 & 3 & \boxed{0} & 1\\\boxed{0} & 2 & 0 & 3 & 3\\4 & 1 & \boxed{0} & 2 & 3\\0 & 5 & 0 & 2 & \boxed{0}\\2 & \boxed{0} & 4 & 1 & 1\end{bmatrix} \end{array}$$

Optimal assignment :

		cost
1 →	4	2
2 →	1	4
3 →	3	1
4 →	5	3
5 →	2	2

total cost = 12