CS565 ARTIFICIAL INTELLIGENCE

HOMEWORK - 2

NAME : MADHAN THANGAVEL
ID : B00819916 (mthanga1)

1) MDP

a) States : { Top, Bottom }

Action : { DRIVE, STOP }

Transition Func : T(s, a, s')

⇒ probablity that action - a taken from state - s leads to state - s'.

ie : $P(s' | s, a)$

Reward Func : +3 (Top), +1 (bottom)

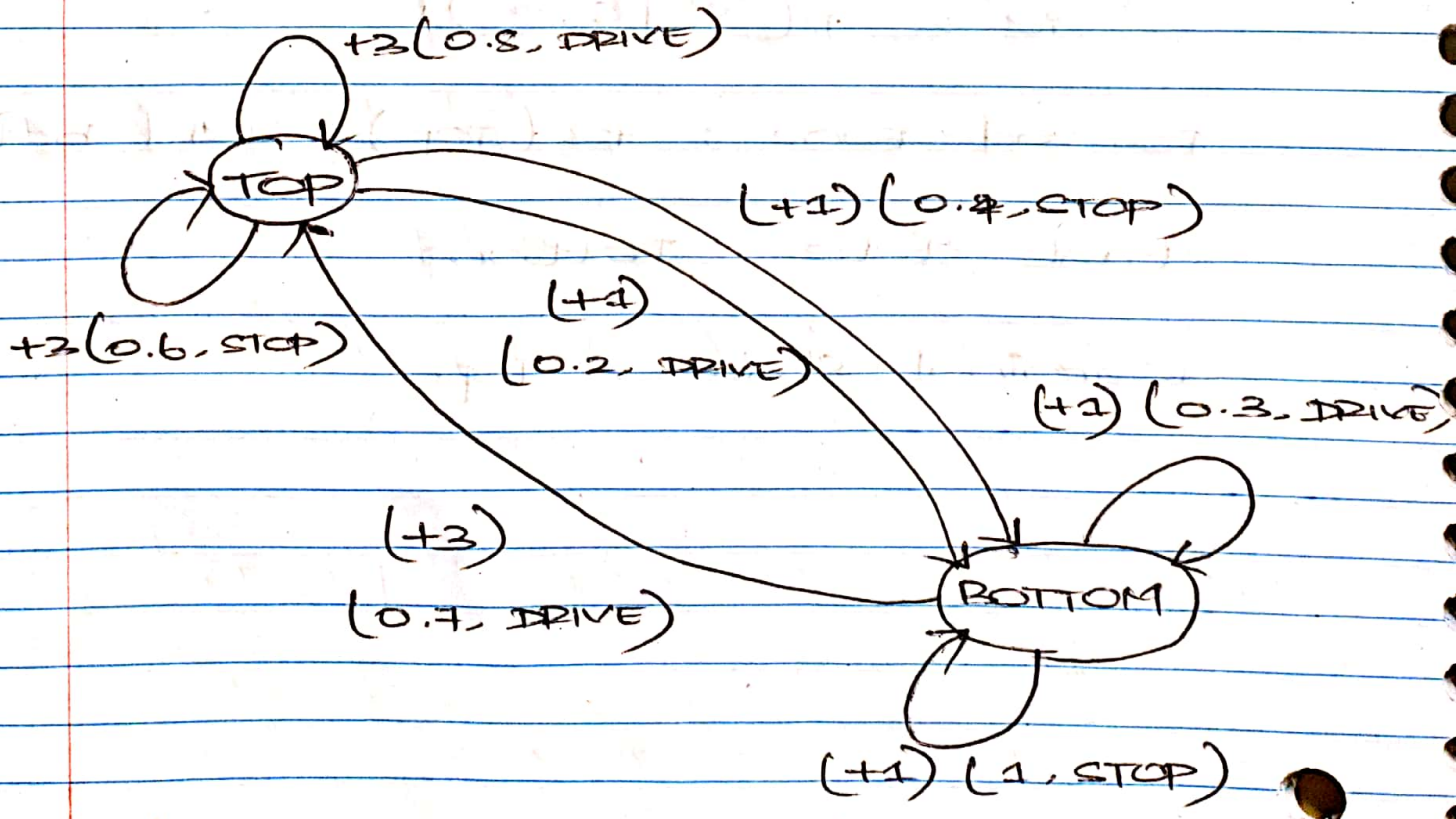
start state : Bottom

Terminal state : Top

(GIVEN) TRANSITION VALUES :

S	a	s'	T(s,a,s')	R(s,a,s')	COST
TOP	DRIVE	TOP	0.8	+3	-1
TOP	DRIVE	BOTTOM	0.2	+1	-1
TOP	STOP	TOP	0.6	+3	
TOP	STOP	BOTTOM	0.4	+1	
BOTTOM	DRIVE	TOP	0.7	+3	-1
BOTTOM	DRIVE	BOTTOM	0.3	+1	-1
BOTTOM	STOP	BOTTOM	1.0	+1	

b) TRANSITION DIAGRAM :



c) VALUE ITERATION :

We know that,

$$V_{k+1}(s) \leftarrow \max_a \sum T(s,a,s') \left[R(s,a,s') + \gamma V_k(s') \right]$$

$\gamma \Rightarrow$ Discount Factor.

So let us compute $V_1(s)$ and $V_2(s)$ for both the states.

$V_1(s)$

* TOP: $V_1(s) = \max [0.8(3-1) + 0.2(1-1), 0.6(3) + 0.4(1)]$

$= \max [1.6, 2.2]$

$V_1 S = 2.2$

* BOTTOM:

$V_1(s) = \max [0.7(3-1) + 0.3(1-1), 1(1)]$

$= \max [1.4, 1]$

$V_1(s) = 1.4$

(b) $V_3(c)$

* TOP :

$$V_3(c) = \max\Big[0.8\big((0.8(3-1)+0.8(2.2)\big) + 0.2\big(0.8(1)\big),\ 0.6\big(3+0.8(2.2)\big) + 0.4\big(1+0.8\big)\Big]$$

$$= \max\big[(0.8(3.76)+0.16),\ (0.6(4.76)+0.32)\big]$$

$$= \max[3.168,\ 3.176]$$

$$= 3.176 \ //$$

* BOTTOM :

$$V_3(c) = \max\Big[0.7\big((3-1)+0.8(1.4)\big) + 0.3\big(1-1+0.8(1.4)\big),\ 1\big(1+0.8(1)\big)\Big]$$

$$= \max\big[(1.4+1.12)+0.3(1.12),\ 1.8\big]$$

$$= \max[5.88,\ 1.8]$$

$$= 5.88 \ //$$

We know that V_0 is zero for both the states.

Therefore

V	TOP	BOTTOM
V_0	0	0
V_1	2.2	1.4
V_2	3.176	5.88

d) We know that,

$$V^{\pi}(s) = \sum_{s'} T(s, \pi(s), s') \left[R(s, \pi(s), s') + \gamma V^{\pi}(s') \right]$$

$\gamma \Rightarrow$ Discount Factor.

As $V_0^{\pi}(s) = 0$

$$V_{k+1}^{\pi}(s) \leftarrow \sum_{s'} T(s, \pi(s), s') \left[R(s, \pi(s), s') + \gamma V_k^{\pi}(s') \right]$$

$V_0^{\pi}(s) = 0$.

$$V_1^{\pi}(s) \leftarrow \sum_{s'} T(s, \pi(s), s') R[(s, \pi(s), s') + \gamma V_0^{\pi}(s')]$$

(TOP)

$= 0.8 [(3-1) + 0.2(0) + 0.6(2) + 0.4]$

$= 1.6 + 1.8 + 0.4$

$= 3.8$.

(BOTTOM)

$= 0.7 [(3-1) + 0.3(0) + 1(1)]$

$= 1.4 + 1$

$= 2.4$.

$$V_2^{\pi}(s) \leftarrow \sum_{s'} T(s, \pi(s), s') R[s, \pi(s), s') + \gamma V_1^{\pi}(s')]$$

(TOP)

$= [0.8(0.8 \times 2) + (0.8)(3.8)]$

$= 1.28 + 3.04$

$= 4.32$.

BOTTOM.

$$= [0.7(2) + 0.8(1.4) + 0.3(0.8) + 1.4(2.8)]$$

$$= [2.52 + 3.04]$$

$$= 5.56.$$

V	TOP	BOTTOM
$V_0^{\pi}(s)$	0	0
$V_1^{\pi}(s)$	3.8	2.4
$V_2^{\pi}(s)$	4.32	5.56

e) POLICY EXTRACTION

We know that

$$Q^*(s,a) = \sum T(s,a,s')[R(s,a,s') + \gamma V^*(s')]$$

$$\pi^*(s) = \arg\max_a Q^*(s,a)$$

Therefore,

$$\pi^*(s) = \arg\max_a \sum T(s,a,s') \left[R(s,a,s') + \gamma V^*(s')\right]$$

Now convert $V_2^*(s)$ into $Q(s,a)$

$$Q[\text{TOP, DRIVE}] = [0.8(2 + 0.8 * 3.176) + 0.2(0.8(5.88)]$$

$$\Rightarrow 5.09$$

$$Q[\text{TOP, STOP}] = [0.6(2 + (0.8)(3.176) + 0.4(1 + 0.8(5.88)]$$

$$= [3.32 + 5.104]$$

$$= 8.42.$$

$$Q[\text{BOTTOM, DRIVE}] =$$

$$[0.7(2 + 0.8(5.88)) + 0.3(0.8(5.88)]$$

$= [4.69 + 1.41]$

$\Rightarrow 6.10$

$Q[\text{BOTTOM}, \text{STOP}] = [1(1 + 0.8(5.88)$

$\Rightarrow 5.70$

$$\pi^*(s) = \arg\max_a \sum_{s'} p(s'|s,a)\, U(s')$$

where

$$U(s^*) = p(s) + \gamma \max_a \sum p(s'/s) \times U(s')$$

so,

$\pi^*(\text{TOP}) = \arg\max [Q[\text{TOP}, \text{DRIVE}], Q[\text{TOP}, \text{STOP}]$

$= \arg\max [5.09, 8.42]$

$= 8.42 \ (\text{stop})$

$\pi^*(\text{BOTTOM}) = \arg\max [Q[\text{BOTTOM}, \text{DRIVE}], Q[\text{BOTTOM}, \text{STOP}]$

$= \arg\max [6.10, 5.70]$

$= 6.10 \ (\text{drive})$

Hence the policy

$\pi^*(s) = \{$ stop, naive $\}$ in this iteration.

2> MAP COLORING

a) code seeds

:

neighbour (y, x) :- adj (x, y)

Block 1 ://

1 {colour (x, c) : colors (c)}

=>

1 :- countries (x).

. Why because, to ensure all the country has exactly one colour with aggregates.

colour (x, red) | colour (x, green)

| colour (x, blue) | colour (x, white)

:- countries (x).

b) 'not' represents two countries cannot have same colors.

If x is the country and c is the colors, then it is unknown that country is having colors. which then it is having a colors.

c) When neighbours rules are removed,

It throws an error/warning stating

"atom does not occurs in any rule head neighbours (x1, x2).

When the action is not performed, then the clingo response by stating atom not present.