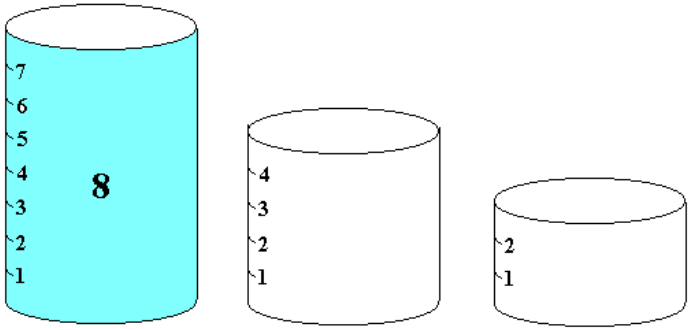
Milkman's Problem
A Graph Theoretical Model

The problem

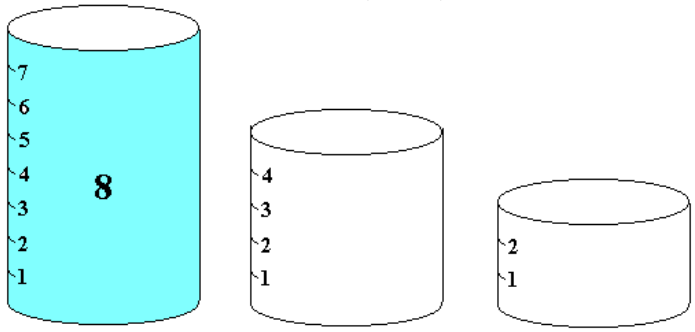
A milkman has three containers of capacities 8 gallons, 5 gallons, and 3 gallons. The 8-gallon container is full of milk. How can he divide the milk into two 4-gallon portions without using anything but his three containers?



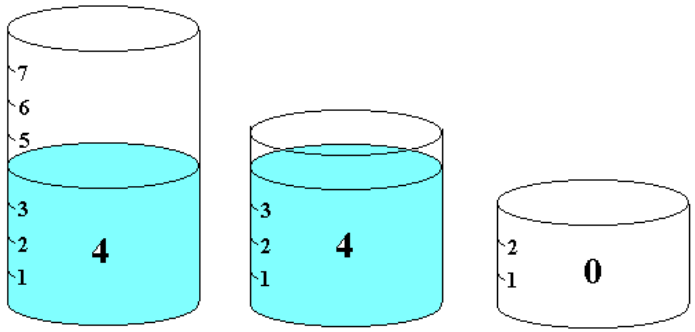
Define the *state* of the system to be (x,y,z) where

x= # gallons of milk in 8-gallon container
y= # gallons of milk in 5-gallon container
z= # gallons of milk in 3-gallon container

The initial state, then, is (8,0,0)



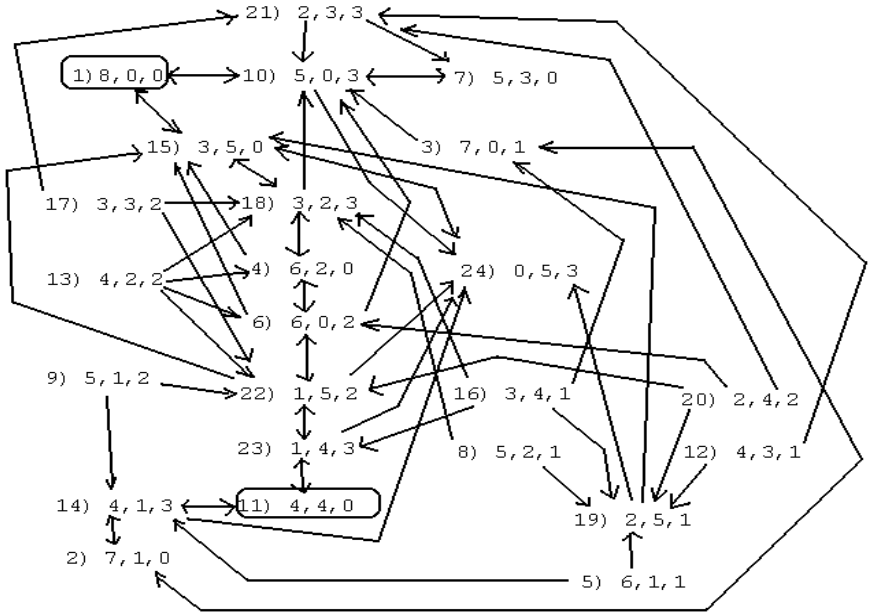
The desired state is (4,4,0)



What are the intermediate states to get from (8,0,0) to (4,4,0)?

Possible states are

#	State	#	State	#	State
1	(8,0,00	9	(5,1,2)	17	(3,3,2)
2	(7,1,0)	10	(5,0,3)	18	(3,2,3)
3	(7,0,1)	11	(4,4,0)	19	(2,5,1)
4	(6,2,0)	12	(4,3,1)	20	(2,4,2)
5	(6,1,1)	13	(4,2,2)	21	(2,3,3)
6	(6,0,2)	14	(4,1,3)	22	(1,5,2)
7	(5,3,0)	15	(3,5,0)	23	(1,4,3)
8	(5,2,1)	16	(3,4,1)	24	(0,5,3)



ADJACENCY

	8 7 7 6 6 6 5 5 5 5 4 4 4 4 3 3 3 3 2 2 2 1 1 0
	0 1 0 2 1 0 3 2 1 0 4 3 2 1 5 4 3 2 5 4 3 5 4 5
	0 0 1 0 1 2 0 1 2 3 0 1 2 3 0 1 2 3 1 2 3 2 3 3
1] *8 0 0*	0 0 0 0 0 0 0 0 0 1 0 0 0 0 1 0 0 0 0 0 0 0 0 0
2] *7 1 0*	0 0 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0
3] *7 0 1*	0 1 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0
4] *6 2 0*	0 0 0 0 0 1 0 0 0 0 0 0 0 0 1 0 0 1 0 0 0 0 0 0
5] *6 1 1*	0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 1 0 0 0 0 0
6] *6 0 2*	0 0 0 1 0 0 0 0 0 1 0 0 0 0 1 0 0 0 0 0 1 0 0 0
7] *5 3 0*	0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0
8] *5 2 1*	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0
9] *5 1 2*	0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 1 0 0
10] *5 0 3*	1 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1
11] *4 4 0*	0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 1 0
12] *4 3 1*	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 0 0 0
13] *4 2 2*	0 0 0 1 0 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0
14] *4 1 3*	0 1 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 1
15] *3 5 0*	1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1
16] *3 4 1*	0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 0 1 0
17] *3 3 2*	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 1 1 0 0
18] *3 2 3*	0 0 0 1 0 0 0 0 0 1 0 0 0 0 1 0 0 0 0 0 0 0 0 0
19] *2 5 1*	0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 1
20] *2 4 2*	0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 1 0 0
21] *2 3 3*	0 0 0 0 0 0 0 1 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0
22] *1 5 2*	0 0 0 0 0 1 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 1 1
23] *1 4 3*	0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 1 0 1
24] *0 5 3*	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0

REACHABILITY

	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4
1	1	1	1	1	0	1	1	0	0	1	1	0	0	1	1	0	0	1	0	0	0	1	1	1
2	1	1	1	1	0	1	1	0	0	1	1	0	0	1	1	0	0	1	0	0	0	1	1	1
3	1	1	1	1	0	1	1	0	0	1	1	0	0	1	1	0	0	1	0	0	0	1	1	1
4	1	1	1	1	0	1	1	0	0	1	1	0	0	1	1	0	0	1	0	0	0	1	1	1
5	1	1	1	1	1	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	1
6	1	1	1	1	0	1	1	0	0	1	1	0	0	1	1	0	0	1	0	0	0	1	1	1
7	1	1	1	1	0	1	1	0	0	1	1	0	0	1	1	0	0	1	0	0	0	1	1	1
8	1	1	1	1	0	1	1	1	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	1
9	1	1	1	1	0	1	1	0	1	1	1	0	0	1	1	0	0	1	0	0	0	1	1	1
10	1	1	1	1	0	1	1	0	0	1	1	0	0	1	1	0	0	1	0	0	0	1	1	1
11	1	1	1	1	0	1	1	0	0	1	1	0	0	1	1	0	0	1	0	0	0	1	1	1
12	1	1	1	1	0	1	1	0	0	1	1	1	0	1	1	0	0	1	1	0	1	1	1	1
13	1	1	1	1	0	1	1	0	0	1	1	0	1	1	1	0	0	1	0	0	0	1	1	1
14	1	1	1	1	0	1	1	0	0	1	1	0	0	1	1	0	0	1	0	0	0	1	1	1
15	1	1	1	1	0	1	1	0	0	1	1	0	0	1	1	0	0	1	0	0	0	1	1	1
16	1	1	1	1	0	1	1	0	0	1	1	0	0	1	1	1	0	1	1	0	0	1	1	1
17	1	1	1	1	0	1	1	0	0	1	1	0	0	1	1	0	1	1	0	0	1	1	1	1
18	1	1	1	1	0	1	1	0	0	1	1	0	0	1	1	0	0	1	0	0	0	1	1	1
19	1	1	1	1	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	1
20	1	1	1	1	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	1	1	1	1	1
21	1	1	1	1	0	1	1	0	0	1	1	0	0	1	1	0	0	1	0	0	1	1	1	1
22	1	1	1	1	0	1	1	0	0	1	1	0	0	1	1	0	0	1	0	0	0	1	1	1
23	1	1	1	1	0	1	1	0	0	1	1	0	0	1	1	0	0	1	0	0	0	1	1	1
24	1	1	1	1	0	1	1	0	0	1	1	0	0	1	1	0	0	1	0	0	0	1	1	1

Row #1 of the first ten powers of the adjacency matrix

n	1	2	3	4	6	7	10	11	14	15	18	22	23	24
1	0	0	0	0	0	0	1	0	0	1	0	0	0	0
2	2	0	0	0	0	1	0	0	0	0	1	0	0	2
3	0	0	0	1	0	0	4	0	0	5	0	0	0	0
4	9	0	0	0	1	4	0	0	0	1	6	0	0	9
5	1	0	0	7	0	0	20	0	0	25	1	1	0	1
6	45	0	0	1	8	20	2	0	0	11	32	0	1	46
7	13	0	0	40	1	2	105	**1**	0	132	12	9	0	14
8	237	0	0	13	49	105	28	0	1	89	172	1	10	246
9	117	1	0	221	14	28	563	11	0	718	102	59	1	129
10	1281	0	0	116	280	563	261	1	12	642	939	15	70	1341

This indicates that there is one path from node # 1 to node # 11, i.e., (8,0,0) → (4,4,0), of length 7 edges.

Shortest Paths Originating at Node #1

From	Length	Predecessor
1	**0**	**0**
2	**9**	**14**
3	**10**	**2**
4	**3**	**18**
6	**4**	**4**
7	**2**	**10**
10	**1**	**1**
11	**7**	**23**
14	**8**	**11**
15	**1**	**1**
18	**2**	**15**
22	**5**	**6**
23	**6**	**22**
24	**2**	**10**

From	Length	Predecessor
1	**0**	**0**
2	**9**	**14**
3	**10**	**2**
4	**3**	**18**
6	**4**	**4**
7	**2**	**10**
10	**1**	**1**
11	**7**	**23**
14	**8**	**11**
15	**1**	**1**
18	**2**	**15**
22	**5**	**6**
23	**6**	**22**
24	**2**	**10**

That is, on the path originating at node #1, the predecessor of node #11 is node #23.

(1,4,3) → (4,4,0)

The predecessor of node #23 is node #22

(1,5,2) → (1,4,3)

The predecessor of node #22 is node #6

(6,0,2) → (1,5,2)

etc.

Tracing through the predecessor list, we find that the path from state 1 to state 11 is:

1®15®18®4®6®22®23®11

That is,

8,0,0®3,5,0	Fill #2 from #1
3,5,0®3,2,3	Fill #3 from #1
3,2,3®6,2,0	Empty #3 into #1
6,2,0®6,0,2	Empty #2 into #3
6,0,2®1,5,2	Fill #2 from #1
1,5,2®1,4,3	Fill #3 from #2
1,4,3®4,4,0	Empty #3 into #1