1. [10%] General AI Knowledge and Applications

True or False [6%]: For each of the statements below, fill in the bubble T if the statement is always and unconditionally true, or fill in the bubble F if it is always false, sometimes false, or just does not make sense.

	T	F
1	T	●
2	●	F
3	T	●
4	T	●
5	●	F
6	●	F

1- Mutually exclusive events are independent.

2- A node X in a Bayesian network is independent of its non-descendants given its parents.

3- A hypothesis that fits the training data better will also generalize better.

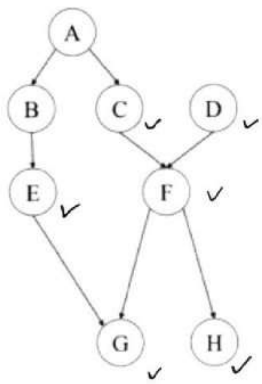
4- Entropy of an unfair coin flip is always less than 1 bit.

5- Cross-validation helps avoid overfitting.

6- Support vectors are data points that lie closest to the separation boundary.

Multiple Choice [4%]: Each question has <u>one or more</u> correct choices. Check the boxes of all correct choices and leave the boxes of wrong choices blank. Please note that there will be no partial credit and you will receive full credit if and only if you choose all the correct choices and none of the wrong choices.

7- [1%] What is the size of the Markov blanket of the node F in the following network?



- [ ] 2
- [ ] 4
- [x] 5
- [ ] 6

8- [1%] Consider the network from the previous question again. If C is a random variable with 3 possible values and D is a random variable with 2 possible values, how large (how many cells) is the conditional probability table (CPT) of the variable F?

3 × 2

- [x] 6
- [ ] 12
- [ ] 8
- [ ] Cannot be determined with the given information.

9- [1%] Which of the following machine learning algorithms is/are parametric?

- [ ] Naïve Bayes
- [x] Neural Networks
- [ ] K-nearest Neighbors
- [x] Linear Regression

10- [1%] Which of the following is/are exact inference method(s) for Bayesian networks?

- [x] Enumeration
- [ ] Gibbs Sampling
- [x] Variable Elimination
- [ ] Rejection Sampling

2. [10%] FOL Resolution

Consider the following logical sentence in CNF form. Show that this sentence is unsatisfiable.

$$(\neg A \vee \neg B \vee C) \wedge (\neg A \vee B) \wedge (A) \wedge (\neg A \vee \neg B \vee \neg C)$$

We use resolution to show:

$\neg A \vee \neg B \vee C$ $\quad$ $\neg A \vee B$ ~~A~~

$\neg A \vee C$ $\quad$ A

C $\quad$ $\neg A \vee \neg B \vee \neg C$

$\neg A \vee \neg B$ $\quad$ A

$\neg B$ $\quad$ $\neg A \vee B$

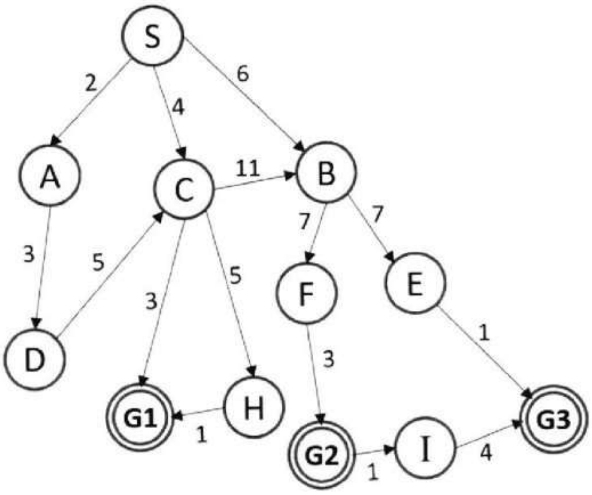
$\neg A$ $\quad$ A

$\{\ \}$

So the original sentence is unsatisfiable.

3. [10%] Search

Consider the following search space, where S is the start state and the goal test is satisfied by each of the three nodes **G1, G2, and G3**. Arcs are labeled with the cost of traversing them.
For each of the following search strategies, indicate which goal state is reached (if any) and list, in order, the nodes in the solution path, and the cost of the solution path to reach the goal state from S. Once you reach any one of the three goals, please **STOP**. When all else is equal, nodes should be removed from OPEN in **alphabetical order**.



a. [3%] Breadth First Search (BFS)

Goal state reached: G1 [1%] Solution path: S - C - G1 [1%] Path cost: 7 [1%]

b. [3%] Depth First Search (DFS)

Goal state reached: G3 [1%] Solution path: S-A-D-C-B-E-G3 [1%] Path cost: 29 [1%]

c. [4%] Uniform Cost Search (UCS)

Goal state reached: G1 [1%] Solution path: S-C-G1 [2%] Path cost: 7 [1%]

4. [10%] Neural Network

a. [3%] Although there is no universal rule, several approaches are available for selecting the number of hidden layers to obtain an optimal neural network architecture. "Genetic algorithms" and "Pruning and weight decay" are two examples mentioned in the class. Explain how **Genetic Algorithms** could be used to determine the number of hidden units.

We can first select ~~a~~ some random numbers (such as 3 or 4) to be the number of hidden ~~layers~~ units and train the network to get appropriate weights. Then every time we randomly select 2 solutions we get and combine their structures (maybe delete or maintain some of the units from these 2 solutions) to get a new solution and train the new ~~new~~ network to get the weights. ~~After~~ When we get new solutions we'll compare all the solutions and delete some of them which are not good. After some time we'll finally choose a satisfying solution to be the neural network we want and decide the final answer of number of hidden units.

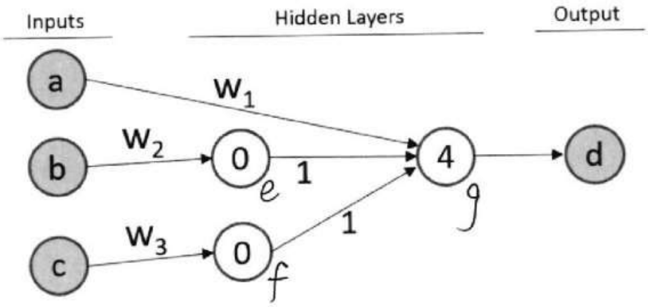
No mention of 'metric that combines network's performance and complexity'.

Q6 Score: **3.0**

b. [**7%**] Consider the following three neural networks, where the threshold values for perceptrons (artificial neurons) are shown inside each circle. For a perceptron with threshold value of Y its firing is based on:

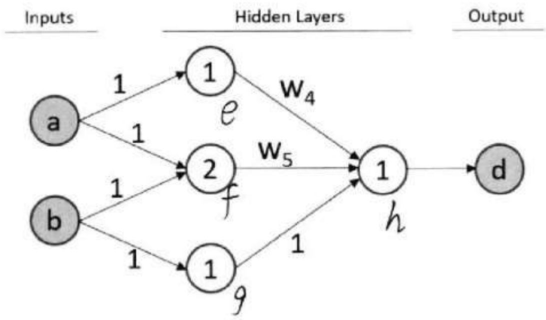
$$f(x) = \begin{cases} 1, & \text{if } x \geq Y \\ 0, & \text{Otherwise} \end{cases}$$

Please assign an integer number to each unspecified weight such that each neural network mimics the intended logic function (inputs are 0 or 1). Write your answers in the provided space.



Desired function: $d = a \wedge \neg b \wedge \neg c$

$W_1 = 2$. $W_2 = -1$. $W_3 = -1$

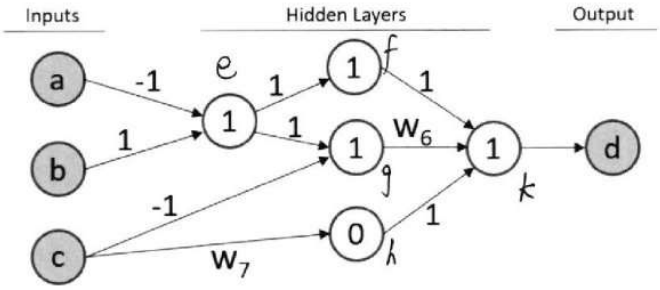


Desired function: $d = a \oplus b$
where $\oplus$ represents exclusive-OR (XOR)

$W_4 = 1$. $W_5 = -2$.

Set the hidden units to be e.f.g.h (as the picture shows)
We get the truth table in the right:

a	b	e	f	g	h	d
0	0	0	0	0	0	0
0	1	0	0	1	1	1
1	0	1	0	0	1	1
1	1	1	1	1	0	0



Desired function: $d = (\neg a \wedge b) \oplus \neg c$
where $\oplus$ represents exclusive-OR (XOR)

~~$W_6 = -2$~~, $W_7 = -1$

$W_6 = -2$

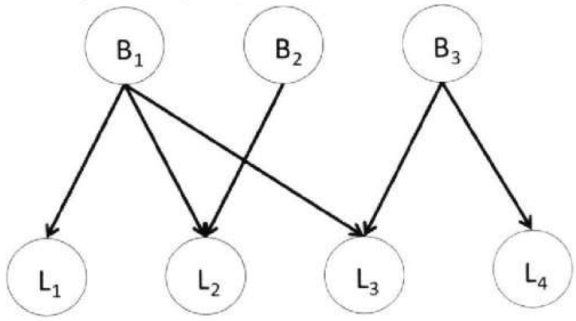
As the ~~e, a, b's~~ weights among e, a, b are ~~As e - a, b it's~~ already set in the given network and ~~st~~ satisfy $e = \neg a \wedge b$

So we only consider the truth table of e, c, f, g, h, k, d.

~~a b~~	c	e	f	g	h	k	d
~~0 0~~	0	0	0	0	~~0~~ 1	~~0~~ 1	~~0~~ 1
	0	1	1	1	~~0~~ 1	0	0
	1	0	0	0	1	0	0
	1	1	1	0	0	1	1

5. [15%] Bayesian Network

We have a strange unreliable light system with 4 lights L_1, L_2, L_3 and L_4 and 3 buttons B_1, B_2 and B_3. Once we press some buttons, some lights may be turned on with some probability. Since it is not a deterministic system, sometimes nothing happens after pressing the buttons. We use $P(B_i)$ to represent the probability of pushing button Bi and $P(L_j)$ to represent the probability of light Li turning on.



The Bayesian network of this light system is provided above. Every arrow means the button has some potential probability to activate the corresponding light. If no arrow exists between a button and a light, then the button has nothing to do with that light.

a. [3%] Expand the joint probability $P(L_1, L_2, L_3, L_4, B_1, B_2, B_3)$ with a product of conditional probabilities.

$P(L_1, L_2, L_3, L_4, B_1, B_2, B_3)$

$\cancel{= P(B_1) \cdot P(B_2) \cdot P(B_3) \cdot P(L_1|B_1) \cdot P(L_2|B_1) \cdot P(L_3|B_1) \cdot P(L_2|B_2) \cdot P(L_3|B_3)}$

$= P(B_1) \cdot P(B_2) \cdot P(B_3) \cdot P(L_1|B_1) \cdot P(L_2|B_1, B_2) \cdot P(L_3|B_1, B_3) \cdot P(L_4|B_3)$

b. [3%] Write the definition of Markov Blanket. What is the Markov Blanket of variable L_2?

Given a node in the Bayesian network, the Markov Blanket is the set of nodes consisting of the node's parents, children and children's parents. All the other nodes beyond the Markov Blanket are considered independent of this node.

The Markov Blanket of L_2 is $\{B_1, B_2\}$

c. [3%] If we find L_4 is on, can we gain information about buttons? Please write down your conclusion for each button.

The probability that B_1 is pushed is:

$P(B_1 | L_4) = P(B_1)$ (So they are independent events)

Similarly, we have $P(B_2 | L_4) = P(B_2)$

$$P(B_3 | L_4) = \frac{P(L_4 | B_3) \cdot P(B_3)}{P(L_4)}$$

(B_3 is more likely to be pressed)

d. [3%] If we press B_1 then find that L_3 is on, can anything be inferred about B_3? Please give a simple reason.

$$P(B_3 \mid B_1, L_3) = \frac{P(B_1, B_3, L_3)}{P(B_1, L_3)} = \frac{P(B_1)P(B_3)\cdot P(L_3 \mid B_1, B_3)}{P(B_1, L_3)}$$

$$= \frac{P(B_1)\cdot P(L_3 \mid B_1, B_3)}{P(B_1)\cdot P(L_3 \mid B_1)}\cdot P(B_3) = P(B_3)\cdot \frac{P(L_3 \mid B_1, B_3)}{P(L_3 \mid B_1)}$$

$\cancel{= P(B_1)\ P(B_1)\cdot P(B_3|B_1)\ P(L_3|B_1,B_3)}$ Generrally speaking, when both B_1, B_3 are ~~tur~~ on, L_3 is more likely to on, so $P(L_3|B_1,B_3) > P(L_3|B_1) > 0$, so $P(B_3|B_1, L_3) > P(B_3)$. Therefore in this case B_3 is more likely to be on.

e. [3%] How many independent parameters are needed to describe $P(L_1, L_2, L_3, L_4, B_1, B_2, B_3)$? For example, $P(B_2)$ needs 1 parameter because once we have probability of one state, probability of another state can be computed. Please write down your reason.

① We need $P(B_1)$. $P(B_2)$. $P(B_3)$: ~~$P(L_1|B_1)$ $P(L_2|B_1,B_2)$~~, ~~$P(L_3$~~ 3 states in total

② $P(L_1|B_1)$: 2 states (when ~~当~~ B_1 ~~是~~ true and B_1 is false)

③ $P(L_2|B_1, B_2)$: $2\times2 = 4$ states

③ $P(L_3|B_1, B_3)$: 4 states

④ $P(L_4|B_3)$: 2 states

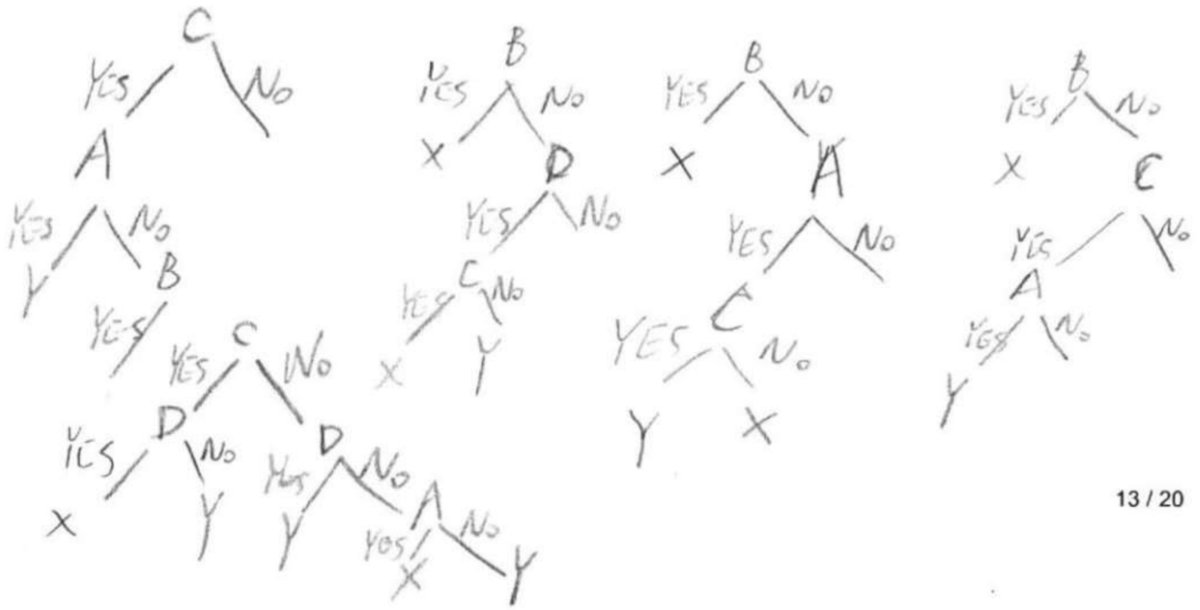
Totally we need $3+2+4+4+2 = 15$ states

EBF91275-A141-4BF4-8605-39A7815CA176
561f17-final
#405

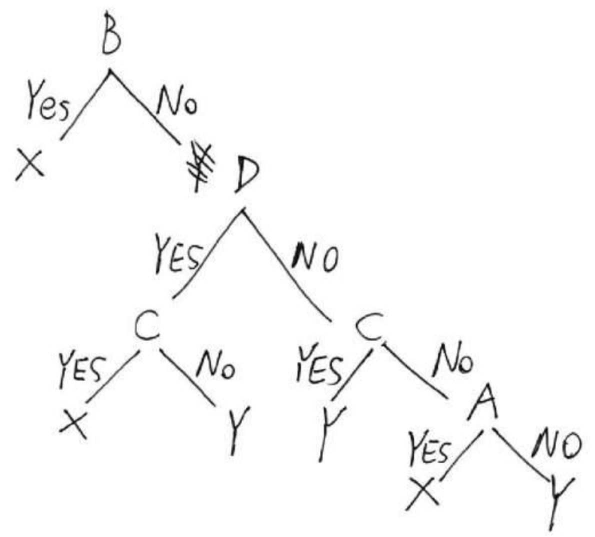
6. [10%] Decision Tree

Researchers have collected some samples of two new materials: X and Y. They select 4 properties to test each sample. Here are the results:

Type	Property A	Property B	Property C	Property D
X	YES	YES	NO	NO
X	NO	NO	YES	YES
X	NO	YES	YES	YES
X	YES	NO	NO	NO
X	NO	YES	YES	YES
Y	YES	NO	YES	NO
Y	NO	NO	NO	NO
Y	NO	NO	YES	NO
Y	NO	NO	NO	YES
Y	YES	NO	YES	NO

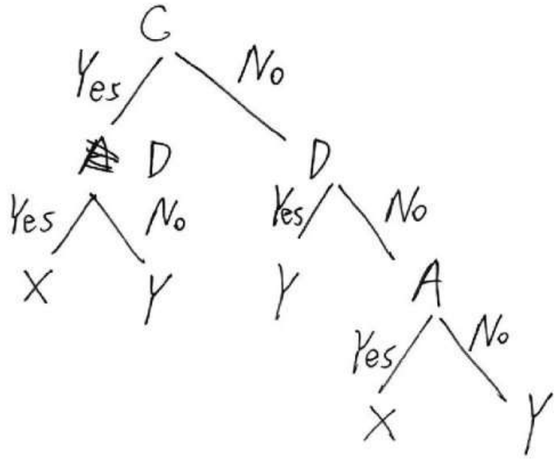


a. [6%] Use the ID3 algorithm to obtain a decision tree greedily. Please draw your tree.

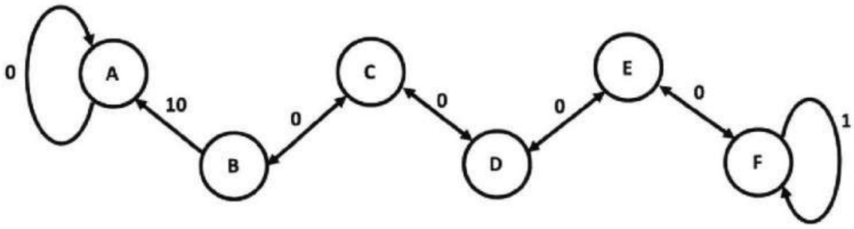


b. [4%] How small can the depth of a 100% accurate decision tree be for this particular problem? Give an example. (Please use only one property on one node)

depth can be at least 3.



7. [15%] MDP



The immediate rewards on the above are indicated by the number on the transition. For example, going from state B to state A gives a reward of 10, but going from state A to itself gives a reward of 0. Some transitions are not allowed, such as from state A to state B.

a. [3%] Please write down the optimal action at each step (A, B, C, D, E, F) if the discount factor (γ) = 1. You **MUST** show the derivation or explain your answers, otherwise you will get 0 points even though your final answers are correct.

~~$U^*(A)$~~ ~~$\pi(A\to A)=0, \pi(A$~~

①

~~$U(A) \le \pi_A(A\to A)=0, \pi_A($~~ the only action for A is go to A.

So $\pi(A)=A$

~~$\pi_B(B\to A)$~~ ~~②~~ ~~②~~ and $U^*(B)=10+\gamma\times 0=10$

~~② Suppose~~ ② $\arg\max_a U^*(B) = \arg\max_a \left(U(B, B\to A), U(B, B\to C)\right)$

$= \arg\max_a (10+\gamma\times 0, \ 0+\gamma\times 0)$

$= B\to A$

$\therefore \pi(B) = B\to A$

③ $\arg\max_a U^*(c) = \arg\max_a \left(U(C, C\to B), U(C, C\to D)\right)$

$= \arg\max_a (0+\gamma\times 10, \ 0+\gamma\times 0) = C\to B$ ~~for C→D~~

$\therefore \pi(C) = C\to B$. $U^*(C) = 10\times\gamma + 0 = 10$

④ ~~#~~ $\pi(D) = \arg\max_a \left(U(D, D\to C), U(D, D\to E)\right) = \arg\max_a (10, 0) = D\to C$

Similarly $\therefore \pi(D) = D\to C$. $U^*(D)=10$.

⑤ $U^*(E)=10$. $\pi(E)=E\to D$. $U^*(F)=$ 10 ~~$F\to E$~~. $\pi(F)= F\to E$

b. [12%] Same as above, but with the discount factor (γ) = 0.01. You **MUST** show the derivation or explain your answers, otherwise you will get 0 points even though your final answers are correct.

for A, the only action is going to A

for B, $U^*(B) = \max(U(B \to A), U(B \to B))$

$= \max(\cdot 10 + \gamma \times 0, 0) = 10$

$\therefore \pi(B) = B \to A$, $U^*(B) = 10$

for C: ~~$U^*(C)$~~ $U(C) = \max(U(C \to B), U(C \to D))$

$= \max(0 + 0.01 \times 10, 0)$

$= 1$

State transition for C missing. -2 marks

for F: $U(F) = \max(U(F \to F), U(F \to E))$

$= \max(0 + \gamma \cdot U^*(E), 1 + 0)$

$= 1$

$\pi(F) = F \to F$

for E: $U(E) = \max(U(E \to F), U(E \to D))$

$= \max(0 + 0.01 \times 1, 0 + 0.01 \times 0)$

$= 0.01$

$\pi(E) = E \to F$

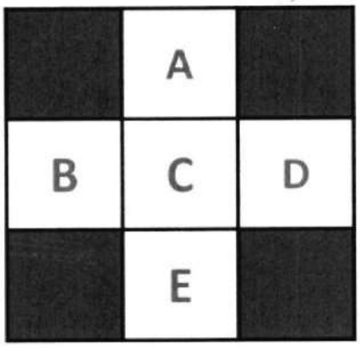
for D: $U(D) = \max(0 + 0.01 \times 1, 0 + 0.01 \times 0.01)$

$= 0.01$

$\pi(D) = D \to C$

8. [20%] Learning in Gridworld

Consider the following gridworld. We would like to use TD learning and q-learning to find the value of these states.



figure

Suppose that we have the following 4 observed transitions:
(B, East, C, 2), (C, South, E, 4), (C, East, A, 6), (B, East, C, 2)
Hint: (B, East, C, 2) means the agent started in state B, went East, received 2 units of reward and ended up in state C.

The initial utility value of each state is 0. Assume that the discount factor γ = 1 and the learning rate α = 0.5.

964E4510-9503-44F2-A09A-52E2FD29796A
561f17-final
#405

a. [8%] What are the learned values for the 5 states (A, B, C, D, E) from TD learning after all four observations? You **MUST** show the derivation or explain your answers, otherwise you will get 0 points even though your final answers are correct.

$U\cancel{Q}(B, East) = U(B, East)\cdot(1-\alpha) + \alpha(R(B) + \cancel{Q}Q(C))$

$= 0 + 0.5 \times (2 + 0)$

$= 1$

2 observations correct. 4 marks awarded

$U\cancel{Q}(C, South) = 0 + 0.5 \times (4 + 0) = 2$

$U\cancel{Q}(C, East) = 0.5 \times 0 + 0.5 \times (6 + 0)$

$= 3$

$U\cancel{Q}(B, East) = 0.5 \times 1 + 0.5 \times (2 + 1 \times 2)$

$= 0.5 + 2 = 2.5$

Q19 Score: **9.0**

b. [12%] What are the learned Q-values from Q-learning after all four observations? You **MUST** show the derivation or explain your answers, otherwise you will get 0 points even though your final answers are correct.

$Q^*(C, East) = \cancel{r \cdot 6 = 6}$ $Q(C, East)(1-\alpha) + \alpha(R(B) + \gamma Q(C))$

$= 0 + 0.5 \times 2 = 1$

3 observations correct. 9 marks awarded

$Q^*(C, South) = \cancel{r \cdot 4 = 4}$ $0 + 0.5 \times 4$

$= 2$

$Q^*(B, East) = \cancel{0.5 \times}\ \cancel{r Q^*(C,}$

$= 0.5 \times 0 + 0.5 \times 6$ $\cancel{r \max(Q^*(C, East), Q^*(C, South))}$

$= 3$

$Q(B, East) = \cancel{6}$ $0.5 \times 1 +$

$0.5 \times (2 + 2)$

$= 2.5$