

CSE 415 Winter 2024 Assignment 4

Last name: Labbé First name: Alexandre

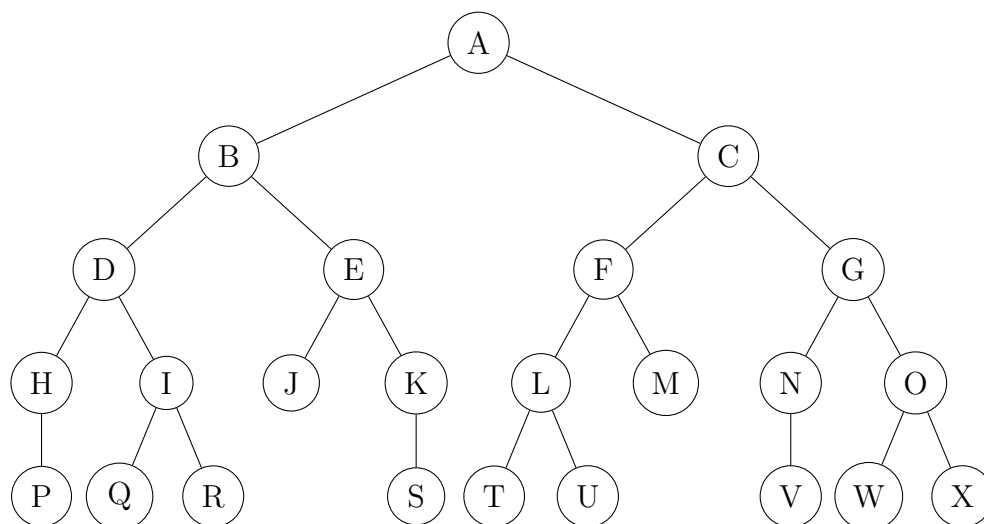
This is an individual-work assignment. Do not collaborate on this assignment.

Prepare your answers in a neat, easy-to-read PDF. Our grading rubric will be set up such that when a question is not easily readable or not correctly tagged or with pages repeated or out of order, then points will be deducted. However, if all answers are clearly presented, in proper order, and tagged correctly when submitted to Gradescope, we will award a 5-point bonus. (Thus the maximum points for A4 will be 105.)

If you choose to typeset your answers in Latex using the template file for this document, please put your answers in **blue** while leaving the original text black.

1 Basic Search

Use the following tree to answer the questions below comparing Breadth First Search, Depth First Search, and Iterative-Deepening Depth First Search. Assume that children are visited from left to right.



- (a) (3 points) Write out the order that the nodes are expanded in using Breadth First Search, starting from A and searching to N.

A B C D E F G H I J K L M N

- (b) (3 points) Write out the order that the nodes are expanded in using Depth First Search, starting from A and searching to N.

A B D H P I Q R E J K S C F L T U M G N

- (c) (3 points) Write out the order that the nodes are expanded in using Iterative-Deepening Depth First Search, starting from A and searching to N. If a node is repeated, make sure to include it each time it is expanded.

A A B C A B D E C F G A B D H I E J K C F L M G N

- (d) (4 points) Which of the three search algorithms (BFS, DFS, IDDFS) has the smallest maximum size of the open list while searching from A to K? What is the maximum size of the open list for that algorithm?

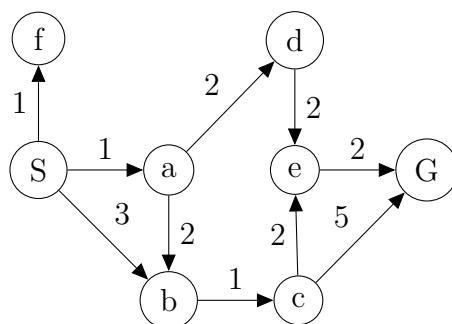
The search algorithm with the smallest maximum size of the open list is IDDFS with a maximum size of 4 searching from A to K

- (e) (2 points) True or False: BFS, DFS, and IDDFS will each return the same path on this tree starting at A searching to K.

True

- (f) (2 points) True or False: Given the same start and goal state, BFS, DFS, and IDDFS will always return the same path for any search graph (not necessarily a tree).
False
- (g) (4 points) Suppose you are running your search algorithm on an embedded system device with highly limited memory available. Which of the three search algorithms (BFS, DFS, IDDFS) would you choose to run to minimize the memory used by the algorithm? Justify your answer with a one sentence explanation.
IDDFS because IDDFS only needs to store the nodes of the tree at the current depth, then discards them when it is time to move to the next iteration of the call.
- (h) (4 points) Suppose you need to run a search algorithm but you are confident that the vast majority of your search goals will be near the root of the tree (for example, G or earlier in the alphabet on this tree). Which of the three search algorithms (BFS, DFS, IDDFS) would you use to find the shortest path in the least amount of time given these conditions? Justify your answer with a one sentence explanation.
BFS because it is more time-efficient than DFS at finding nodes higher in the tree on average, and IDDFS' advantage over BFS is the memory, which is not of concern.

2 Heuristic Search



For the following questions, consider three heuristics h_1 , h_2 , h_3 . The table below indicates the estimated cost to goal, G , for each of the heuristics for each node in the search graph.

state (s)	S	a	b	c	d	e	f	G
heuristic $h_1(s)$	6	5	2	2	2	2	6	0
heuristic $h_2(s)$	6	6	4	5	4	1	5	0
heuristic $h_3(s)$	6	6	4	3	4	2	8	0

- (a) (2 point) What does it mean for a heuristic to be "admissible"?
- A heuristic is admissible when the estimated cost it provides to reach the goal node is never higher than the actual cost to reach the goal node, for any goal state and for any starting node.
- (b) (3 points) Which heuristics among $\{h_1, h_2, h_3\}$ shown above are admissible? For heuristics which are not admissible, please identify the nodes where admissibility is violated.
- Heuristics $\{h_1, h_3\}$ are admissible. Admissibility is violated in h_2 at node c
- (c) (2 point) What does it mean for a heuristic to be "consistent"?
- A heuristic is consistent when, for each edge connecting two nodes, the difference of the heuristic value between the two nodes is never greater than the true distance between the two nodes.
- (d) (3 points) Which heuristics are consistent? For heuristics which are not consistent, please identify the edges where consistency is violated.
- Heuristic h_3 is consistent. h_1 violates consistency at edges $\{(S, b), (a, d), (a, b)\}$. h_2 violates consistency at edges $\{(c, e), (d, e)\}$.
- (e) (2 points) What does it mean for one heuristic to dominate another heuristic?
- A heuristic dominates another when the heuristic value for all nodes for one function

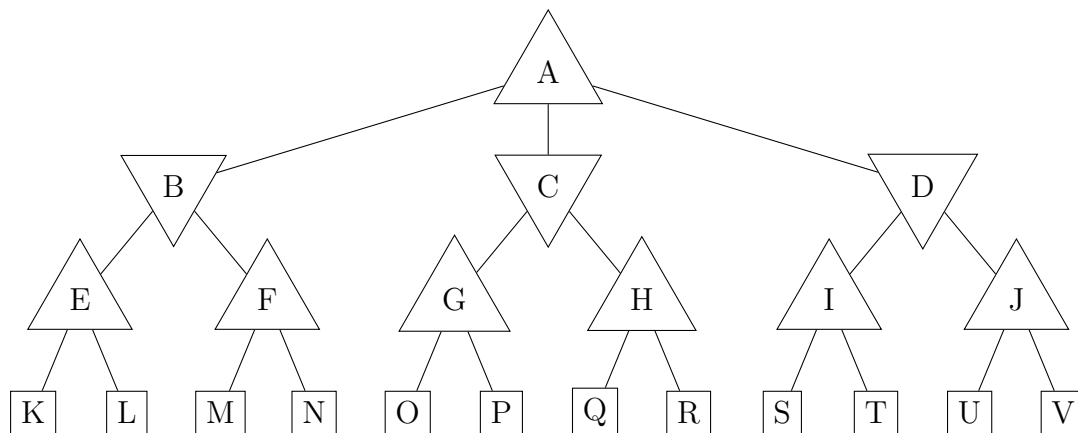
is greater than the heuristic value for all nodes in the other. We assume that both heuristics are admissible.

- (f) (2 points) Of the heuristics provided above, which would you consider the best heuristic? Explain your reasoning in terms of admissibility, consistency, and dominance.
Heuristic h_3 is the best heuristic because it is the only heuristic that is both admissible and consistent. It also dominates h_1 , the only other heuristic that is admissible.
- (g) (4 points) Show the node expansion order going from S to G using A^* with the heuristic you identified in the previous question. Also provide the final (solution) path obtained.
Node expansion path: S, a, b, d, c, e, G . The final path is S, a, d, e, g .
- (h) (1 points) Consider h_1 – what single change would you make to improve this heuristic?
I would change $h(a)$ to equal 4, rather than 5.
- (i) (4 points) Having made the change above, has your determination of the best heuristic changed? Explain your reasoning in terms of admissibility, consistency, and dominance.
This change would keep the heuristic admissible, but improve its consistency. The edges $\{(a, d), (a, b)\}$ would become consistent. Edge (S, b) would remain not-consistent, but since that edge is not used in the shortest path, it would not be important. The heuristic would remain dominated by h_3 , but I believe this is the best single change to improve h_1 . Full consistency could be achieved by also decreasing the value of $h(s)$ by one.
- (j) (2 points) Explain why, when using the A^* search algorithm, it is important to continue the expansion process until the goal state is removed from the OPEN list and becomes the current state, rather than terminating as soon as the GOAL state is found.
The priority value of the goal state can be updated by other nodes being expanded. If other nodes have not had the chance to update the goal nodes priority, then the shortest path has not been found. A path has been found, but possibly not the shortest.

3 Adversarial Search and Alpha-Beta Pruning

3.1 Alpha-Beta Pruning

Use this search tree along with the table of state evaluation values to answer the questions below. By default, children are processed from left to right.



state	K	L	M	N	O	P	Q	R	S	T	U	V
evaluation	5	-3	6	8	-3	-5	6	-4	9	-5	2	1

- (a) (8 points) Use minimax to perform adversarial search with alpha-beta pruning on the tree above. Fill in the values in the table below as you go. For α and β values, write the values that are passed from the state's parent to that state. For example, the value of state K will not be shown in the $\alpha - \beta$ values of state E, but would be reflected in the $\alpha - \beta$ values of state L. If a state does not have to be evaluated, do not write any values for it in the table.
- (b) (7 points) Let's assume that you ~~reverse~~ *change* the order of the leaf nodes in the above tree A (call that ~~reversed~~ tree B). The evaluation function f represents the utilities for the first tree A, and let the evaluation function g represent the utilities for the second tree B. ~~After performing alpha beta on this reversed tree, what observations can you make about the two evaluation functions? Be sure to answer what effect switching the order of the evaluation function utilities has. What are some ways to improve the evaluation functions in general?~~

Is it possible to completely prevent pruning in the tree by just reordering the leaf nodes in the tree? If so, explain why. If not, what is the least number of leaf nodes whose values

state	value	α	β
A	5	N/A	N/A
B	5	$-\infty$	∞
C	-3	5	∞
D	2	5	∞
E	5	$-\infty$	∞
F	6	$-\infty$	5
G	-3	5	∞
H			
I	9	5	∞
J	2	5	9
K	5	$-\infty$	∞
L	-3	5	∞
M	6	$-\infty$	5
N	8		
O	-3	5	∞
P	-5	5	∞
Q	6		
R	-4		
S	9	5	∞
T	-5	9	∞
U	2	5	9
V	1	5	2

need to be changed to prevent pruning. In either case, describe your new tree. (Include a drawing if you think it will be more clear that way.)

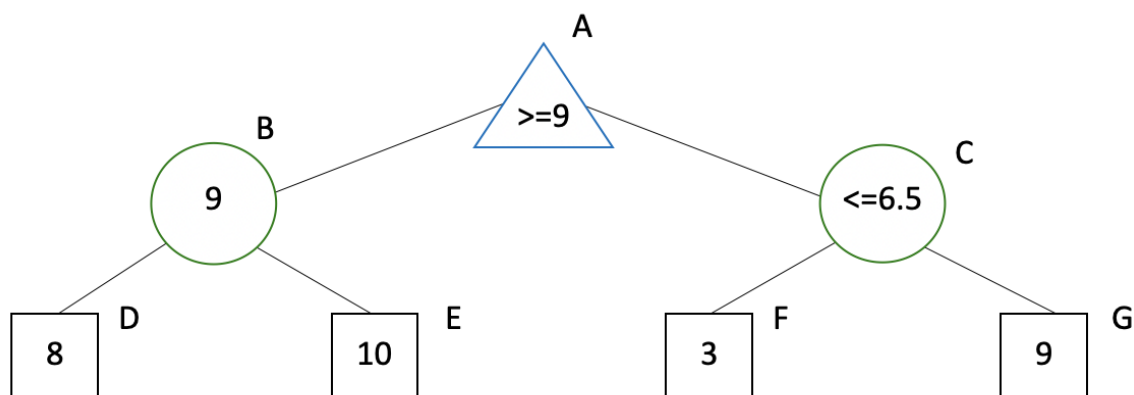
Two leaf node values must be changed. A possible reordering of the leaves that would result in no pruning of the tree is as follows (assuming leaf nodes can only be reordered under their parent leaf node as is stated on ed thread 163):

state	K	L	M	N	O	P	Q	R	S	T	U	V
evaluation	5	-3	-4	8	6	-5	5	-4	9	-5	2	1

This reordering was found by decreasing the value of node M to a value less than the beta value passed to it, and increasing node O to a value greater than the beta value passed to it, and switching nodes Q and R. Since switching nodes under the first E and F does not occur in avoiding pruning, value N must be changed to avoid pruning. By increasing the value of node O, we avoid the pruning of everything to the right of node C. But in that case, node R still is pruned. We avoid this by switching Q and R since we need node Q to have a value less than the beta value passed to it.

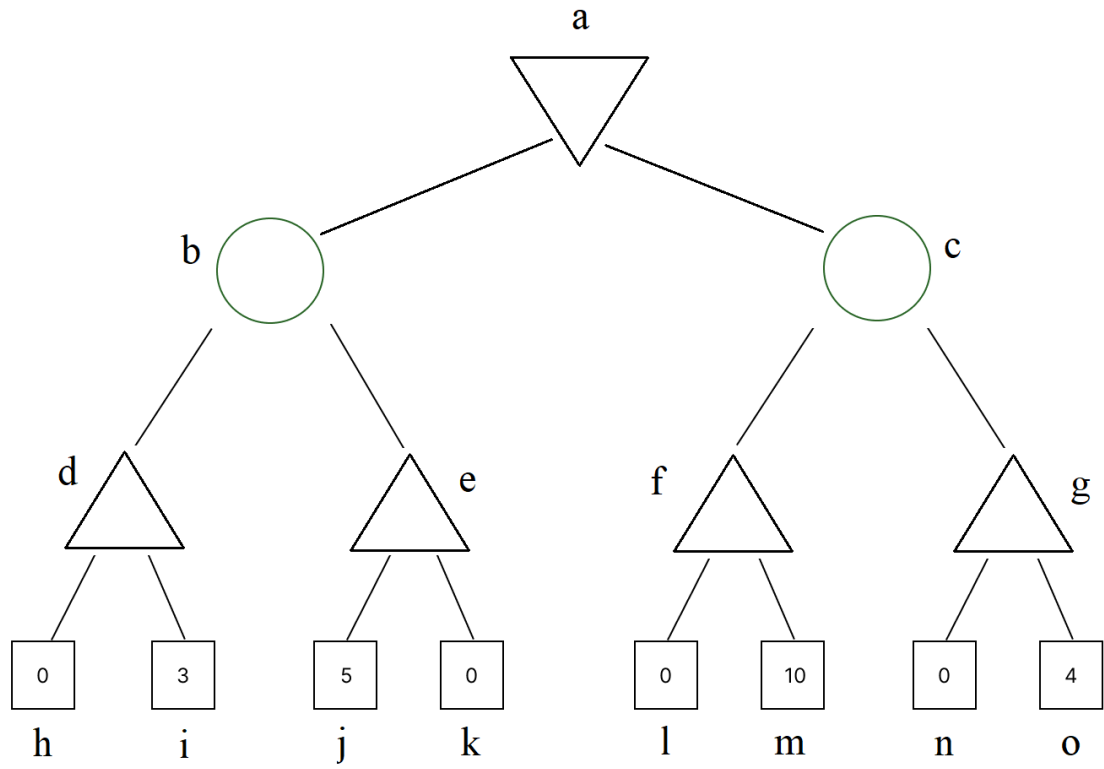
3.2 Pruning with Chance Nodes

Although alpha-beta pruning cannot be applied directly to searching trees that contain chance nodes, reasoning like that inherent in alpha-beta search can sometimes be applied when static values are constrained to lie within given ranges of values. For example, if all leaf-node values must be values $f(s)$ such that $0 \leq f(s) \leq 10$, then node G can be pruned in the tree below, because Max can get 9 by moving left at A to B, and if Max goes to C and finds $f(F) = 3$, Max can reason that getting $f(G)$ is useless because if $f(G) = 10$ which is largest allowed, then the value at C is $(3+10)/2 = 6.5$, which is inferior to 9.



- (c) (6 points) In the following game tree, determine where pruning can be performed using the same range assumption as above. Show where there are cutoffs. Also after pruning, change the value of one of the nodes such that the pruning will no longer occur.
 Pruning will occur at node k. If node j is changed to a value less than node i, then pruning will not occur at node k. Pruning will also occur at node g. If node m is changes to 0, then pruning will not occur at node g.

- (d) (4 points) Explain your reasoning for each cutoff.
 Pruning will occur at node k because of regular alpha-beta pruning behavior. Pruning will occur at node g because since node m is of value 10, the average value that will be passed up to node c will be greater than 5, meaning that the minimizing head node will go to b since the average value there will be 4.



4 Probabilistic Inference and Factoring Joint Distributions

- (a) (10 points) Consider a situation where we are modeling the sale of Ice cream (S), which is dependent on three variables: Weather condition (W) and Day of the week (D). You are given the joint probability distribution of these variables in the table below:

Weather (W)	Day (D)	Ice Cream Sales (S)	P(W,D,S)
Sunny (Su)	Weekday (Wd)	High	0.108
Sunny (Su)	Weekday (Wd)	Low	0.252
Sunny (Su)	Weekend (We)	High	0.216
Sunny (Su)	Weekend (We)	Low	0.024
Cloudy (Cl)	Weekday (Wd)	High	0.024
Cloudy (Cl)	Weekday (Wd)	low	0.096
Cloudy (Cl)	Weekend (We)	High	0.032
Cloudy (Cl)	Weekend (We)	Low	0.048
Rainy (Ra)	Weekday (Wd)	High	0.012
Rainy (Ra)	Weekday (Wd)	Low	0.108
Rainy (Ra)	Weekend (We)	High	0.016
Rainy (Ra)	Weekend (We)	Low	0.064

Table 1: Joint Probability Distribution of Weather, Day, and Ice Cream Sales.

Answer these questions based on the joint probability distribution table above:

- (i) What is the marginal distribution $P(W)$?

Weather (W)	P(W)
Sunny (Su)	0.6
Cloudy (Cl)	0.2
Rainy (Ra)	0.2

(ii) What is the marginal distribution $P(W, D)$?

Weather (W)	Day (D)	$P(W,D)$
Sunny (Su)	Weekday (Wd)	0.36
Sunny (Su)	Weekend (We)	0.24
Cloudy (Cl)	Weekday (Wd)	0.12
Cloudy (Cl)	Weekend (We)	0.08
Rainy (Ra)	Weekday (Wd)	0.12
Rainy (Ra)	Weekend (We)	0.08

(iii) What is the marginal distribution $P(W, S)$?

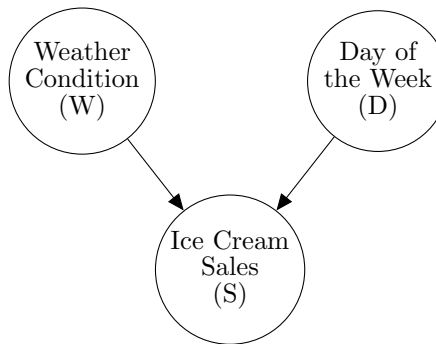
Weather (W)	Ice Cream Sales (S)	$P(W,S)$
Sunny (Su)	High	0.324
Sunny (Su)	Low	0.276
Cloudy (Cl)	High	0.056
Cloudy (Cl)	low	0.144
Rainy (Ra)	High	0.028
Rainy (Ra)	Low	0.172

(iv) What is the marginal distribution $P(D)$?

Day (D)	$P(D)$
Weekday (Wd)	0.6
Weekend (We)	0.4

(v) What is the conditional distribution $P(S \mid W, D)$?

Weather (W)	Day (D)	Ice Cream Sales (S)	$P(S W,D)$
Sunny (Su)	Weekday (Wd)	High	0.108/0.36
Sunny (Su)	Weekday (Wd)	Low	0.252/0.36
Sunny (Su)	Weekend (We)	High	0.216/0.24
Sunny (Su)	Weekend (We)	Low	0.024/0.24
Cloudy (Cl)	Weekday (Wd)	High	0.024/0.12
Cloudy (Cl)	Weekday (Wd)	low	0.096/0.12
Cloudy (Cl)	Weekend (We)	High	0.032/0.08
Cloudy (Cl)	Weekend (We)	Low	0.048/0.08
Rainy (Ra)	Weekday (Wd)	High	0.012/0.12
Rainy (Ra)	Weekday (Wd)	Low	0.108/0.12
Rainy (Ra)	Weekend (We)	High	0.016/0.08
Rainy (Ra)	Weekend (We)	Low	0.064/0.08



(b) (6 points) Consider the Bayes Net of the same Ice Cream Sales problem we were considering. Answer the following question based on it:

- (i) Write the equation for the joint distribution of the three variables as a product of the appropriate marginal and conditional distributions, according to the Bayes net graph.

$$P(W, D, S) = P(W) * P(D) * P(S|W, D)$$

- (ii) What is the probability of high ice cream sales on a weekend, given that the weather is sunny?

$$P(S=h \mid D=We, W=s) = P(W=s, D=We, S=h) / (P(W=s) * P(D=We)) \\ = \frac{0.216}{0.6 * 0.4}$$

- (iii) Given that ice cream sales are high, what is the probability that the day is a weekend?

$$P(\text{Weekend} \mid \text{High}) = \frac{0.264}{0.6}$$

- (iv) What is the probability of having a sunny weekday with high ice cream sales compared to having a rainy weekend with low ice cream sales?

$$P(W = \text{sunny}, D = \text{weekday}, S = \text{high}) = P(W = \text{sunny})P(D = \text{weekday})P(S = \text{high} \mid W = \text{sunny}, D = \text{weekday})$$

$$= 0.6 * 0.6 * \frac{0.108}{0.408} = 0.095$$

$$P(W = \text{rainy}, D = \text{weekend}, S = \text{low}) = P(W = \text{rainy})P(D = \text{weekend})P(S = \text{low} \mid W = \text{rainy}, D = \text{weekend})$$

$$= 0.4 * 0.4 * \frac{0.064}{0.052} = 0.197$$

- (c) (9 points) Compare the sizes of the representations for the full joint distribution in (a) and the Bayes net in (b) as follows:
- (i) Give an expression for the number of free parameters in Table 1 in terms of the sizes of the domains of the variables W, D, and S. You can use $|W|$ to represent the size of the domain of W, for example.

$$\text{number of free parameters} = |W| * |D| * |S| - 1$$
 - (ii) Give the resulting integer value for your expression.

$$11$$
 - (iii) Give a corresponding expression for the number of free parameters in the Bayes net in (b).

$$\text{Number of free parameters} = |W| - 1 + |D| - 1 + |W| * |D| * (|S| - 1)$$
 - (iv) Give the integer value of the expression for the Bayes net.

$$2 + 1 + 3 * 2 * (2 - 1) = 9$$
 - (v) Is there any savings in storage, according to the number of free parameters? Explain.
 Since bayes nets' exploit conditional independance among the variables in the distribution to reduce the number of free parameters, there is infact a saving in storage when comparing the bayes net to the full joint distribution.
 - (vi) What kind of changes to this joint distribution and Bayes net could lead to a significant savings?
 Increasing the number of variables and the sizes of their domains would have saving by decreasing the number of free parameters in relation to the full joint distribution that these changes would cause.