ARTIFICIAL INTELLIGENCE

LAB ASSIGNMENT - 6

The state space for this problem can be described as the set of ordered pairs of integers (x, y)

$x \rightarrow$ quantity of water in 4-gallon jug
$x = 0, 1, 2, 3, 4$

$y \rightarrow$ quantity of water in 3-gallon jug
$y = 0, 1, 2, 3$

Start state $\equiv (0,0)$
Goal state $\equiv (2,0)$

RULE	STATE	PROCESS
1	$(X, Y \mid X < 4)$	$(4, Y)$ Fill 4-gal jug
2	$(X, Y \mid Y < 3)$	$(X, 3)$ Fill 3-gal jug.
3	$(X, Y \mid X > 0)$	$(0, Y)$ Empty 4-gal jug.
4.	$(X, Y \mid Y > 0)$	$(X, 0)$ Empty 3-gal jug.

5	$(x, y \mid x+y >= 4$, $y > 0)$	$(4, y-(4-x))$ Pour water from 3-gal to 4-gal until its full.
6	$(x, y \mid x+y >= 3$, $x > 0)$	$(x-(3-y), 3)$ pour water from 4-gal to 3-gal until its full
7.	$(x, y \mid x+y <= 4$, $y > 0)$	$(x+y, 0)$ Pour water (all) from 3-gal to 4-gal)
8.	$(x, y \mid x+y \le 3$, $x > 0)$	$(0, x+y)$ pour all water from 4-gal to 3-gal.
9	$(0,2)$	$(2,0)$ pour 2-gal water from 3-gal to 4-gal.

INITIALIZATION:

start state $= (0,0)$

Apply Rule 2:

$(x, y \mid y<3) \rightarrow (x,3)$

(Fill 3-gal jug)

Now the state is $(x,3)$

Iteration 1:

current state $\equiv (X,3)$
Apply Rule 7:

$(X,Y \mid X+Y \leq 4$ $\rightarrow$ $(X+Y,0)$
$Y>0)$

Pour all water from 3-gal to 4-gal jug.

Now the state is $(3,0)$.

Iteration 2:

current state $= (3,0)$
Apply Rule 2:

$(X,Y \mid Y<3)$ $\rightarrow$ $(3,3)$

Fill 3-gal jug.

Now the state is $(3,3)$.

Iteration 3:

current state $\equiv (3,3)$
Apply Rule 5:

$(X,Y \mid X+Y \geq 4$ $\rightarrow$ $(4, Y-(4-X))$
$Y>0)$

Pour water from 3-gal to 4-gal until full.

Now the state is $(4,2)$.

classmate
Date
Page

Iteration 4 :

current state = (4,2)
Apply Rule 3 :
(x, y | x > 0) → (0, y)
empty 4-gal jug.

Now the state is (0,2).

Iteration 5 :

current state = (0,2)
Apply Rule 9 :
(0,2) → (2,0)
Pour 2-gal water from 3 to 4-gal jug.

Now the state is (2,0).

GOAL ACHIEVED

Date
Page

Following is my approach for solving the above problem:

1. R'ship was always mutual, which means that if 1 has a friend 2, then 2 has a friend 1.

2. 1 has a friend 2, 2 has a friend 3, doesn't mean that 1 and 3 are friends. Thus, be careful to use the disjoint set.

3. ANALYSIS: seemingly, two-point diagram. But it's simpler than a binary chart.

Open two-tag array, record whether every one has friend in the first group, the second group.

Sweep the person once:

1) If the first group doesn't have his friend: then put it in the first group and mark his friends in the first group with friends.

2) Otherwise:

i) If the 2^{nd} group doesn't have his friend, then put it in the 2^{nd} group and mark his friends in the 2^{nd} group with friends.

2) Otherwise, the description in both groups has no friends.
you can put him casually is a group, also cannot handle it.

3. Analysis: Because there are M representatives of country A and N representatives of country B, and K pairs of representatives were chosen such that for each pair, 1 member is from delegation A & other member is from delegation B.

The problem can be representated bipartite graph 'G' in which vertices are divided into 2 disjoint sets X & Y, such that M representatives of country A set X and N representatives of country B are set Y.

And N+M is the number of vertices G.

Based on graph theory, edge cover no. + the number of edges in a max. matching = N+M.

For the problem, the number of edges in a max. matching is computed first.

Then the edge cover is obtained:

N+M − ans.

It is the minimum number of needed telephone connections.

Scanned with CamScanner