Please add your answers under the question number. If there is already an answer, check it. If you agree with it, add a :[1]) underneath to verify it. Lots of :) means it's a good answer! If you have questions or disagree with an answer, pop a comment! If you want to add notes or explanations of problems, create a new document in this folder and link to it below the relevant question.

- Shevon
- Akira
- Jin
- Callum
- Rowan
- Tom
- Asil

Question 1

a) 1 ☺ ☺ ☺
b) 3 ☺ ☺ ☺ ☺
c) 4 ☺ ☺ ☺ ☺
d) 3 ☺ ☺ ☺ ☺
e) 1 ☺ ☺ ☺ ☺

Question 2

a) A higher learning rate means that the algorithm 'changes its mind more quickly', i.e. new values have a greater effect. When the learning rate is high, the difference between the evaluation (z) and true utility (t) of s has a greater impact on the weights.
    - η determines how much we wish to change/update our weight parameters each iteration
    - Too high of an η may overshoot the optimal minimum whilst too small of an η may never converge due to time complexity

b) Two problems with gradient descent learning:
    i) **Delayed reinforcement**: it takes some time to know whether the move was useful or not. ☺ ☺ ☺
    ii) **Credit assignment**: it can be difficult to know which state in a sequence was responsible for the final utility (t). For example, a player might make a great move followed by a terrible move, then another great one. It is difficult for the agent to learn which moves were great and which were terrible, because the final utility will only reflect the game overall. ☺ ☺ ☺

c) 3 Environments:
    i) TD ☺ ☺
        - We want multi-step predictions in chess, and we are also playing against a skilled opponent
    ii) Supervised learning ☺ ☺

[1] 信じたのものは、都合のいい妄想を繰り返し映し出す鏡。 #Lyricova

- We have labels and since backgammon is stochastic, there is no need for a multi-step prediction so a single-step prediction works

iii) TD ☺
   - We are self playing over several simulations hence TD can be used to converge to optimal weights
   - *"Play against yourself" refers to the algorithm playing against itself. [Ref: Matt]*

Question 3

a) 6 ☺ ☺ ☺
~~b)~~ 8, 18, 26-31 :) ☺ ☺ ☺ ☺
c) 14. *Assuming "reordering" means swapping nodes within a branch. (otherwise it might not make sense to a game tree)*
   Sequence: 6 5 9 4 8 2 5 1 7 6 6 3 4 1 9 0 6 5[2], Tested on a simulator. ☺ ☺
d) 3 main components that define an auction mechanism:
   - A **language** to describe the allowable strategies that an agent can follow ☺ ☺ ☺
   - A **protocol** for communicating bids from bidders to the auctioneer ☺ ☺ ☺
   - An **outcome rule**, used by the auctioneer to determine the outcome ☺ ☺ ☺

Question 4

a)

Node	Goal	Fringe
-	-	S: g=0, f=16
S	No	S->A: g=5, f=7 S->B: g=6, f=20
A	No	S->B: g=6, f=20 A->G1: g=15, f=15 ☺ ☺ ☺
G1	Yes	S->B: g=6, f=20 (but who cares at this point amirite ☺ ☺ ☺)

** Given in question*

b)
i) **Regular A*** ☺ ☺ ☺
ii) When ε = 0, f(n) = g(n) ⇒ **Uniform-cost search** ☺ ☺ ☺ ☺
iii) ~~When ε → ∞, f(n) → ∞ ⇒ all nodes have same the same f(n), so Breadth-first search?~~
Actually should be **BFS** since all nodes are infinite, hence all equal cost so BFS (assume all nodes have infinite cost and not arbitrary large values that are not equal but are close to infinite)

[2] Sequence generated by placing the child node chosen to the leftmost slot for each subtree.

This question is worded badly, but the intent might actually be that it's Greedy search? As it makes g(x) insignificant, only h(x) will be used ←-I was thinking the same
I think it becomes greedy search. You can't equate 2 infinities to be the same. (am2 throwback) ☺

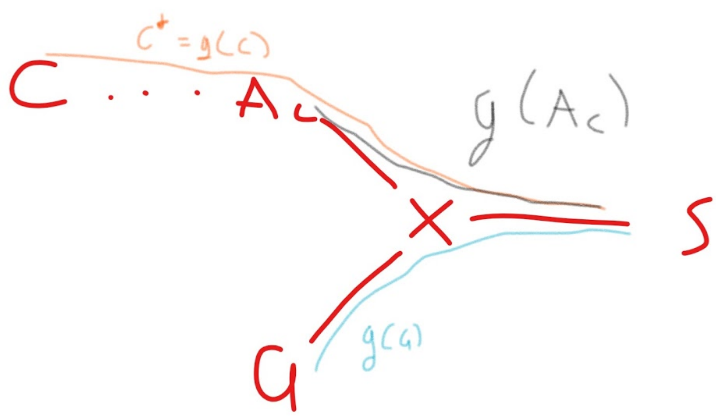
c)
I think it's i) but I'm not sure why

(Please critique, but I hope it's clear enough) :))))) _blank_
I don't think it's (ii) or (iii), just look at Q4a; G1 had cost 15 but G3 (the optimal) had cost 10, and as $\varepsilon = 2$ then clearly $15 < 10 + 2$ and $15 < 10 + 2 \times 2$ are both wrong.

I say it's **(i)**, which is stronger than (iv) by a longshot.

INTUITION FOR PROOF: If this was an optimal algorithm, then even if the search got one node away from expanding ANY suboptimal goal node, that evaluation would always be more expensive than the options around it. From this intuition I propose the following.

I assume two things: first, that the graph is acyclic, and second, that no path cost is negative (which I think are reasonable).
Let S be the root node, X be some generic location adjacent to G (the suboptimal goal), C the optimal goal, and Ac some generic node that 'leads to' C. (Afterthought: Ac doesn't necessarily have to be adjacent to X, it just needs to be somewhere on the way to C). So the structure of the graph is something like (apologies for the quality)



Now as G, C are goal states and the heuristic h is admissible,
$f(G) = g(G)$ and $f(C) = C^*$ ... (1)
Now $f(Ac) = g(Ac) + \varepsilon \times h(Ac)$
But h is admissible and so as the 'true cost' of reaching C from Ac is $C^* - g(Ac)$,
$f(Ac) < g(Ac) + \varepsilon \times (C^* - g(Ac))$ ... (2)

Now G is chosen over Ac because Weighted A* has suboptimally chosen G, which means that
$f(G) < f(Ac)$
Using (1) and (2) this means
$f(G) = g(G) < f(Ac) < g(Ac) + \varepsilon \times C^* - \varepsilon \times g(Ac)$
I.e. $g(G) < \varepsilon \times C^* + g(Ac) * (1-\varepsilon)$
But $\varepsilon >= 1$ so $1 - \varepsilon \leq 0$, and as $g(Ac)$ is non-negative by assumption we get
$g(G) \leq \varepsilon \times C^*$ which is (i). //

Question 5

a) 1 ☺ ☺
b) {1, 2, 3, 4} ☺ ☺
c) {1, 2, 3} ☺ ☺
d) {1} ☺ ☺
e) 8
   1. $O(d)$ best case (minimum number - for worst case we have $O(nd^2)$) ☺ (Why $O(d)$ best case?)
      We're using AC-3, not the tree search algorithm. AC-3 is $O(n^2d^3)$, which has d^2 per arc, so the answer is actually $O(n^2d)$ arcs. (Via Matt)

Question 6

a) No. A->G->J
b) Answer should be 441/1431 = 49/159 (was confirmed with a few others, I'll write it up tonight after I get q7 confirmed) :)

$$
\begin{aligned}
P(g|\neg j, a, c) &= \frac{P(g, \neg j, a, c)}{P(\neg j, a, c)} \\
&= \frac{\sum_v P(g, \neg j, a, c, v)}{\sum_g \sum_v P(g, \neg j, a, c, v)} \\
&= \frac{\sum_v P(\neg j|g) P(g|c, a, \underline{v}) P(a|c, \underline{v}) P(c) P(\underline{v})}{\sum_g \sum_v P(\neg j|\underline{g}) P(\underline{g}|c, a, \underline{v}) P(a|c, \underline{v}) P(c) P(\underline{v})} \quad (\textit{underlining sum variables}) \\
&= \frac{\cancel{P(c)} P(\neg j|g) \sum_v P(g|c, a, \underline{v}) P(a|c, \underline{v}) P(\underline{v})}{\cancel{P(c)} \sum_g P(\neg j|\underline{g}) \sum_v P(\underline{g}|c, a, \underline{v}) P(a|c, \underline{v}) P(\underline{v})} \\
&= \frac{0.1 \times (0.9 \times 0.9 \times 0.1 + 0.8 \times 0.5 \times 0.9)}{0.1 \times (0.9 \times 0.9 \times 0.1 + 0.8 \times 0.5 \times 0.9) + 1.0 \times (0.1 \times 0.9 \times 0.1 + 0.2 \times 0.5 \times 0.9)} \\
&= \frac{441}{1431} \\
&= \frac{49}{159}
\end{aligned}
$$

$$
\begin{aligned}
P(g \mid \neg j, a, c) &= \alpha \,.\, P(g, \neg j, a, c) \\
&= \alpha \,.\, \Sigma_v\, P(g, \neg j, a, c, V) \\
&= \alpha \,.\, \Sigma_v\, P(g \mid c, a, v) \,.\, P(\neg j, g) \,.\, P(a \mid c, v) \,.\, P(c) \,.\, P(V) \\
&= \alpha \,.\, P(c) \,.\, P(\neg j, g) \,.\, \Sigma_v\, P(g \mid c, a, v) \,.\, P(a \mid c, v) \,.\, P(V) \\
&= \alpha \times 0.9 \times 0.1 \times [\, (0.9 \times \cancel{0.1}\ 0.9 \times 0.1) + (0.8 \times 0.5 \times 0.9)\,] \\
&= 0.03969\, \alpha
\end{aligned}
$$

$$
\textit{where} \quad \alpha = \frac{1}{P(\neg j, a, c)}
$$

$$
= \frac{1}{\sum_v \sum_g P(G, \neg j, a, c, V)}
$$

$$
= \frac{1}{\sum_v \sum_g P(g \mid c, a, v) \,.\, P(\neg j, g) \,.\, P(a \mid c, v) \,.\, P(c) \,.\, P(V)}
$$

$$
= \frac{1}{P(c) \,.\, \sum_v P(a \mid c, v) \,.\, P(V) \sum_g P(g \mid c, a, v) \,.\, P(\neg j, g)}
$$

- $P(g|\neg j, a, c) = \frac{441}{1431} = \frac{49}{159}$ ☺ ☺

Question 7

Define h as hypothesis, m as measurements, $P_i(h_j)_t$ denotes the probability of hypothesis on zone j at time t in iteration i.

$P(M\|H)$	h_1	h_2	h_3
m_1	0.5	0.25	0.25
m_2	0.25	0.5	0.25
m_3	0.25	0.25	0.5

$$P(h_1) = P(h_2) = P(h_3) = P_1(m_1) = P_1(m_2) = P_1(m_3) = \frac{1}{3}$$

$$P_1(m_1) = P(m_1|h_1)P(h_3) + \cdots + P(m_1|h_3)P(h_3) = \frac{1}{6} + \frac{1}{12} + \frac{1}{12} = \frac{1}{3}$$

i)

Solving for probability of $h_{t=1} = 1, h_{t=2} = 2$ given $m_{t=1} = 1,\ m_{t=2} = 1$:

$$\begin{aligned}
P_1(h_1)_1 &= \frac{P(m_1|h_1)P(h_1)_1}{P(m_1)} \\
&= \frac{\frac{1}{2}\times\frac{1}{3}}{\frac{1}{3}} \\
&= \frac{1}{2} \\
P_1(h_2)_1 &= \frac{1}{4} \\
P_1(h_3)_1 &= \frac{1}{4}
\end{aligned}$$

$$P_1(h_2)_2 = P_1(h_1)_1 = \frac{1}{2};\ P_1(h_3)_2 = \frac{1}{4}; P_1(h_1)_2 = \frac{1}{4};$$

$$\begin{aligned}
P_2(h_2)_2 &= \frac{P(m_1|h_2)P_1(h_2)_2}{P_1(m_1)_2} \\
&= \frac{\frac{1}{4}\times\frac{1}{2}}{\frac{1}{4}\times\frac{1}{2}+\frac{1}{4}\times\frac{1}{2}+\frac{1}{4}\times\frac{1}{4}} \\
&= \frac{2}{5}
\end{aligned}$$

(ii)

Solving for probability of $h_{t=1} = 2, h_{t=2} = 3$ given $m_{t=1} = 1,\ m_{t=2} = 1$:

$$\begin{aligned}
P_2(h_3)_2 &= \frac{P(m_1|h_3)P_1(h_3)_2}{P_1(m_1)_2} \\
&= \frac{\frac{1}{4}\times\frac{1}{4}}{\frac{1}{4}\times\frac{1}{2}+\frac{1}{4}\times\frac{1}{2}+\frac{1}{4}\times\frac{1}{4}} \\
&= \frac{1}{5}
\end{aligned}$$

☺

A sneaky way to do (ii) might be to note that after two consecutive zone 1 reports, we are equally likely to have gone from zone 1 to 2 or 3 to 1 (as both involve a correct and incorrect report). This leaves us with $1 - \frac{2}{5} - \frac{2}{5} = \frac{1}{5}$ for the probability of zone 2 to 3 (which is what this part is looking for).