UNBALANCED TRANSPORTATION PROBLEM

Algorithm :>

↳ Take input of cost, source, destination as vector

↳ Check if balanced / unbalanced (Demand == supply)
↳ balanced

→ If unbalanced :>

Check :>

if demand < supply
→ create dummy destination

else
→ Can't solve as demand > supply

Ex (balanced transportation problem)

		8		
23				
		16		4 3 2 4
12				2 3 2 3
		13		1 1 2 2
9				
		7		

Ex (Unbalanced transportation problem)

		30		
50				
		25		4 3 2 4
40				2 3 2 3
		35		1 1 2 2
70				
		40		
160		130		

① Create dummy destination with cost 0

50	30	4 3 2 4 0
	25	2 3 2 3 0
40		1 1 2 2 0
	35	...
70		
	40	
	30	
160	160	

→ North-West Corner Rule:-

	30	25	35	40	30
0 20 50	30	20	0	0	0
0 35 40	0	5	35	0	0
0 30 70	0	0	0	40	30

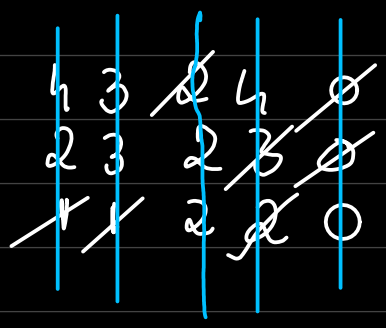
0 5 0 0 0

11

→ Minimum Cost Rule

[source] [destination]

	30
50	
	25
40	
	35
70	
	40
	30
160	160



	0	0	0	0
	30	25	35	40	30
0 15 50			35		15
0 15 40				25	15
0 15 40 70	30	25		15	

if dest == 0
    break

④ 2 1

1) min = 4 loc = 0
2) min = 2 loc = 1
3) min = 1 loc = 2

4 3 2 4 0
2 3 2 3 0
1 2 2 0 0

0 0 20 200
30 25 35 40 30

30 50 0 0 0 20 30
0 20 40 0 0 20 20
15 40 70 30 25 15
0

① 421 loc = 2

② 331

0
30 25 35 40 30

50
40
40 70 30

4 3 2 4 0
2 3 2 3 0
1 1 2 2 0

	8	16	13	7
23	8 (11)	8 (12)	0 (13)	7 (14)
12	0 (21)	0 (22)	12 (23)	0 (24)
9	0	8 (32)	1	0

$C_{11} = U_1 + V_1$

$C_{12} = U_1 + V_2$

$C_{14} = U_1 + V_4$

$C_{23} = U_2 + V_3$

$C_{32} = U_3 + V_2$

$C_{33} = U_3 + V_3$

8

23

16

12

13

9

7

4 3 2 4

2 3 2 3

1 1 2 2

U [supply]

V [demand]

$C_{11} = U_1 + V_1$

$4 = U_1 + V_1$

$C_{12} = U_1 + V_2$

$3 = U_1 + V_2$

$4 = U_1 + V_4$

$2 = U_2 + V_3$

$U_1 = 0$

$V_1 = 4$

	8	16	13	7
23	8	15	0 (23)	0
12	0	22 1	11	0
9	0*	0	2	7
	31	32	33	

$u_1 = 0$

$u_2 = 0$

$u_3 = 0$

$v_1 = 4$

$v_2 = 3$

$= 2$

$u_4 = 2$

4 3 2 4

2 3 2 3

1 1 2 2

we cant get pts like 3,1 as it doesnt form a cycle u,v helps in setting cycle

$C_{13} - u_1 - v_3$

$2 - 0 - 2 = 0$

$C_{14} - u_1 - v_4$

$4 - 0 - 2 = 2$

$C_{21} - u_2 - v_1$

$2 - 0 - 4 = -2$

$C_{32} - u_3 - v_2$

$1 - 0 - 3 = -2$

$C_{24} - u_2 - v_4$

$3 - 0 - 2 = 1$

$C_{31} - u_3 - v_1$

$1 - 0 - 4 = -3$

	8	16	13	7
23	8	15	∞	0
12	0	1	11	0
9	0	*0	2	7

loc = 2, 1

lc = 0
mi = 2 (loc i)
mj = 1 (loc j)

Sign

a a a a
a a a a
a a a a

loop

1 1 0 0
0 1 1 0
0 0 1 1

k = 0
i = 0

j = 0 lc = 1
j = 3 lc = 2

if only 1 basic cell in neighbourhood column then its zero

lc ≠ 1
lc = 0

next rows same algorithm

i = 1

j = 0 lc = 0
j = 3 lc = 2
lc ≠ 1
lc = 0

0 1 0 0
0 1 1 0
0 0 1 0

0 0 0 0
0 1 1 0
0 0 1 0

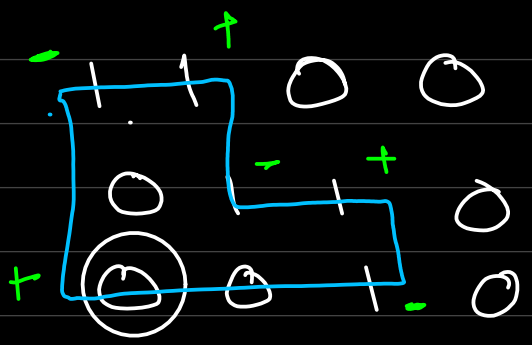
0 1 0 0
0 1 1 0
0 0 1 1

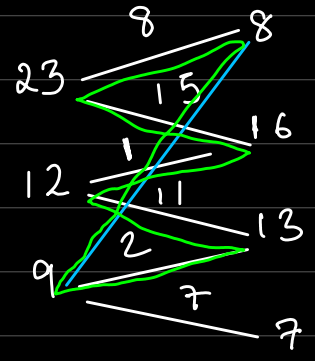
0 1 0 0
0 1 1 0
0 0 1 0

$k \geq 1$

$k \leq 2$

1, 2	1	1	0	0
	0	1	1	0
	0	0	1	1





	8	16	13	7
23	8	15	0	0
12	0	1	11	0
9	0	0	2	7

8 12

	8	16	13	7
23	7	16	0	0
12	0	0	12	0
9	1	0	1	7

4 3 2 4
2 3 2 3
1 1 2 2

$u_1 = 0$

$2 - (u_1 + v_3) = 2 - (0 + 5) = -3$ $\quad u_2 = -3$

$4 - (u_1 + v_4) = 4 - (0 + 5) = -1$ $\quad u_3 = -3$

$2 - (u_2 + v_1) = 2 - (-3 + 4) = 1$ $\quad v_1 = 4$

$3 - (u_2 + v_2) = 3 - (-3 + 3) = 3$ $\quad v_2 = 3$

$v_3 = 5$

$v_4 = 5$

N	N	-3	-1	
1	3	N	1	
N	1	N	N	

1	1	0	0
0	0	1	0
1	0	1	1

		0	1	2	3
		~~250~~	350	400	200
0	300	0	300	0	0
1	400	250	0	0	150
2	500	0	50	400	50

3 1 7 4
2 6 5 9
8 3 3 2

3, 2, 8

3, -1, 8

$3-(-7)=10$

$7-(0+1)=6$

$4-(0+0)=4$

$6-(9+1)=-4$

$5-(9+1)=\underline{-5}$

u 2
u

0 1 0 0
1 0 0* 1
0 1 1 1

0 0 0 0
0 0 0 1
0 0 1 1

0 1 0 1
0 1 0 1 (1,3)
0 0 0 0

a a a a
a a a a
a a a a

0 1 1 1
1 1 0 0
1 0 0 0

0 1 0 +
0 1 0 0
0 0 0 0

sign[1][3] = +

a a a a
a a a +
a a a a

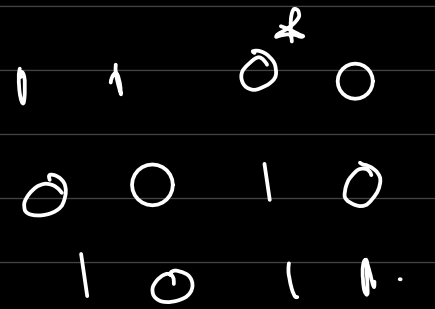
i = 1
j = 0

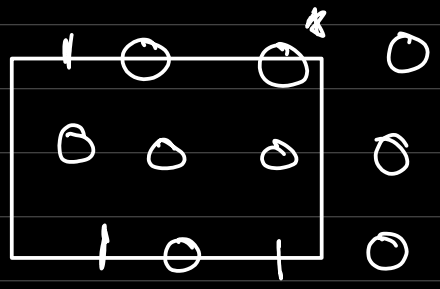
a a a a
a - a +
a a a a

a + a a
a - a +
a a a a

i = 0

a





c[0][0]==1

j=0

-1 0 2 0
0 0 0 0
1 0 1 0

c[2][0]=1 i=2

-1 0 2 0
0 0 0 0
1 0 1 0

cycle[2]

1 1 0 0
0 1 1 0
1 0 1 0

0 0 0 0
0 0 0 0
1 0 -1 0

k = 1

i = 2
j = 2

0 0 0 0
0 0 1 0
1 0 -1 0

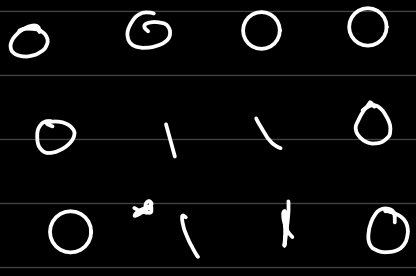
i = 1
j = 2

0 0 0 0
0 -1 1 0
1 0 -1 0

i = 1
j = 1

0 1 0 0
0 -1 1 0
1 0 -1 0

i = 0
j = 1



i=2