National University of Singapore

CS2109S—Introduction to AI and Machine Learning

MIDTERM ASSESSMENT

Semester 1, 2023/2024

Time allowed: 2 hours

INSTRUCTIONS TO STUDENTS

1. Write down your **Student Number** on the answer sheet and shade completely the corresponding bubbles in the grid for each digit or letter. **DO NOT WRITE YOUR NAME!**

2. The assessment paper contains **FIVE (5) questions** and comprises **THIRTEEN (13) pages** including this cover page.

3. Weightage of questions is given in square brackets. The maximum attainable score is 100.

4. This is an **OPEN-SHEET** assessment, which means that you are permitted to use a double-sided A4-sized cheatsheet.

5. You are allowed to bring a calculator, but it cannot have any form of external communication capability, i.e. not Wifi- or 4G-enabled. Mobile phones and tablets are not allowed.

6. All questions must be answered in the space provided on the answer sheet; no extra sheets will be accepted as answers.

7. You are allowed to write with pencils, as long as it is legible.

8. **Marks may be deducted** for unrecognisable handwriting and/or for not shading the student number properly.

9. You must submit only the **ANSWER SHEET** and no other documents. The question set may be used as scratch paper.

10. An excerpt of the question may be provided in the answer sheet to aid you in answering in the correct box, where applicable. It is not the exact question. You should still refer to the original question in this question booklet.

This page is intentionally left blank.

It may be used as scratch paper.

Question 1: Make Square [32 marks]

Your young cousin, Ben Bitdiddle, presents you with a box filled with n sticks [s_1, ..., s_n], where s_i represents the length of the i-th stick. Your task is to arrange **all** of these sticks to form a square. You need to use every stick, and you are not allowed to break or modify any of the sticks. You can only select one stick at a time from the box.

For instance, if the sticks provided are of lengths [1, 3, 3, 2, 3], we can arrange them into a square as illustrated in Figure 1.

Note: *While this is a written exam question, you are expected to be able to write the code required to solve this problem. In your answers below, your goal is to provide enough detail to convince the examiners that you can write the code required to solve this problem. In other words, someone who reads your answers below should be able to write the code required to solve this problem.*

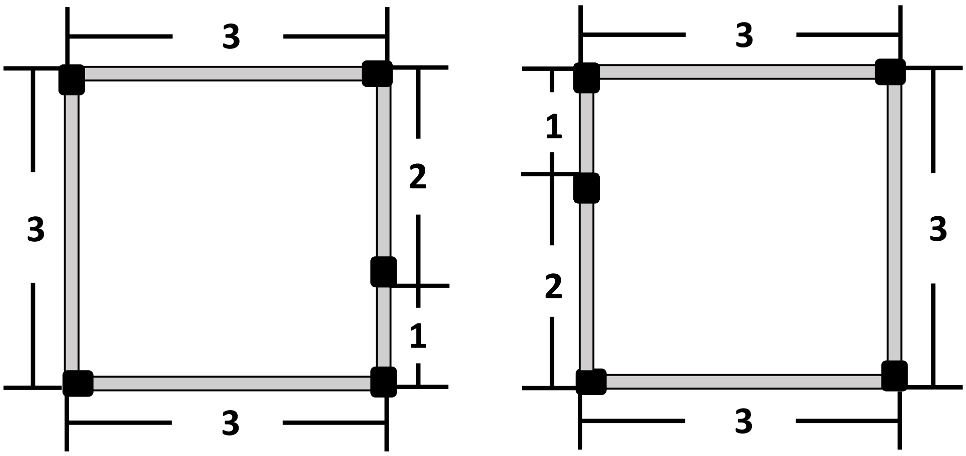


Figure 1: Possible squares

Uninformed Search

A. Propose a state representation to formulate the problem as an uninformed search. [2 marks]

B. What is the representation invariant for your state representation from Part (A), i.e. what are the condition(s) that the state representation must satisfy, in order to be valid? [2 marks]

C. What are the initial and goal states for the problem when using the state representation proposed in Part (A)? [2 marks]

D. Define the actions for the state representation in Part (A)? [2 marks]

E. If we opt to employ TREE-SEARCH (See Appendix) with the search formulation in Parts (A) to (D) above, we can utilize one of the following algorithms:

1. Depth-first search (DFS)
2. Breath-first search (BFS)
3. Depth-limited search (DLS)
4. Iterative-deepening search (IDS)

Which of the above search algorithms should we use? Explain. [4 marks]

F. Should we have opted for GRAPH-SEARCH (See Appendix) instead of TREE-SEARCH? Explain. [3 marks]

Local Search

As a student enrolled in CS2109S, you have acquired knowledge about various search algorithms. After thinking about this problem, you realize that it is also possible to formulate the problem as a local search problem.

Remember that formulating a solution as a local search problem involves three essential steps:

1. Establish a candidate solution.
2. Define a transition function to create new candidate solutions
3. Develop a heuristic to evaluate the "goodness" of a candidate solution.

G. Provide an initial candidate solution based on the square-making description above. [2 marks]

H. Define a reasonable transition function for generating new candidate solutions. [2 marks]

I. Suggest a reasonable heuristic function that can be used with your local search formulation in Parts (G) and (H) above. Explain how this heuristic can also serve as a goal test to determine if a solution has been achieved. [2 marks]

Variant #1: Make $n \times n$ Square

Ben Bitdiddle has a new request. This time, instead of using all the available sticks, we are required to create a square of size $n \times n$ using the smallest number of sticks from the box. There is no guarantee however that we can necessarily create the $n \times n$ square using the available sticks.

J. Supposed we want to frame Variant #1 for TREE-SEARCH. Describe the modifications we will need to make from the formulation in Parts (A) to (D) above. Your formulation should terminate after a finite number of steps. If you are unable to come up with a formulation that can terminate, you can still get partial credit for this problem. [6 marks]

K. Suppose we decide to employ TREE-SEARCH (refer to the Appendix) with the state representation described in Part (J) using one of the following algorithms:

1. Depth-first search (DFS)
2. Breath-first search (BFS)
3. Depth-limited search (DLS)
4. Iterative-deepening search (IDS)

Which of the above search algorithms should we use? Explain. [2 marks]

L. In your proposed formulation in Parts (J) and (K), is your TREE-SEARCH complete and/or optimal? Will your TREE-SEARCH always terminate? Explain. [3 marks]

Question 2: Snake [26 marks]

In an $n \times n$ grid, there is a snake spanning 2 cells, with its current position denoted as $\{(x_t, y_t), (x_h, y_h)\}$, where t represents the tail and h represents the head. The snake's objective is to move to the goal position at $\{(x_{gt}, y_{gt}), (x_{gh}, y_{gh})\}$, where gt stands for the tail goal and gh denotes the head goal, in the minimal number of steps. It is essential to note that the snake must remain within the boundaries of the grid at all times.

In each step, the snake can to make the following moves:

1. Move one cell to the left/right if the snake is in a horizontal position. This move maintains the snake's horizontal orientation.
2. Move up/down one cell if the snake is in a vertical position. This move preserves the snake's vertical alignment.
3. Rotate by 90°, 180°, or 270° while ensuring that the tail remains fixed in the same position.

For instance, consider Figure 2 (1), where the snake's current position is $\{(2,2),(3,2)\}$, and the goal position is $\{(3,4),(4,4)\}$. As the snake is currently oriented horizontally, it can shift by 1 cell to the left/right, as shown in Figure 2 (2). Additionally, the snake can also rotate to one of three valid positions, as illustrated in Figure 2 (3).

After discussing this problem with your cousin, Ben Bitdiddle, he proposed some heuristics to be used with A* search. Your task is to evaluate each heuristic and use **shading** to denote whether it is admissible/consistent or not. Subsequently, please provide justifications for your assessments in the designated boxes.

Hint: The Manhattan Distance is defined as $MD(a,b) = |x_a - x_b| + |y_a - y_b|$, where x_a and y_a represent the row and column index of a, and the same applies to b.

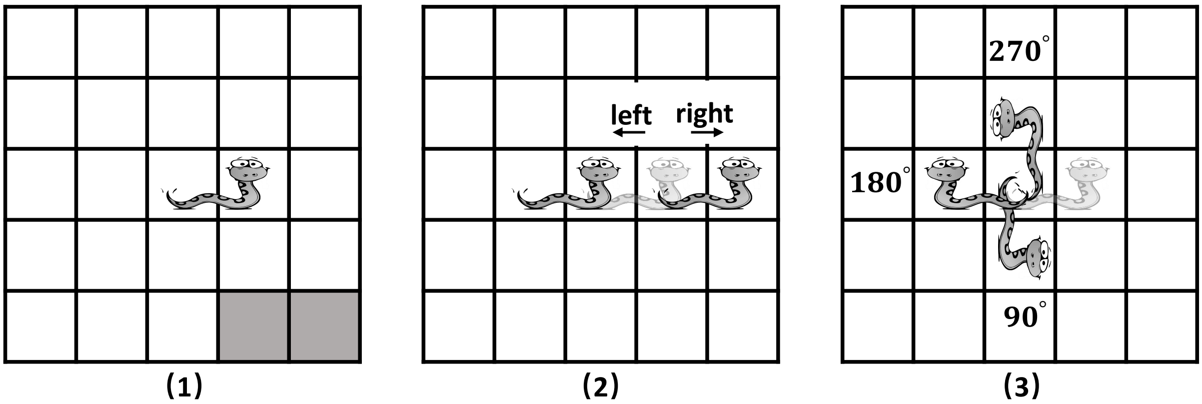


Figure 2: Snake

A. The sum of the misplaced head and misplaced tail.

$$h_A = m_h + m_t \tag{1}$$

Here, m_h and m_t denote the misplacement status of the head and tail, respectively. Specifically, $m_h = 0$ indicates that $x_h = x_{gh}$ and $y_h = y_{gh}$, while $m_h = 1$ indicates that the head is not at its goal location. m_t is similarly defined for the tail. Is h_A admissible? Explain. [3 marks]

B. The Manhattan distance between the current snake head position and its goal head position:

$$h_B = MD(h, gh) \tag{2}$$

Is h_B admissible? Explain. [3 marks]

C. The Manhattan distance between the current snake tail position and its goal tail position:

$$h_C = MD(t, gt) \tag{3}$$

Is h_C admissible? Explain. [3 marks]

Variant #1: Snake and Apple

Now, let us consider a modified version of the Snake problem, where the snake is required to 1) consume an apple located at (x_{ap}, y_{ap}), before 2) moving to the final goal position. The apple can be placed at any location within the grid. To eat the apple, the head of the snake needs to moves to the cell with the apple. And there is no additional cost associated with eating the apple. After the snake eats the apple, the apple disappears from the grid, causing the distances between the apple and other positions all reset to zero. The valid actions available to the snake remain the same as described previously.

As shown in Figure 3 (1), there is an apple situated at $(1,4)$. To consume the apple, the snake's head must move to the position $(1,4)$ (Figure 3 (2)). Subsequently, the snake should proceed towards the ultimate goal, i.e. coordinates $(4,3)$ and $(4,4)$ for the tail and head (Figure 3 (3)).

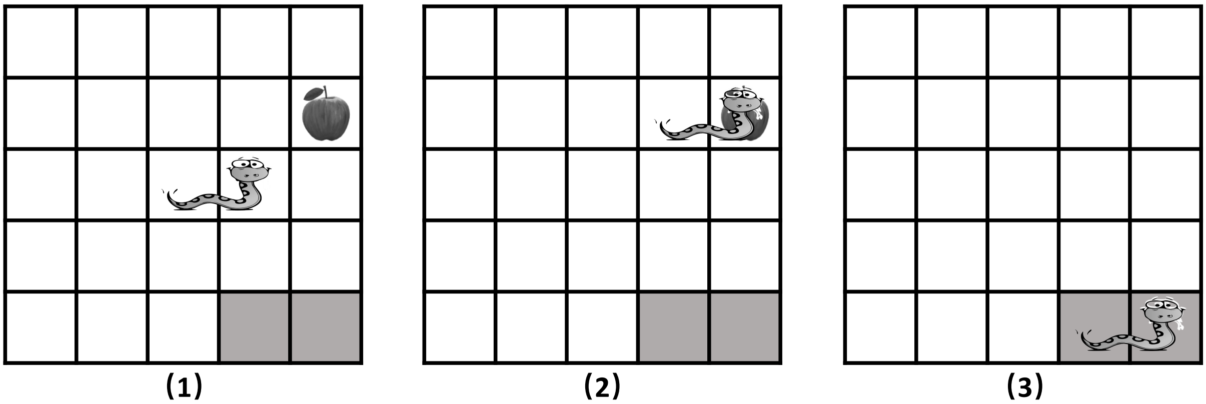


Figure 3: Snake and Apple

D. The sum of two Manhattan distances: one from the current snake tail position to the position of the apple and the other from the position of the apple to the snake tail goal position.

$$h_D = MD(t, ap) + MD(ap, gt) \tag{4}$$

Is h_D admissible and/or consistent for the current problem? Explain. [6 marks]

E. Divide the sum of the Manhattan distance of the current snake head position to the position of the apple and the Manhattan distance of the apple position to the goal position of the head by 2.

$$h_E = (MD(h, ap) + MD(ap, gh))/2 \tag{5}$$

Is h_E admissible and/or consistent for the current problem? Explain. [6 marks]

F. The maximum of the horizontal and vertical distances between the current snake head and apple positions, plus the maximum of the horizontal and vertical distances between the apple and the goal head positions.

$$h_F = \max\{|x_h - x_{ap}|, |y_h - y_{ap}|\} + \max\{|x_{ap} - x_{gh}|, |y_{ap} - y_{gh}|\} \tag{6}$$

If you are asked to pick between h_E and h_F, which one would you pick? Explain. [5 marks]

Question 3: Predicting Loan Eligibility [24 marks]

DBZ Bank has brought you on board as a Machine Learning specialist with the task of developing a system to streamline their loan approval process, ultimately replacing the need for a dedicated loan team. To start this project, you initiated discussions with the team, urging them to elucidate their existing decision-making process for loan approvals. From these insights, you drafted an initial decision tree, as depicted in Figure 4.

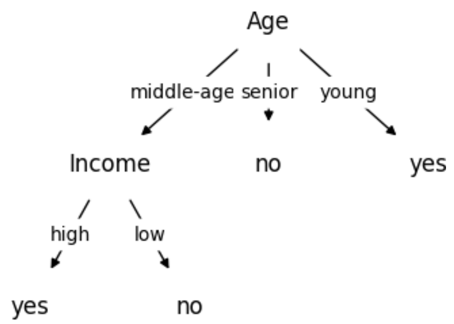


Figure 4: Experts' Decision Tree

Please note that showing your working is not required for this question. However, if you wish to do so, you may include it just in case. All multiple-choice questions will be automatically graded and we will not review your working, unless you submit an appeal due to an error (such as shading incorrectly by mistake).

A. **[Warm Up]** A potential loan taker has a low income. Based on the decision tree in Figure 4, what is the predicted eligibility (Yes or No)? [2 marks]

You're not sure if the experts' decision tree is reliable for predicting loan eligibility. Instead, you plan to create a model using past data. The historical data is presented in Table 1 below.

	Age	Credit Rating	Income	Criminal Record	Loan
0	middle-aged	excellent	low	no	yes
1	senior	excellent	high	yes	no
2	senior	excellent	low	no	yes
3	senior	fair	high	no	yes
4	senior	fair	low	no	no
5	senior	poor	low	yes	no
6	senior	poor	low	yes	no
7	young	excellent	high	yes	yes
8	young	excellent	high	yes	yes
9	young	fair	low	yes	yes

Table 1: Loan Eligibility Data

The information content for a given probability distribution p_i, for $i = 1, ..., n$ is given by:

$$\text{Entropy} = -\sum_{i=1}^{n} p_i \log_2(p_i)$$

B. What is the entropy of the outcomes (Yes/No) in Table 1, rounded to 2 decimal places? *Hint: remember the* log *is in base 2!* [2 marks]

C. You decide to start by creating a one-level decision tree with only one split, using information gain.

[6 marks]

Complete the decision tree in the given template by answering the multiple choice questions. Make sure to use the appropriate calculations to choose the attributes to split.

D. Not content with the performance of the one-level decision tree, you choose to improve prediction accuracy by building a full decision tree.

Suppose that you pick "Income" attribute as the root of your decision tree. Using the data in Table 1 and information gain to split the data, create the remaining decision tree for "Income" = "low". In case of a tie, the priority order for constructing the tree is Income > Age > Credit Rating > Criminal Record. [6 marks]

E. To finish up your work, you aim to demonstrate that the experts' decision tree falls short. To achieve this, you intend to assess its performance using real-world data.

Using the data in Table 1, Figure out the true positives, false positives, true negatives, and false negatives of the experts' decision tree (Figure 4). Then, calculate the precision, recall, and F1 score. Use "Yes" as a positive label and "No" as a negative label. [5 marks]

F. **[Min-Sample Pruning]** Suppose you want to prune the full decision tree you constructed in Part (D) to have at least 3 training data points per leaf. What is the pruned decision tree? Note: the full decision tree includes the "Income" as a root node. [3 marks]

Question 4: Solving Games [15 marks]

A. **[Warm Up]** Consider a two-player game of bowling with a row of **3 pins**. Players take turns, choosing to either remove 1 pin by bowling directly at it, or 2 adjacent pins by bowling to strike both. The game concludes when all pins are removed, and a player is left unable to make a move. At this juncture, the other player is declared the winner (i.e., the player who cannot make a move **loses**).

For instance, envision a scenario where all three bowling pins are intact. The first player has the option to remove pin number 1, 2, 3, 1&2, or 2&3. Assuming the first player chooses to remove pin number 1, the second player can then opt to remove pin number 2, 3, or 2&3. The game proceeds in this manner until one player is unable to remove any pins, signifying that there are no pins left. At this point, the other player emerges victorious.

Consider the **full** game tree that solves this problem and answer the following questions:

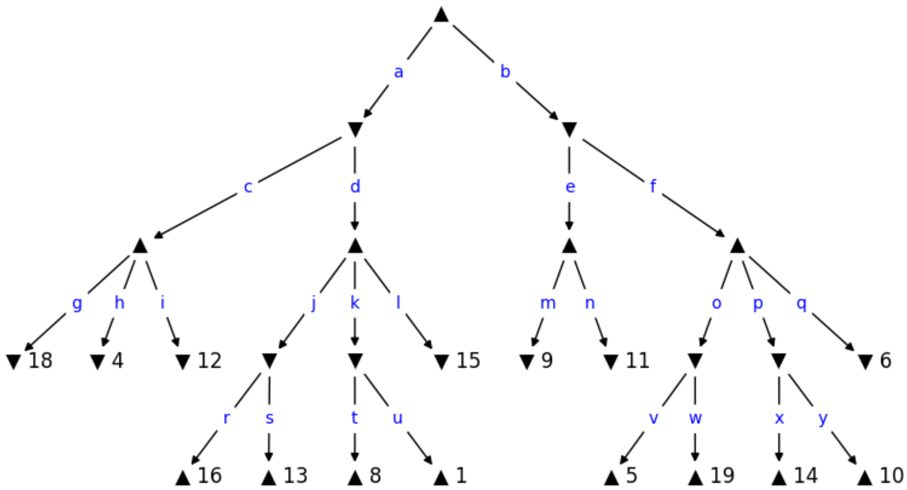
- What is the minimum number of variables required to maintain the game state?
- How many nodes (not including the leaves) are there in the game tree?
- What is the number of leaves where the first player wins?
- What is the number of leaves where the first player loses?
- What is the total number of edges in the game tree?
- Which player do we expect to win?

[6 marks]

Note: Please ensure you choose the most efficient representation, i.e., a state representation uses the minimal number of variables, for your game tree. It is important to note that the information regarding **game turn is not part of the state** (i.e., whether it is the first or second player's turn). Also, ensure that all leaves are terminal states.

Please note that showing your working is not required for this question. However, if you wish to do so, you may include it just in case. All multiple-choice questions will be automatically graded and we will not review your working, unless you submit an appeal due to an error (such as shading incorrectly by mistake).

B. **[Alpha-beta left-to-right]** In lecture, we discussed *Alpha-beta pruning*. Consider the following minimax tree:



Suppose we traverse this tree with DFS from left to right. Shade all the link(s) that would be pruned by *alpha-beta*. Select only the links that are directly pruned by alpha-beta and not those that are indirectly pruned because they are in a subtree of a pruned link. Indicate the final value of the root node. [5 marks]

C. **[Alpha-beta right-to-left]** Suppose we traverse the minimax tree in Part (B) with DFS from right to left instead. Shade all the link(s) that would be pruned by *alpha-beta*. Select only the links that are directly pruned by alpha-beta and not those that are indirectly pruned because they are in a subtree of a pruned link. [4 marks]

Question 5: Reflections (Free marks!) [3 marks]

What are the 3 most important lessons that you think you learnt in CS2109S thus far? Explain. [3 marks]

Appendix

The following is one of the algorithms that was introduced in class that is reproduced here for your reference.

function TREE-SEARCH(*problem,frontier*) **returns** a solution, or failure
frontier ← INSERT(MAKE-NODE(INITIAL-STATE[*problem*]),*frontier*)
loop do
if *frontier* is empty **then return** failure
node ← REMOVE-FRONT(*frontier*)
if GOAL-TEST[*problem*] applied to STATE(*node*) succeeds **return** *node*
frontier ← INSERTALL(EXPAND(*node*,*problem*),*frontier*)

function GRAPH-SEARCH(*problem,frontier*) **returns** a solution, or failure

closed ← an empty set
frontier ← INSERT(MAKE-NODE(INITIAL-STATE[*problem*]),*frontier*)
loop do
if *frontier* is empty **then return** failure
node ← REMOVE-FRONT(*frontier*)
if GOAL-TEST(*problem*, STATE[*node*]) **then return** *node*
if STATE[*node*] is not in *closed* **then**
add STATE[*node*] to *closed*
frontier ← INSERTALL(EXPAND(*node*,*problem*),*frontier*)
end

function DECISION-TREE-LEARNING(*examples, attributes, default*) **returns** a decision tree

inputs: *examples*, set of examples

attributes, set of attributes

default, default value for the goal predicate

if *examples* is empty **then return** *default*

else if all *examples* have the same classification **then return** the classification

else if *attributes* is empty **then return** MAJORITY-VALUE(*examples*)

else

$best \leftarrow$ CHOOSE-ATTRIBUTE($attributes, examples$)

$tree \leftarrow$ a new decision tree with root test *best*

for each value v_i of *best* **do**

$examples_i \leftarrow \{$elements of *examples* with $best = v_i\}$

subtree $\leftarrow$ DECISION-TREE-LEARNING($examples_i$, $attributes - best$,

MAJORITY-VALUE(*examples*))

add a branch to *tree* with label v_i and subtree *subtree*

end

return *tree*

— END OF PAPER —

CS2109S—Introduction to AI and Machine Learning

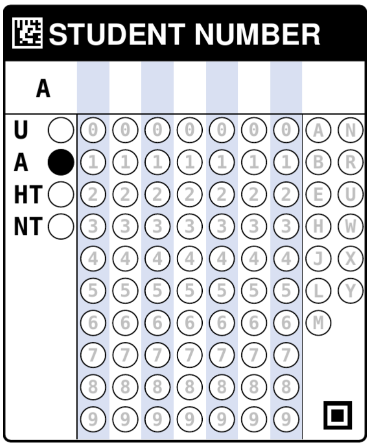
Midterm Assessment Answer Sheet

Semester 1, 2023/2024

Time allowed: 2 hours

Instructions (please read carefully):

1. Write down your **Student Number** on the right and using ink or pencil, **shade completely** the corresponding bubbles in the grid for each digit or letter. **DO NOT WRITE YOUR NAME!**
2. This answer booklet comprises **EIGHTEEN (18) pages**, including this cover page.
3. This is a <u>**OPEN-SHEET**</u> assessment. You are allow one A4-sized double-sided cheatsheet.
4. Weightage of questions is given in square brackets. The maximum attainable score is 100.
5. You are allowed to bring a calculator, but it cannot have any form of external communication capability, i.e. not Wifi- or 4G-enabled. Mobile phones and tablets are not allowed.
6. All questions must be answered in the space provided on the answer sheet; no extra sheets will be accepted as answers.
7. You are allowed to write with pencils, as long as it is legible.
8. **Marks may be deducted** for unrecognisable handwriting and/or for not shading the student number properly.
9. You must submit only the **ANSWER SHEET** and no other documents. The question set may be used as scratch paper.
10. An excerpt of the question may be provided to aid you in answering in the correct box, where applicable. It is not the exact question. You should still refer to the original question in the question booklet.



For Examiner's Use Only

Question	Marks
Q1	/ 32
Q2	/ 26
Q3	/ 24
Q4	/ 15
Q5	/ 3
Total	/100

Question 1A State representation. [2 marks]

Question 1B Invariant. [2 marks]

Question 1C Initial and goal state. [2 marks]

Question 1D Actions. [2 marks]

Question 1E Best search algorithm. [4 marks]

Question 1F Use Graph-Search or not? [3 marks]

Question 1G Initial candidate solution. [2 marks]

Question 1H Transition function. [2 marks]

Question 1I Heuristic function. [2 marks]

Question 1J New formulation for Variant #1. [6 marks]

Question 1K Best search algorithm for Variant #1. [2 marks]

Question 1L Is your TREE-SEARCH complete and/or optimal? Is the search tree finite for Variant #1? [3 marks]

Question 2A Is h_A admissible? Explain. [3 marks]

○ Admissible ○ Not admissible

Question 2B Is h_B admissible? Explain. [3 marks]

○ Admissible ○ Not admissible

Question 2C Is h_C admissible? Explain. [3 marks]

○ Admissible ○ Not admissible

Question 2D Is h_D admissible and/or consistent? Explain. [6 marks]

○ Admissible ○ Not admissible
○ Consistent ○ Not consistent

Question 2E Is h_E admissible and/or consistent? Explain. [6 marks]

◯ Admissible ◯ Not admissible

◯ Consistent ◯ Not consistent

Question 2F Choosing between h_E and h_F? Explain. [5 marks]

Question 3A The eligibility of a low income loan taker? [2 marks]

◯ Yes ◯ No ◯ Undefined

Question 3B What is the entropy? [2 marks]

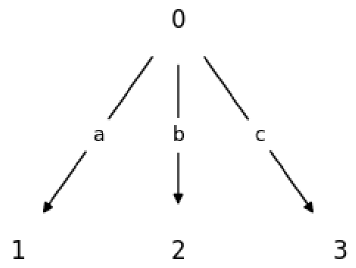
◯ $0.0 \leq \text{Entropy} < 0.2$ ◯ $0.2 \leq \text{Entropy} < 0.4$

◯ $0.4 \leq \text{Entropy} < 0.6$ ◯ $0.6 \leq \text{Entropy} < 0.8$

◯ $0.8 \leq \text{Entropy} \leq 1.0$

Show your working below (if you wish):

Question 3C One-level Tree. [6 marks]



What is node 0?

- ◯ Age ◯ Credit Rating ◯ Criminal Record ◯ Income
- ◯ no ◯ yes ◯ no/yes ◯ -

What is node 1?

- ◯ Age ◯ Credit Rating ◯ Criminal Record ◯ Income
- ◯ no ◯ yes ◯ no/yes ◯ -

What is node 2?

- ◯ Age ◯ Credit Rating ◯ Criminal Record ◯ Income
- ◯ no ◯ yes ◯ no/yes ◯ -

What is node 3?

- ◯ Age ◯ Credit Rating ◯ Criminal Record ◯ Income
- ◯ no ◯ yes ◯ no/yes ◯ -

What is edge a?

- ◯ excellent ◯ fair ◯ high ◯ low ◯ middle-aged ◯ no
- ◯ poor ◯ senior ◯ yes ◯ young ◯ -

What is edge b?

- ◯ excellent ◯ fair ◯ high ◯ low ◯ middle-aged ◯ no
- ◯ poor ◯ senior ◯ yes ◯ young ◯ -

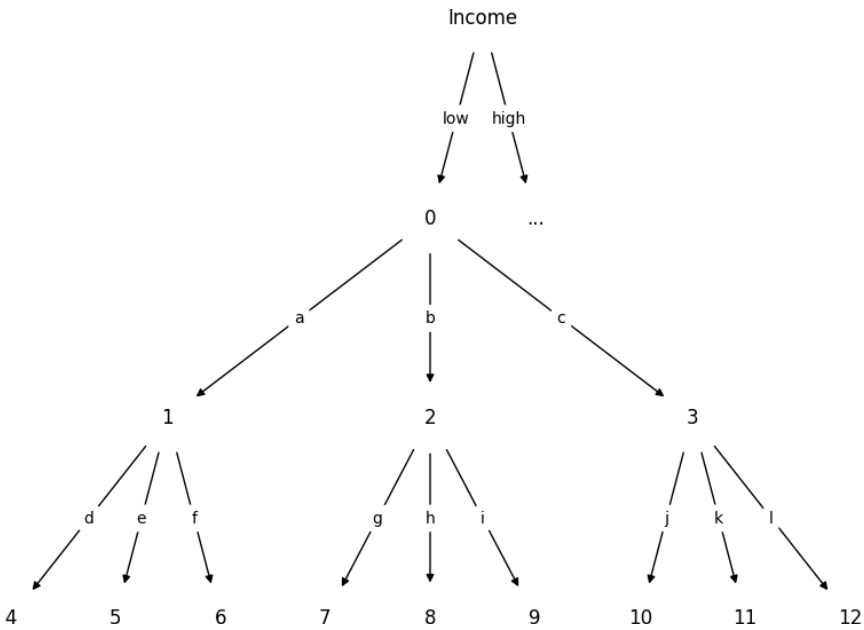
What is edge c?

- ◯ excellent ◯ fair ◯ high ◯ low ◯ middle-aged ◯ no
- ◯ poor ◯ senior ◯ yes ◯ young ◯ -

Question 3C One-level Tree. (Continued) [6 marks]

Show your working below (if you wish):

Question 3D Full Decision Tree. [6 marks]



What is node 0?

◯ Age ◯ Credit Rating ◯ Criminal Record ◯ no
◯ yes ◯ no/yes ◯ -

What is node 1?

◯ Age ◯ Credit Rating ◯ Criminal Record ◯ no
◯ yes ◯ no/yes ◯ -

What is node 2?

◯ Age ◯ Credit Rating ◯ Criminal Record ◯ no
◯ yes ◯ no/yes ◯ -

What is node 3?

◯ Age ◯ Credit Rating ◯ Criminal Record ◯ no
◯ yes ◯ no/yes ◯ -

What is node 4?

◯ Age ◯ Credit Rating ◯ Criminal Record ◯ no
◯ yes ◯ no/yes ◯ -

What is node 5?

◯ Age ◯ Credit Rating ◯ Criminal Record ◯ no
◯ yes ◯ no/yes ◯ -

What is node 6?

◯ Age ◯ Credit Rating ◯ Criminal Record ◯ no
◯ yes ◯ no/yes ◯ -

Are there any nodes from 7 onwards?

◯ Yes ◯ No

Question 3D Full Decision Tree. (Continued) [6 marks]

What is edge a?

◯ excellent ◯ fair ◯ middle-aged ◯ no ◯ poor
◯ senior ◯ yes ◯ young ◯ -

What is edge b?

◯ excellent ◯ fair ◯ middle-aged ◯ no ◯ poor
◯ senior ◯ yes ◯ young ◯ -

What is edge c?

◯ excellent ◯ fair ◯ middle-aged ◯ no ◯ poor
◯ senior ◯ yes ◯ young ◯ -

What is edge d?

◯ excellent ◯ fair ◯ middle-aged ◯ no ◯ poor
◯ senior ◯ yes ◯ young ◯ -

What is edge e?

◯ excellent ◯ fair ◯ middle-aged ◯ no ◯ poor
◯ senior ◯ yes ◯ young ◯ -

What is edge f?

◯ excellent ◯ fair ◯ middle-aged ◯ no ◯ poor
◯ senior ◯ yes ◯ young ◯ -

Question 3D Full Decision Tree. (Continued) [6 marks]

Show your working below (if you wish):

Question 3E Precision, Recall, F1. [5 marks]

Number of True Positives (TP)?

○ 0 ○ 1 ○ 2 ○ 3 ○ 4 ○ >4

Number of False Positives (FP)?

○ 0 ○ 1 ○ 2 ○ 3 ○ 4 ○ >4

Number of True Negatives(TN)?

○ 0 ○ 1 ○ 2 ○ 3 ○ 4 ○ >4

Number of False Negatives (FN)?

○ 0 ○ 1 ○ 2 ○ 3 ○ 4 ○ >4

Precision?

○ $0.0 \leq$ Precision < 0.2
○ $0.2 \leq$ Precision < 0.4
○ $0.4 \leq$ Precision < 0.6
○ $0.6 \leq$ Precision < 0.8
○ $0.8 \leq$ Precision ≤ 1.0

Recall?

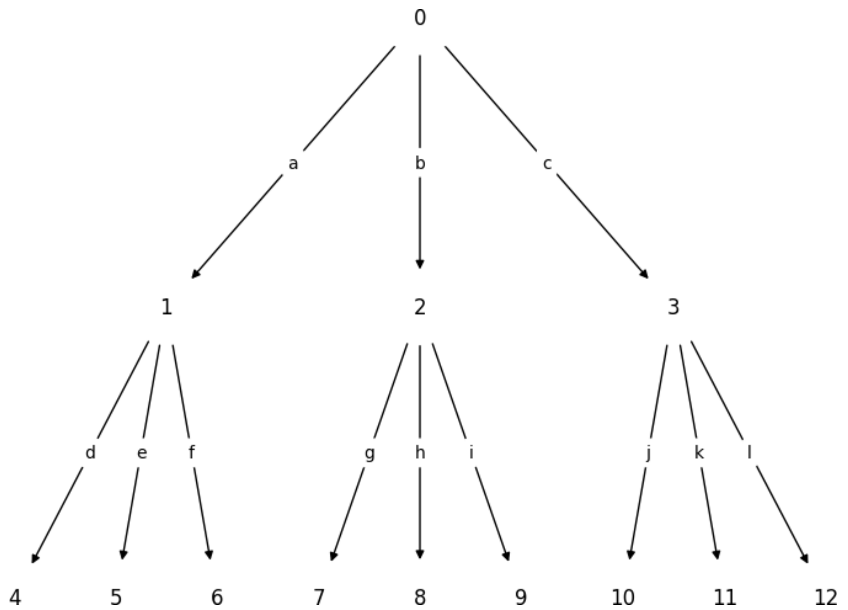
○ $0.0 \leq$ Recall < 0.2
○ $0.2 \leq$ Recall < 0.4
○ $0.4 \leq$ Recall < 0.6
○ $0.6 \leq$ Recall < 0.8
○ $0.8 \leq$ Recall ≤ 1.0

F1 Score?

○ $0.0 \leq$ F1 Score < 0.2
○ $0.2 \leq$ F1 Score < 0.4
○ $0.4 \leq$ F1 Score < 0.6
○ $0.6 \leq$ F1 Score < 0.8
○ $0.8 \leq$ F1 Score ≤ 1.0

Show your working below (if you wish):

Question 3F Min-Sample DT. [3 marks]



What is node 0?

○ Age ○ Credit Rating ○ Criminal Record ○ Income
○ no ○ yes ○ no/yes ○ -

What is node 1?

○ Age ○ Credit Rating ○ Criminal Record ○ Income
○ no ○ yes ○ no/yes ○ -

What is node 2?

○ Age ○ Credit Rating ○ Criminal Record ○ Income
○ no ○ yes ○ no/yes ○ -

What is node 3?

○ Age ○ Credit Rating ○ Criminal Record ○ Income
○ no ○ yes ○ no/yes ○ -

What is node 4?

○ Age ○ Credit Rating ○ Criminal Record ○ Income
○ no ○ yes ○ no/yes ○ -

What is node 5?

○ Age ○ Credit Rating ○ Criminal Record ○ Income
○ no ○ yes ○ no/yes ○ -

What is node 6?

○ Age ○ Credit Rating ○ Criminal Record ○ Income
○ no ○ yes ○ no/yes ○ -

Are there any nodes from 7 onwards?

○ Yes ○ No

Question 3F Min-Sample DT. (Continued) [3 marks]

What is edge a?

○ excellent ○ fair ○ high ○ low ○ middle-aged ○ no
○ poor ○ senior ○ yes ○ young ○ -

What is edge b?

○ excellent ○ fair ○ high ○ low ○ middle-aged ○ no
○ poor ○ senior ○ yes ○ young ○ -

What is edge c?

○ excellent ○ fair ○ high ○ low ○ middle-aged ○ no
○ poor ○ senior ○ yes ○ young ○ -

What is edge d?

○ excellent ○ fair ○ high ○ low ○ middle-aged ○ no
○ poor ○ senior ○ yes ○ young ○ -

What is edge e?

○ excellent ○ fair ○ high ○ low ○ middle-aged ○ no
○ poor ○ senior ○ yes ○ young ○ -

What is edge f?

○ excellent ○ fair ○ high ○ low ○ middle-aged ○ no
○ poor ○ senior ○ yes ○ young ○ -

Show your working below (if you wish):

Question 4A Game Tree. [6 marks]

What is the minimum number of variables required to maintain the game state?

◯ 1 ◯ 2 ◯ 3 ◯ 7 ◯ 8 ◯ Others

How many nodes (not including the leaves) are there in the game tree?

◯ 1	◯ 2	◯ 3	◯ 4	◯ 5	◯ 6	◯ 7
◯ 8	◯ 9	◯ 10	◯ 11	◯ 12	◯ 13	◯ 14
◯ 15	◯ 16	◯ 17	◯ 18	◯ 19	◯ 20	◯ >20

What is the number of leaves where the first player wins?

◯ 0 ◯ 1 ◯ 2 ◯ 3 ◯ 4 ◯ 5 ◯ 6 ◯ >6

What is the number of leaves where the first player loses?

◯ 0 ◯ 1 ◯ 2 ◯ 3 ◯ 4 ◯ 5 ◯ 6 ◯ >6

What is the total number of edges in the game tree?

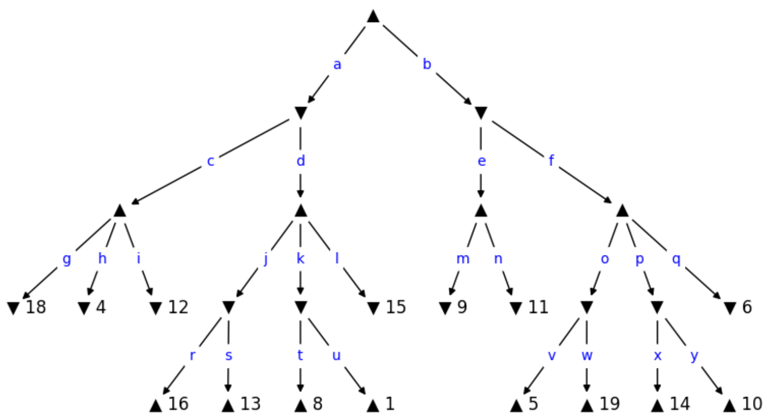
◯ 1	◯ 2	◯ 3	◯ 4	◯ 5	◯ 6	◯ 7
◯ 8	◯ 9	◯ 10	◯ 11	◯ 12	◯ 13	◯ 14
◯ 15	◯ 16	◯ 17	◯ 18	◯ 19	◯ 20	◯ >20

Which player do we expect to win?

◯ First player ◯ Second player ◯ It depends ◯ Draw

Show your working below (if you wish):

Question 4B Alpha-beta left-to-right. [5 marks]



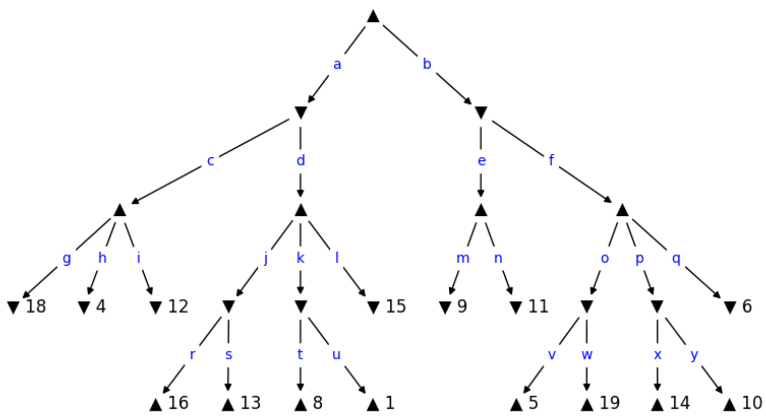
What is the root value?

○ 1 ○ 4 ○ 5 ○ 6 ○ 8 ○ 9 ○ 10 ○ 11
○ 12 ○ 13 ○ 14 ○ 15 ○ 16 ○ 18 ○ 19

Which of the following link(s) are pruned? Shade <u>all</u> that is/are true.

○ a ○ b ○ c ○ d ○ e ○ f ○ g
○ h ○ i ○ j ○ k ○ l ○ m ○ n
○ o ○ p ○ q ○ r ○ s ○ t ○ u
○ v ○ w ○ x ○ y

Question 4C Alpha-beta right-to-left. [4 marks]



Which of the following link(s) are pruned? Shade <u>all</u> that is/are true.

○ a ○ b ○ c ○ d ○ e ○ f ○ g
○ h ○ i ○ j ○ k ○ l ○ m ○ n
○ o ○ p ○ q ○ r ○ s ○ t ○ u
○ v ○ w ○ x ○ y

Question 5 3 most important lessons [3 marks]

This page is intentionally left blank.

It may be used as scratch paper.

This page is intentionally left blank.

It may be used as scratch paper.

— END OF ANSWER SHEET —

Question 1A State representation. [2 marks]

The state can be represented with fives lists, 4 of the lists l_1, l_2, l_3 and l_4 will contain the sticks used for each side of the square, while the final list l_r will contain the *currently* unused sticks. Each list will contain the integers corresponding to the lengths of the sticks used for each side.

Question 1B Invariant. [2 marks]

The total length of the sticks in each list l_1, l_2, l_3 and l_4 should not exceed one-fourth of the sum of all the stick lengths, represented as $\frac{1}{4}\sum_{i=0}^{n} s_i$. The union of all the lists is equivalent to the original list of sticks $[s_0, s_1, ..., s_n]$.

Question 1C Initial and goal state. [2 marks]

Initial State: Five lists: l_1=[], l_2=[], l_3=[], l_4=[], l_r= $[s_0, s_1, ..., s_n]$.

Goal State: Where $T = \sum_i^n s_i$, sum(l_1) = sum(l_3) = sum(l_3) = sum(l_4) = $\frac{T}{4}$. l_r = [].

Question 1D Actions. [2 marks]

In each step, the first edge form l_r is moved into one of l_1, l_2, l_3 and l_4.

Question 1E Best search algorithm. [4 marks]

We choose DFS because it will find a solution if one exists (i.e. it is complete) and constant space usage. We avoid BFS due to its higher space requirements. Similarly, IDS, while functional, incurs unnecessary overhead. DLS is not suitable as we do not prema facie know the correct depth limit.

[+1] each for brief explanations for each algorithm.
Note that the correct answer here depends on depends on Parts (A) to (D).

Question 1F Use Graph-Search or not? [3 marks]

The answer depends on the details of the problem formulation. If all stick lengths are distinct ($s_i \neq s_j$ for all i, j where $i \neq j$), then there are no repeated states, and there is no necessity to employ GRAPH-SEARCH. However, if the box contains sticks with equal lengths ($s_i = s_j$ where $i \neq j$), this can lead to the generation of repeated states. In such cases, utilizing GRAPH-SEARCH allows us to avoid expanding the same states. However, to implement GRAPH-SEARCH efficiently, we will either have to ensure that the lists are sorted. This advantage comes at the cost of increased memory and time required to maintain a visited state set.

[+2] No need to use GRAPH-SEARCH and valid reason.

Question 1G Initial candidate solution. [2 marks]

A possible candidate solution is a random allocation of the sticks into 4 lists representing the 4 sides $[s_0, s_1 ...]$, $[s_2, s_3 ...]$, $[s_4, s_5 ...]$, $[s_6, s_7 ... s_n]$. Each stick is allocated to one of the lists.

Question 1H Transition function. [2 marks]

The transition function involves moving a stick from one list to another. For instance, if the current state is $\{[s_0, s_1...], [s_2, s_3...], [s_4, s_5...], [s_6, s_7..., s_n]\}$, a neighboring state could be $\{[s_1...], [s_0, s_2, s_3...], [s_4, s_5...], [s_6, s_7...s_n]\}$.

If a student suggests swapping 2 sticks as the transition function, it is incorrect because we cannot be sure that swapping sticks along will necessarily allow the state representation to transition to the goal state, even if such a state exists, depending on the initial configuration.

Question 1I Heuristic function. [2 marks]

We define:

$$h = \sum_{i \in \{0,1,2,3\}} \delta(l_i)$$

where $\delta(l_i)$ is an indicator function such that $\delta(l_i) = 1$ when l_i is equal to l_{avg}, and $\delta(l_i) = 0$ otherwise. Here, l_{avg} represents one-quarter of the total length of all the sticks provided by Ben Bitdiddle.

Upon reaching the goal state, we achieve $h = 4$, which is greater than any non-goal state.

Question 1J New formulation for Variant #1. [6 marks]

However, as implied by Part (K) below, formulating the problem carefully so that the search will complete requires some thought. The most straightforward way to do so is to add a new list l_{unused}. In each step, instead of moving a stick from l_r into l_1, l_2, l_3 and l_4, we can also move it into l_{unused}.

-2 marks for formulation that is correct, but that cannot guarantee termination.

Question 1K Best search algorithm for Variant #1. [2 marks]

BFS, because we need to use the minimal number of sticks.

IDS is also a viable choice since it enables us to find the optimal solution. Compared to BFS, it saves memory but node in the upper level of search tree are generated multiple times.

DFS is not recommended because it doesn't guarantee finding the minimal solution.

DLS is likewise not recommended for the same reason; it doesn't guarantee a minimal solution.

Question 1L Is your TREE-SEARCH complete and/or optimal? Is the search tree finite for Variant #1? [3 marks]

If the answer to Part (K) is either BFS or IDS, then TREE-SEARCH is both complete and optimal.

If the answer for Part (K) is DFS then the TREE-SEARCH is complete but not optimal.

If the answer for Part (K) is DLS then the TREE-SEARCH is not complete and not optimal.

Whether TREE-SEARCH will always terminate will depend on whether the search is set up carefully such that each stick in the box is used at most once. If the same stick can be used multiple times then the search is not guaranteed to terminate.

Question 2A Is h_A admissible? Explain. [3 marks]

○ Admissible ● Not admissible

Consider a scenario where the snake's current position is $\{(4,2),(4,3)\}$, and the goal position is $\{(4,3),(4,4)\}$. In this case, $h_A = 2$, which is greater than the optimal cost, $h^* = 1$.

Question 2B Is h_B admissible? Explain. [3 marks]

○ Admissible ● Not admissible

Consider a situation where the snake's current position is $\{(4,3),(3,3)\}$, and the goal position is $\{(4,3),(4,4)\}$. In this case, $h_B = 2$, which exceeds the optimal cost, $h^* = 1$.

Question 2C Is h_C admissible? Explain. [3 marks]

● Admissible ○ Not admissible

Admissible. The tail can move at most 1 step towards the goal tail in every move. If we consider a relaxed version of the Snake game where the only objective is to move the tail to its goal position, the real cost for this relaxed problem is indeed h_C. Therefore, h_C is admissible.

Question 2D Is h_D admissible and/or consistent? Explain. [6 marks]

○ Admissible ● Not admissible

○ Consistent ● Not consistent

Not admissible. Consider the case where the current position of snake is $\{(4,3),(3,3)\}$ and goal is $\{(4,3),(4,4)\}$. Position of apple is $(4,4)$. In this case, $h_D = 1+1 > h^* = 1$.

Not consistent. After the snake eats the apple, all distances between the apple and other positions are reset to zero. Thus h_D holds a value of 0 for the goal. According to the above scenario (snake current position: $\{(4,3),(3,3)\}$; snake goal position: $\{(4,3),(4,4)\}$; apple position: $(4,4)$), the next node can be the goal node by rotating 90°. $h_D = 1+1 > c(n,a,n') + h_D(n') = 1+0$.

Question 2E Is h_E admissible and/or consistent? Explain. [6 marks]

● Admissible ○ Not admissible

○ Consistent ● Not consistent

Admissible. $MD(h,ap)$ can decrease by a maximum of 2 for each move. In order to eat apple, we want to achieve $MD(h,ap) = 0$. The value $MD(h,ap)/2$ is an underestimate of the actual number of steps needed to move from the current head position to the apple. The snake's head must also advance towards the goal head position, and the same reasoning applies to $MD(ap,gh)$. Consequently, we can conclude that the heuristic function h_E is admissible.

Not consistent. Consider the case where the current position of snake is $\{(4,0),(4,1)\}$ and goal is $\{(4,3),(4,4)\}$. Position of apple is $(4,2)$. In this case, for the current node n, $h_E(n) = (1+2)/2$. By moving 1 cell to the right, a successor n' of n can be generated. The position of snake becomes $\{(4,1),(4,2)\}$ and the apple can be eaten, then all distances between the apple and other positions are reset to zero ($h_E(n') = 0$). Thus $h_E(n) = 1.5 > c(n,a,n') + h_E(n') = 1+0$.

Question 2F Choosing between h_E and h_F? Explain. [5 marks]

Consider the case where the current position of snake is $\{(4,3),(4,2)\}$ and goal is $\{(4,3),(4,4)\}$. Position of apple is $(4,4)$. In this case, $h_G = 2+0 > h^* = 1$.

Since h_F is not admissible, so we choose h_E.

Question 3A The eligibility of a low income loan taker? [2 marks]

○ Yes ○ No ● Undefined

This question tests the student's ability to reason about a decision tree.

According to the given decision tree, we can't make a decision solely from the income.

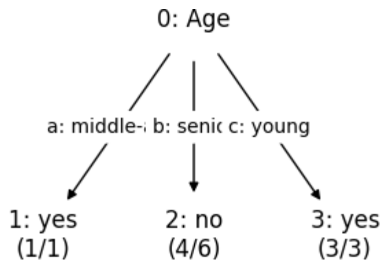
Question 3B What is the entropy? [2 marks]

- ◯ $0.0 \leq \text{Entropy} < 0.2$
- ◯ $0.2 \leq \text{Entropy} < 0.4$
- ◯ $0.4 \leq \text{Entropy} < 0.6$
- ◯ $0.6 \leq \text{Entropy} < 0.8$
- ● $0.8 \leq \text{Entropy} \leq 1.0$

Show your working below (if you wish):

$$\begin{aligned} Entropy &= I(\frac{6}{10}, \frac{4}{10}) \\ &= -\frac{6}{10}\log_2(\frac{6}{10}) - \frac{4}{10}\log_2(\frac{4}{10}) \\ &= 0.971 \end{aligned}$$

Question 3C One-level Tree. [6 marks]



Note: (X/Y) in the leaf nodes represent the number of samples belonging to the majority class over the total number of samples across all classes. To illustrate, a leaf node labeled as "no (4/6)" indicates that out of 6 samples, 4 of them belong to the "no" class.

What is node 0?

● Age ○ Credit Rating ○ Criminal Record ○ Income
○ no ○ yes ○ no/yes ○ -

What is node 1?

○ Age ○ Credit Rating ○ Criminal Record ○ Income
○ no ● yes ○ no/yes ○ -

What is node 2?

○ Age ○ Credit Rating ○ Criminal Record ○ Income
● no ○ yes ○ no/yes ○ -

What is node 3?

○ Age ○ Credit Rating ○ Criminal Record ○ Income
○ no ● yes ○ no/yes ○ -

What is edge a?

○ excellent ○ fair ○ high ○ low ● middle-aged ○ no
○ poor ○ senior ○ yes ○ young ○ -

What is edge b?

○ excellent ○ fair ○ high ○ low ○ middle-aged ○ no
○ poor ● senior ○ yes ○ young ○ -

What is edge c?

○ excellent ○ fair ○ high ○ low ○ middle-aged ○ no
○ poor ○ senior ○ yes ● young ○ -

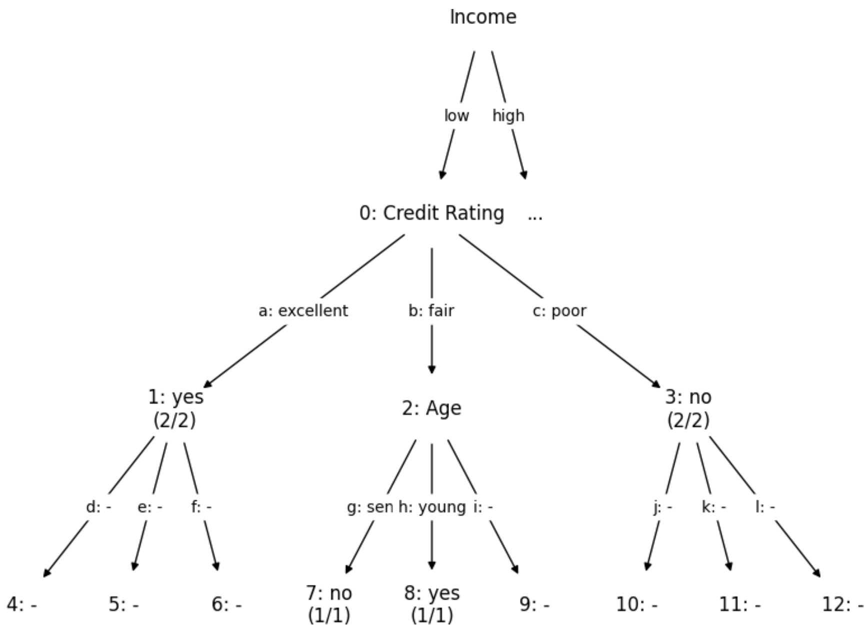
Question 3C One-level Tree. (Continued) [6 marks]

Show your working below (if you wish):

$$
\begin{aligned}
Remainder(Age) &= \frac{1}{10}\left(I(\frac{1}{1})\right) + \frac{6}{10}\left(I(\frac{4}{6},\frac{2}{6})\right) + \frac{3}{10}\left(I(\frac{3}{3})\right) \\
&= \frac{1}{10}(0.000) + \frac{6}{10}(0.918) + \frac{3}{10}(0.000) \\
&= 0.000 + 0.551 + 0.000 \\
&= 0.551 \\
Remainder(CreditRating) &= \frac{5}{10}\left(I(\frac{4}{5},\frac{1}{5})\right) + \frac{3}{10}\left(I(\frac{2}{3},\frac{1}{3})\right) + \frac{2}{10}\left(I(\frac{2}{2})\right) \\
&= \frac{5}{10}(0.722) + \frac{3}{10}(0.918) + \frac{2}{10}(0.000) \\
&= 0.361 + 0.275 + 0.000 \\
&= 0.636 \\
Remainder(Income) &= \frac{6}{10}\left(I(\frac{3}{6},\frac{3}{6})\right) + \frac{4}{10}\left(I(\frac{1}{4},\frac{3}{4})\right) \\
&= \frac{6}{10}(1.000) + \frac{4}{10}(0.811) \\
&= 0.600 + 0.325 \\
&= 0.925 \\
Remainder(CriminalRecord) &= \frac{4}{10}\left(I(\frac{3}{4},\frac{1}{4})\right) + \frac{6}{10}\left(I(\frac{3}{6},\frac{3}{6})\right) \\
&= \frac{4}{10}(0.811) + \frac{6}{10}(1.000) \\
&= 0.325 + 0.600 \\
&= 0.925
\end{aligned}
$$

We split based on "Age" as it results in the smallest remaining entropy.

Question 3D Full Decision Tree. [6 marks]



What is node 0?

○ Age ● Credit Rating ○ Criminal Record ○ no
○ yes ○ no/yes ○ -

What is node 1?

○ Age ○ Credit Rating ○ Criminal Record ○ no
● yes ○ no/yes ○ -

What is node 2?

● Age ○ Credit Rating ○ Criminal Record ○ no
○ yes ○ no/yes ○ -

What is node 3?

○ Age ○ Credit Rating ○ Criminal Record ● no
○ yes ○ no/yes ○ -

What is node 4?

○ Age ○ Credit Rating ○ Criminal Record ○ no
○ yes ○ no/yes ● -

What is node 5?

○ Age ○ Credit Rating ○ Criminal Record ○ no
○ yes ○ no/yes ● -

What is node 6?

○ Age ○ Credit Rating ○ Criminal Record ○ no
○ yes ○ no/yes ● -

Are there any nodes from 7 onwards?

● Yes ○ No

Question 3D Full Decision Tree. (Continued) [6 marks]

What is edge a?

- ● excellent ○ fair ○ middle-aged ○ no ○ poor
- ○ senior ○ yes ○ young ○ -

What is edge b?

- ○ excellent ● fair ○ middle-aged ○ no ○ poor
- ○ senior ○ yes ○ young ○ -

What is edge c?

- ○ excellent ○ fair ○ middle-aged ○ no ● poor
- ○ senior ○ yes ○ young ○ -

What is edge d?

- ○ excellent ○ fair ○ middle-aged ○ no ○ poor
- ○ senior ○ yes ○ young ● -

What is edge e?

- ○ excellent ○ fair ○ middle-aged ○ no ○ poor
- ○ senior ○ yes ○ young ● -

What is edge f?

- ○ excellent ○ fair ○ middle-aged ○ no ○ poor
- ○ senior ○ yes ○ young ● -

Question 3D Full Decision Tree. (Continued) [6 marks]

Show your working below (if you wish):

$$
\begin{aligned}
Remainder(CreditRating) &= \frac{2}{6}\left(I(\frac{2}{2})\right) + \frac{2}{6}\left(I(\frac{1}{2},\frac{1}{2})\right) + \frac{2}{6}\left(I(\frac{2}{2})\right) \\
&= \frac{2}{6}(0.000) + \frac{2}{6}(1.000) + \frac{2}{6}(0.000) \\
&= 0.000 + 0.333 + 0.000 \\
&= 0.333 \\
Remainder(Age) &= \frac{1}{6}\left(I(\frac{1}{1})\right) + \frac{4}{6}\left(I(\frac{1}{4},\frac{3}{4})\right) + \frac{1}{6}\left(I(\frac{1}{1})\right) \\
&= \frac{1}{6}(0.000) + \frac{4}{6}(0.811) + \frac{1}{6}(0.000) \\
&= 0.000 + 0.541 + 0.000 \\
&= 0.541 \\
Remainder(CriminalRecord) &= \frac{3}{6}\left(I(\frac{2}{3},\frac{1}{3})\right) + \frac{3}{6}\left(I(\frac{2}{3},\frac{1}{3})\right) \\
&= \frac{3}{6}(0.918) + \frac{3}{6}(0.918) \\
&= 0.459 + 0.459 \\
&= 0.918
\end{aligned}
$$

We split based on "Credit Rating" as it results in the smallest remaining entropy.

If Credit Rating = "fair", then we split again:

	Age	**Criminal Record**	**Loan**
4	senior	no	no
9	young	yes	yes

Based on the priority order, we should split based on "Age".

Question 3E Precision, Recall, F1. [5 marks]

Number of True Positives (TP)?

- ○ 0 ○ 1 ○ 2 ● 3 ○ 4 ○ >4

Number of False Positives (FP)?

- ● 0 ○ 1 ○ 2 ○ 3 ○ 4 ○ >4

Number of True Negatives(TN)?

- ○ 0 ○ 1 ○ 2 ○ 3 ● 4 ○ >4

Number of False Negatives (FN)?

- ○ 0 ○ 1 ○ 2 ● 3 ○ 4 ○ >4

Precision?

- ○ $0.0 \leq \text{Precision} < 0.2$
- ○ $0.2 \leq \text{Precision} < 0.4$
- ○ $0.4 \leq \text{Precision} < 0.6$
- ○ $0.6 \leq \text{Precision} < 0.8$
- ● $0.8 \leq \text{Precision} \leq 1.0$

Recall?

- ○ $0.0 \leq \text{Recall} < 0.2$
- ○ $0.2 \leq \text{Recall} < 0.4$
- ● $0.4 \leq \text{Recall} < 0.6$
- ○ $0.6 \leq \text{Recall} < 0.8$
- ○ $0.8 \leq \text{Recall} \leq 1.0$

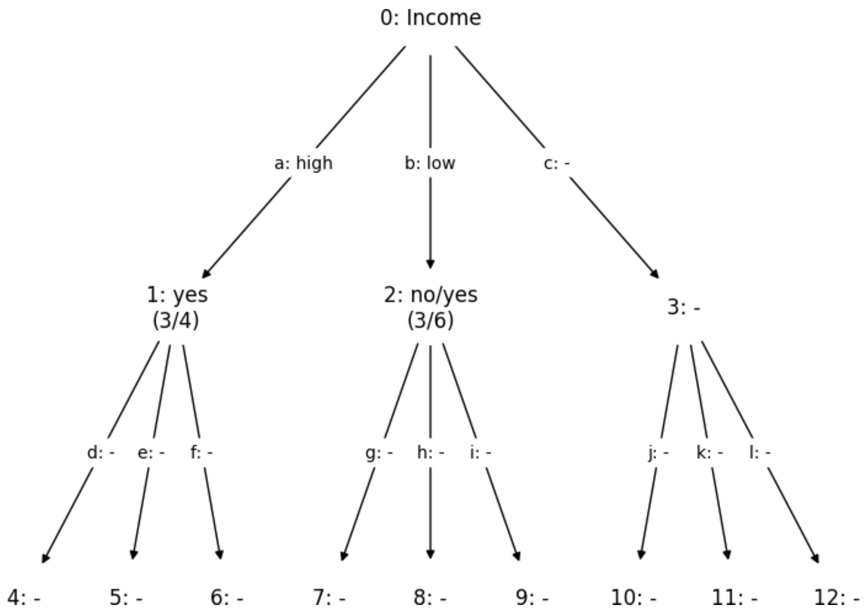
F1 Score?

- ○ $0.0 \leq \text{F1 Score} < 0.2$
- ○ $0.2 \leq \text{F1 Score} < 0.4$
- ○ $0.4 \leq \text{F1 Score} < 0.6$
- ● $0.6 \leq \text{F1 Score} < 0.8$
- ○ $0.8 \leq \text{F1 Score} \leq 1.0$

Show your working below (if you wish):

$$
\begin{aligned}
Precision &= \frac{TP}{TP+FP} \\
&= \frac{3}{3+0} \\
&= 1.000 \\
Recall &= \frac{TP}{TP+FN} \\
&= \frac{3}{3+3} \\
&= 0.500 \\
F1 &= \frac{2}{\frac{1}{P}+\frac{1}{R}} \\
&= \frac{2}{\frac{1}{1.000}+\frac{1}{0.500}} \\
&= 0.667
\end{aligned}
$$

Question 3F Min-Sample DT. [3 marks]



What is node 0?

○ Age ○ Credit Rating ○ Criminal Record ● Income
○ no ○ yes ○ no/yes ○ -

What is node 1?

○ Age ○ Credit Rating ○ Criminal Record ○ Income
○ no ● yes ○ no/yes ○ -

What is node 2?

○ Age ○ Credit Rating ○ Criminal Record ○ Income
○ no ○ yes ● no/yes ○ -

What is node 3?

○ Age ○ Credit Rating ○ Criminal Record ○ Income
○ no ○ yes ○ no/yes ● -

What is node 4?

○ Age ○ Credit Rating ○ Criminal Record ○ Income
○ no ○ yes ○ no/yes ● -

What is node 5?

○ Age ○ Credit Rating ○ Criminal Record ○ Income
○ no ○ yes ○ no/yes ● -

What is node 6?

○ Age ○ Credit Rating ○ Criminal Record ○ Income
○ no ○ yes ○ no/yes ● -

Are there any nodes from 7 onwards?

○ Yes ● No

Question 3F Min-Sample DT. (Continued) [3 marks]

What is edge a?

- ○ excellent ○ fair ● high ○ low ○ middle-aged ○ no
- ○ poor ○ senior ○ yes ○ young ○ -

What is edge b?

- ○ excellent ○ fair ○ high ● low ○ middle-aged ○ no
- ○ poor ○ senior ○ yes ○ young ○ -

What is edge c?

- ○ excellent ○ fair ○ high ○ low ○ middle-aged ○ no
- ○ poor ○ senior ○ yes ○ young ● -

What is edge d?

- ○ excellent ○ fair ○ high ○ low ○ middle-aged ○ no
- ○ poor ○ senior ○ yes ○ young ● -

What is edge e?

- ○ excellent ○ fair ○ high ○ low ○ middle-aged ○ no
- ○ poor ○ senior ○ yes ○ young ● -

What is edge f?

- ○ excellent ○ fair ○ high ○ low ○ middle-aged ○ no
- ○ poor ○ senior ○ yes ○ young ● -

Show your working below (if you wish):

Question 4A Game Tree. [6 marks]

What is the minimum number of variables required to maintain the game state?

○ 1 ○ 2 ● 3 ○ 7 ○ 8 ○ Others

How many nodes (not including the leaves) are there in the game tree?

○ 1 ○ 2 ○ 3 ○ 4 ○ 5 ○ 6 ○ 7
○ 8 ● 9 ○ 10 ○ 11 ○ 12 ○ 13 ○ 14
○ 15 ○ 16 ○ 17 ○ 18 ○ 19 ○ 20 ○ >20

What is the number of leaves where the first player wins?

○ 0 ● 1 ○ 2 ○ 3 ○ 4 ○ 5 ○ 6 ○ >6

What is the number of leaves where the first player loses?

○ 0 ● 1 ○ 2 ○ 3 ○ 4 ○ 5 ○ 6 ○ >6

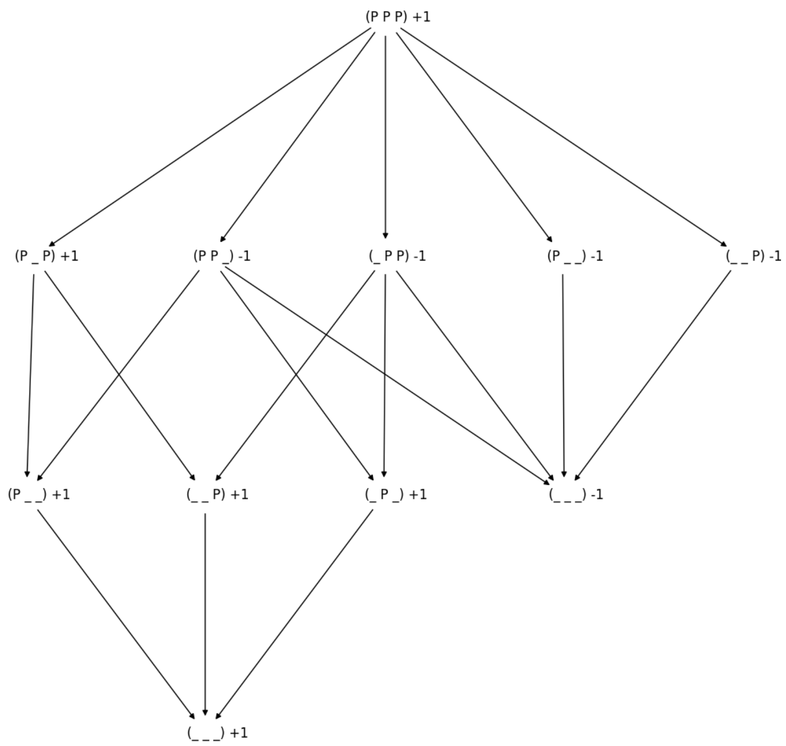
What is the total number of edges in the game tree?

○ 1 ○ 2 ○ 3 ○ 4 ○ 5 ○ 6 ○ 7
○ 8 ○ 9 ○ 10 ○ 11 ○ 12 ○ 13 ○ 14
○ 15 ○ 16 ○ 17 ● 18 ○ 19 ○ 20 ○ >20

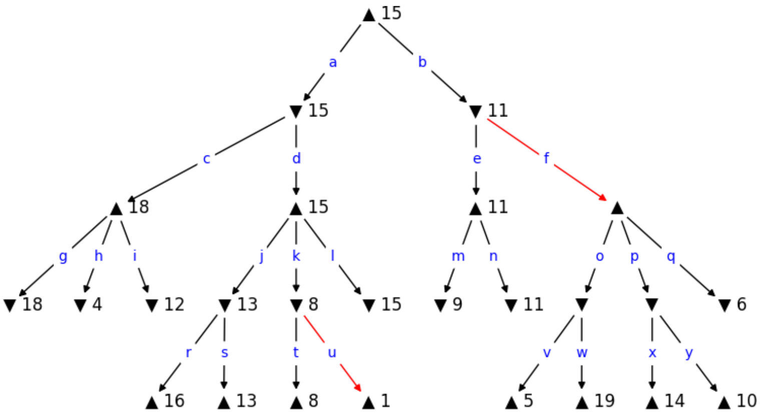
Which player do we expect to win?

● First player ○ Second player ○ It depends ○ Draw

Show your working below (if you wish):



Question 4B Alpha-beta left-to-right. [5 marks]



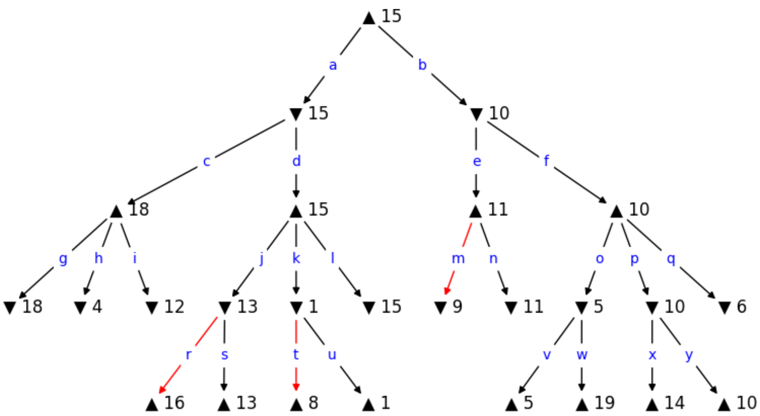
What is the root value?

○ 1 ○ 4 ○ 5 ○ 6 ○ 8 ○ 9 ○ 10 ○ 11
○ 12 ○ 13 ○ 14 ● 15 ○ 16 ○ 18 ○ 19

Which of the following link(s) are pruned? Shade <u>all</u> that is/are true.

○ a ○ b ○ c ○ d ○ e ● f ○ g
○ h ○ i ○ j ○ k ○ l ○ m ○ n
○ o ○ p ○ q ○ r ○ s ○ t ● u
○ v ○ w ○ x ○ y

Question 4C Alpha-beta right-to-left. [4 marks]



Which of the following link(s) are pruned? Shade <u>all</u> that is/are true.

○ a ○ b ○ c ○ d ○ e ○ f ○ g
○ h ○ i ○ j ○ k ○ l ● m ○ n
○ o ○ p ○ q ● r ○ s ● t ○ u
○ v ○ w ○ x ○ y

Question 5 3 most important lessons [3 marks]

There are no right answers here. Students will get credit for 3 well-explained learning points that are reasonable and justified. Student needs to put in *some* effort. Clearly if the student makes a patently false statement, marks will be deducted.