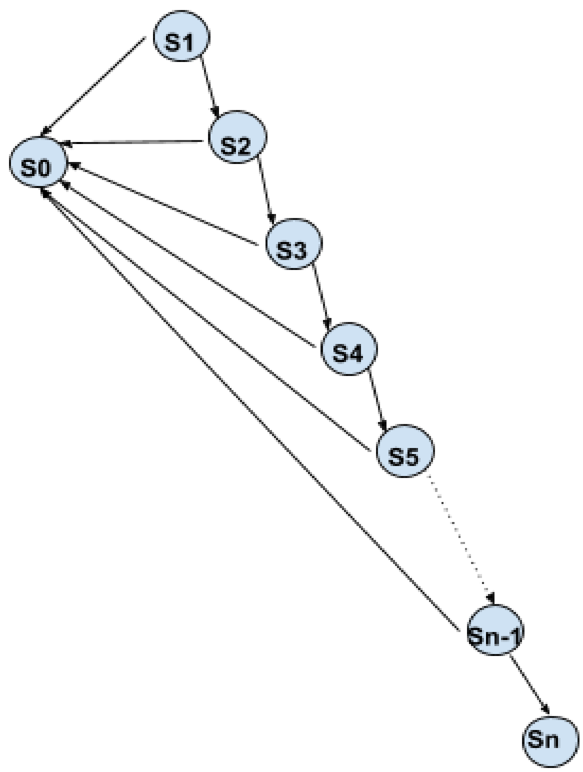
CS 540: Introduction to Artificial Intelligence
Homework Assignment # 11
Assigned: 12/11 Due: 12/18 before class

Name: Sushma Kudlur Nirvanappa **ID: 9079813292**

1. (Search) Consider n + 1 states. S1 is the initial state, Sn is the goal state. S0 is a dead-end state with no successors. For each state Si , i = 1, . . . , n − 1, it has two successors: Si+1 and S0. Sn also has no successors. There is no cycle check nor CLOSED list. How many goal-checks will be performed by depth first search? Assume everything being equal, state with small index is checked first. If a state is goal-checked multiple times, count it multiple times.



At depth 1: (Root at depth 1)
States in CLOSED: S1
Total No of Goal checks: 1

At depth 2:
States in CLOSED: S1, S0, S2
Total No of Goal checks: 3

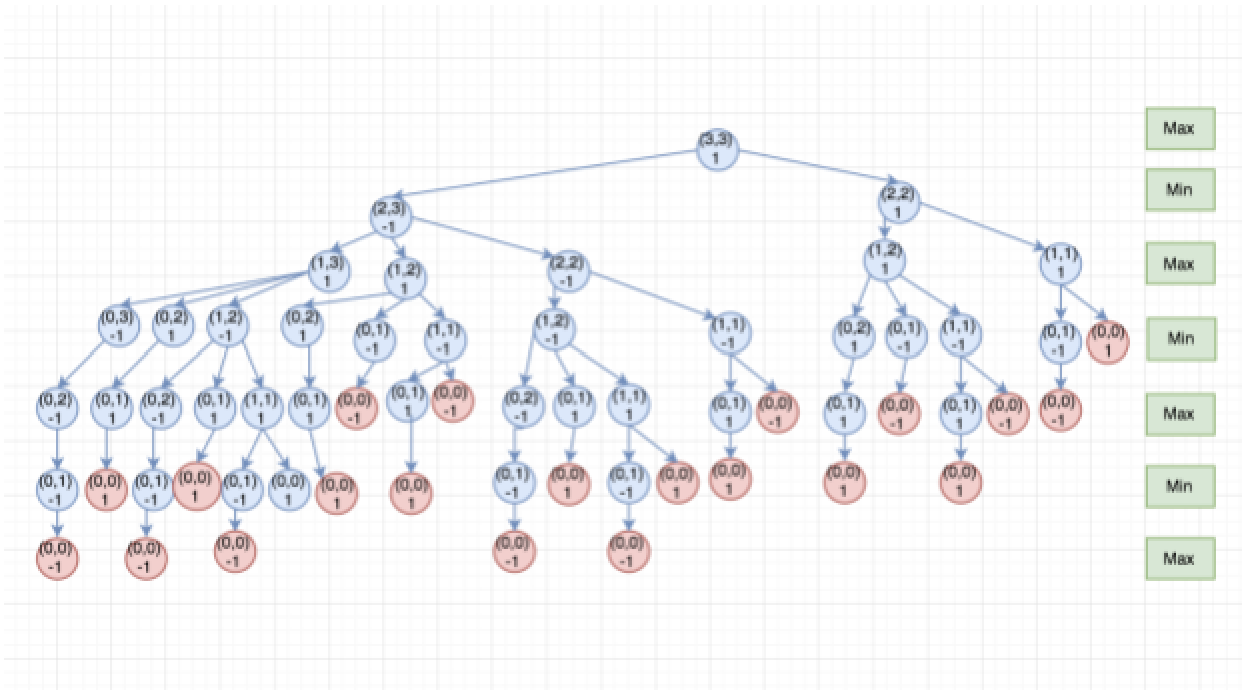
At depth 3:
States in CLOSED: S1, S0, S2, S0, S3
Total No of Goal checks: 5

At depth n:
States in CLOSED: S1, S0, S2, S0, S3,..........S0, Sn-1, S0, Sn
Total No of Goal checks:

Sequence observed: 1,3,5,7 ……..

This is in AP (Arithmetic progression) nth term is given by $T_n = a + (n-1)d$ = 1+(n-1)2 = **2n-1 goal checks will performed**

2. (Game) Consider a variant of the II-nim game. There are two piles, each pile has three sticks. A player can take one stick from a single pile; or take two sticks: one from each pile (when available). The player who takes the last stick wins. Let the game value be 1 if the first player wins. Show the game tree and give the game theoretical value at all nodes.



Game theoretical value is 1.

3. (Probability) There are two biased coins in my wallet: coin A has P(Heads) = a, coin B has P(Heads) = b. I took out one coin at random (with equal probability choosing A or B) and flipped it twice: the outcome was Head, Tail. What is the probability that the coin was A?

Given,

P(A) = P(B) = 0.5

Coin A, P(H) = a, so P(T) = 1-a

Coin B, P(H) = b, so P(T) = 1-b

Find P(A|H,T) = ?

P(A|H,T) = $\frac{P(A,H,T)}{P(H,T)}$

= $\frac{P(H,T|A) * P(A)}{P(H,T|A)*P(A) + P(H,T|B)*P(B)}$ // By chain rule and joint probability

Probability of getting head/tail from coin A/B are independent events

= $\frac{P(H|A)*P(T|A)*P(A)}{P(H|A)*P(T|A)*P(A) + P(H|B)*P(T|B)*P(B)}$

= $\frac{a * (1-a) * 0.5}{a * (1-a) * 0.5 + b * (1-b) * 0.5}$ = $\frac{0.5 * a * (1-a)}{0.5 * [a * (1-a) + b * (1-b)]}$

= $\frac{a(1-a)}{a(1-a) + b(1-b)}$

4. (PCA) You performed PCA in R2 . It turns out that the first principal component is u1 = (√ 1 2 , √ 1 2), and the second principal component is u2 = (− √ 1 2 , √ 1 2). One of your data points has its new representation as (1, 2). What was the original coordinates of the point?

New data point = (1, 2)

first principal component, $u_1 = (\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}})^T$

second principal component, $u_2 = (-\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}})^T$

Let's assume that the original point is **X = (a, b)** , which we have to find here.

Projected data point (1, 2) is given by (u_1^Tx, u_2^Tx)

So we have:

$u_1^Tx = 1$

$u_2^Tx = 2$

$\frac{a}{\sqrt{2}} + \frac{b}{\sqrt{2}} = 1$ ………. (1)

$\frac{-a}{\sqrt{2}} + \frac{b}{\sqrt{2}} = 2$ ………. (2)

By Adding (1) and (2):

=> $\frac{b}{\sqrt{2}} + \frac{b}{\sqrt{2}} = 3$

=> $\frac{2b}{\sqrt{2}} = 3$

=> $b = \frac{3}{\sqrt{2}}$

By subtracting (2) from (1):

=> $\frac{a}{\sqrt{2}} + \frac{a}{\sqrt{2}} = -1$

=> $\frac{2a}{\sqrt{2}} = -1$

=> $a = \frac{-1}{\sqrt{2}}$

Thus the original point is **X (a,b) = ($\frac{-1}{\sqrt{2}}$, $\frac{3}{\sqrt{2}}$)**

5. (Resolution) Given the knowledge base (a) A ∨ B (b) C ⇒ A use resolution to prove the query A ∨ B ∨C.

Given KB:

(a) A ∨ B

(b) C ⇒ A, in CNF form = ~C ∨A

Query:

(a) A ∨ B ∨ C

To prove by resolution we take **KB ∧~Query**

~Query = ~(A ∨ B ∨ C) = ~A ∧~B ∧ ~C

For resolution we have:

KB terms:

1. A ∨ B
2. ~C ∨ A

~Query:

3. ~A
4. ~B
5. ~C

Now, using (1), (3) and (4) we get empty result

(A ∨ B) ∧(~A) ∧(~B) = A ∧~A ∨ B ∧~B = Empty, thus resolution is done with the given KB and query.

6. (Clustering) There are six points in two-dimensional space: a = (0, 0), b = (1, 0), c = (3, 1), d = (7, 7), e = (9, 9), f = (3, 6). Perform Hierarchical Agglomerative Clustering with single linkage and Euclidean distance. Complete the resulting clustering tree diagram (i.e., the dendrogram).

Note: Used simple linkage Euclidean distance to find distance between clusters.

Iteration 1:

	a(0,0)	b(1,0)	c(3,1)	d(7,7)	e(9,9)	f(3,6)
a(0,0)		1	3.162278	9.899495	12.727922	6.708204
b(1,0)			2.236068	9.219544	12.041595	6.324555
c(3,1)				7.211103	10	5
d(7,7)					2.828427	4.123106
e(9,9)						6.708204
f(3,6)						

So, a(0,0) and b(1,0) are close to each other and they merge to form a cluster :
(a,b), c, d, e, f

Iteration 2:

	a(0,0)+b(1,0)	c(3,1)	d(7,7)	e(9,9)	f(3,6)
a(0,0)+b(1,0)		2.236068	9.219544	12.041595	6.324555
c(3,1)			7.211103	10	5
d(7,7)				2.828427	4.123106
e(9,9)					6.708204
f(3,6)					

So, cluster (a(0,0),b(1,0)) and c(3,1) are close to each other and they merge to form a new cluster : **((a,b), c), d, e, f**

Iteration 3:

	a(0,0)+b(1,0)+c(3,1)	d(7,7)	e(9,9)	f(3,6)
a(0,0)+b(1,0)+c(3,1)		7.211103	10	5
d(7,7)			2.828427	4.123106
e(9,9)				6.708204
f(3,6)				

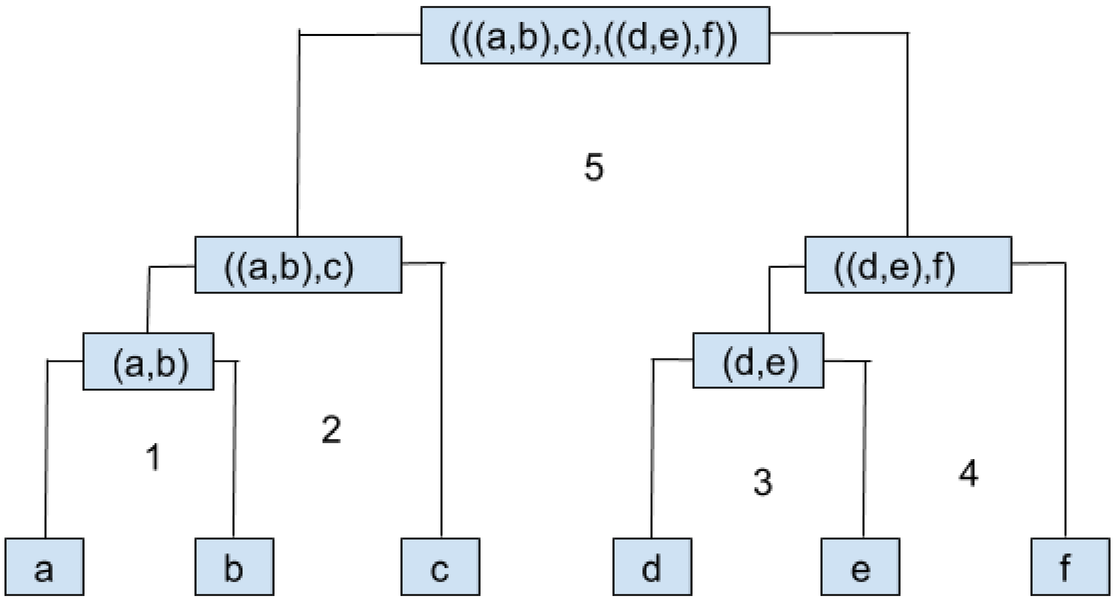
So, d(7,7) and e(9,9) are close to each other and they merge to form a cluster :
((a,b), c), (d, e), f

Iteration 4:

	a(0,0)+b(1,0)+c(3,1)	d(7,7)+e(9,9)	f(3,6)
a(0,0)+b(1,0)+c(3,1)		7.211103	5
d(7,7)+e(9,9)			4.123106
f(3,6)			

So, cluster (d(7,7),e(9,9)) and f(3,6) are close to each other and they merge to form a new cluster : **((a,b), c), ((d, e), f)**

Iteration 5:
So in this iteration the two clusters **((a,b), c) and ((d, e), f)** merge to form one cluster **(((a,b), c), ((d, e), f))**



7. (Gradient descent) Let x = (x1, x2) ∈ R 2 . We want to minimize the objective function f(x) = sin((x1 + x2)π) + e x1−x2 + x1x2. Let the stepsize η = 0.1. If we start at x (0) = (1, 1), what is the next vector x (1) produced by gradient descent?

Let x = (x_1 , x_2) ∈ R2

Given function f(x) = $sin((x_1 + x_2)\pi) + e^{x_1 - x_2} + x_1x_2$

stepsize η = 0.1

$x^{(0)}$ = (1, 1)

$x^{(1)}$ = ?

From gradient descent, we know

$x^{(1)}$ = $x^{(0)}$ - η∇f($x^{(0)}$)

∇f(x) = ∇($sin(x_1 + x_2)\pi) + e^{x_1 - x_2} + x_1x_2$)

= ($\partial(sin((x_1 + x_2)\pi) + e^{x_1 - x2} + x_1x_2)/\partial x_1,\ \partial(sin((x_1 + x_2)\pi) + e^{x_1 - x_2} + x_1x_2)/\partial x_2)$

= $(\pi.cos((x_1 + x_2)\pi) + e^{x_1 - x2} + x_2,\ \pi.cos((x_1 + x_2)\pi) - e^{x_1 - x_2} + x_1)$

now, ∇f($x^{(0)}$) = $(\pi.cos((1 + 1)\pi) + e^{1-1} + 1,\ \pi.cos((1 + 1)\pi) - e^{1-1} + 1)$

= $(\pi.cos(2\pi) + e^0 + 1,\ \pi.cos(2\pi) - e^0 + 1)$

= $(\pi + 1 + 1,\ \pi - 1 + 1)$

= $(\pi + 2,\ \pi)$

Now substituting values in gradient descent equation, $x^{(1)}$ = $x^{(0)}$ - η∇f($x^{(0)}$)

$x^{(1)}$ = (1,1) - 0.1*∇f($x^{(0)}$)

= (1,1) - 0.1* $(\pi + 2,\ \pi)$

= (1,1) - $(0.1 * \pi + 0.2,\ 0.1 * \pi)$

= (0.8 - 0.1* π ,1 - 0.1* π)

$x^{(1)}$ **= (0.8 - 0.1* π ,1 - 0.1* π)**

8. (Sigmoid) Derive the derivative of the sigmoid function σ(x) = 1 / 1+exp(−x) .

Sigmoid function σ(x) = $\frac{1}{1+exp(-x)}$

Derivative of the sigmoid function is given by, σ'(x) = $\frac{d}{dx}\frac{1}{1+exp(-x)}$

Using the quotient and chain rule, we can differentiate above.

$$\frac{d}{dx}\frac{f(x)}{g(x)} = \frac{g(x)\,f'(x) - f(x)\,g'(x)}{(g(x))^2}$$

σ'(x) = $\frac{(1+exp(-x))\,(\frac{d}{dx}1) - 1(\frac{d}{dx}(1+exp(-x))\,)}{(1+exp(-x))^2}$

σ'(x) = $\frac{0 - 1(0+ exp(-x)(-1))}{(1+exp(-x))^2}$

σ'(x) = $\frac{exp(-x)}{(1+exp(-x))^2}$ = $\frac{1 + exp(-x) - 1}{(1+exp(-x))^2}$

σ'(x) = $\frac{1 + exp(-x)}{(1+exp(-x))^2} - \frac{1}{(1+exp(-x))^2}$

σ'(x) = $\frac{1}{1+exp(-x)} - (\frac{1}{1+exp(-x)})^2$

σ'(x) = $\sigma(x) - \sigma(x)^2$

σ'(x) = $\sigma(x)(1 - \sigma(x))$

9. (MDP) Consider state space S = {s1, . . . , sn} and action space A = {lef t, right}:
s1 . . . sn
The actions move the agent one step in the corresponding direction, except when it is at an end: attempting to move beyond the end makes the agent stay in the current state.

When the agent is in state sn−1, taking the "right" action also gives it reward r = 1. All other state-action pairs have zero reward. Let γ be the discounting factor. What is the value v(s1) under the optimal policy?

The optimal policy for our case is
$\pi(s_1)$ = right
$\pi(s_2)$ = right
$\pi(s_3)$ = right
...
...
...
…
$\pi(s_{n-1})$ = right
$\pi(s_n)$ = left

Now,
$v(s_1) = \gamma^0.R0 + \gamma^1.R1 + \gamma^2.R2 + \gamma^3.R3 + ……$
$v(s_1) = \gamma^0.0 + \gamma^1.0 + \gamma^2.0 + \gamma^3.0 + …… \gamma^{n-3}.0 + \gamma^{n-2}.1 + \gamma^{n-1}.0 + \gamma^n.1 + \gamma^{n+1}.0 + \gamma^{n+2}.1 + ….$
$v(s_1) = \gamma^{n-2}.1 + \gamma^n.1 + \gamma^{n+2}.1 + ….$
$\mathbf{v(s_1)} = \dfrac{\gamma^{n-2}}{1-\gamma^2}$

10. (Q-learning) A robot initializes Q-learning by setting q(s, a) = 1 for all state s and action a. It has a learning rate α, and discounting factor γ. The robot senses that it is in state s1 and decides to performs action a1. For this action, the robot receives reward 100 and arrives at state s2. After this one step of Q-learning, for all s, a pairs show their value q(s, a).

Given that: $q(s, a) = 1 \ \forall \ s,a$
Learning rate = α
Discounting factor = γ

s1, a1 -> s2
R(s1, a1) = 100

From bellman's equation,

$q'(s1, a1) = \alpha[R + \gamma.\max(q(s', a'))] + (1 - \alpha).q(s1,a1)$
$q'(s1, a1) = \alpha[100 + \gamma.\max(q(s2, a'))] + (1 - \alpha).1$

now, $q(s2, a') = 1 \ \forall \ a$
$\Rightarrow \max(q(s2, a')) = 1$

$\Rightarrow q'(s1, a1) = \boldsymbol{\alpha}[100 + \boldsymbol{\gamma}.1] + (1 - \boldsymbol{\alpha}).1$
$q'(s1, a1) = \boldsymbol{\alpha}[100 + \boldsymbol{\gamma}] + (1 - \boldsymbol{\alpha})$
$q'(s1, a1) = \boldsymbol{\alpha}[99 + \boldsymbol{\gamma}] + 1$

Rest of the Q-table remains intact.
Thus, the updated values will be

$q(s, a) = 1 \;\forall\; s,a - \{s1, A1\}$
$q(s1, a1) = \boldsymbol{\alpha}[99 + \boldsymbol{\gamma}] + 1$