Question 1: Show the operation of A* search.

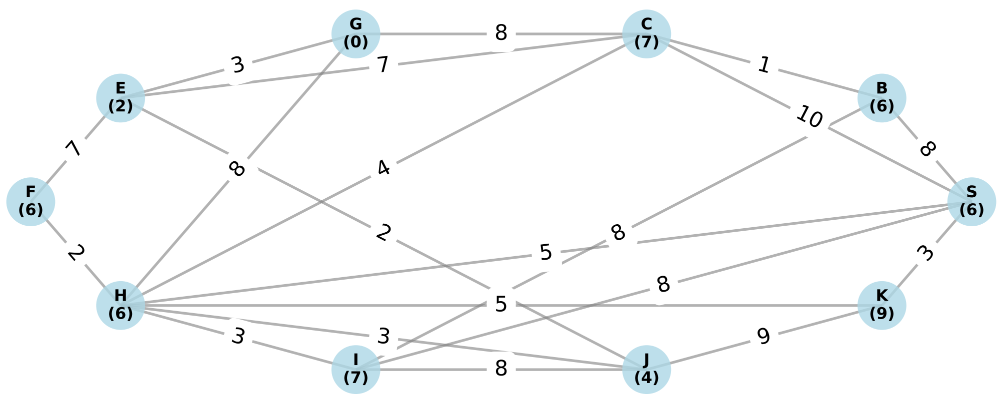
In the graph below you must show the operation of a search algorithm, starting at node S and searching for node G.

You must show all work in a table showing the node expanded and the search fringe.

In case of a tie between nodes, choose the one that comes earlier in the alphabet.

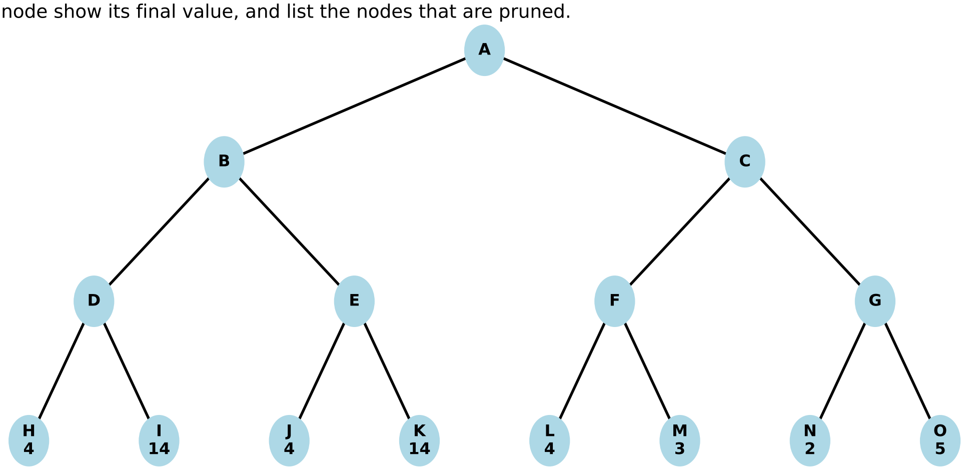
The heuristic value (the estimated distance to the goal) is the integer in each node.

Graph:



Question 2: Perform alpha-beta minimax on the following tree.

For each node show its final value, and list the nodes that are pruned.



Question 3: Draw a decision tree for the problem of deciding whether to play tennis or not.

You are given the following table of data, which has three features: the Outlook, the Temperature and the Wind, and must draw a decision tree that classifies the positive and negative instances.

Show your computations for the information gain for each node in the tree.

Day	Outlook	Temp	Wind	Play Tennis
1	Rain	Mild	Weak	Yes
2	Sunny	Mild	Weak	No
3	Overcast	Mild	Strong	No
4	Sunny	Cool	Weak	No
5	Overcast	Cool	Strong	Yes
6	Sunny	Hot	Strong	Yes
7	Rain	Hot	Strong	Yes
8	Overcast	Hot	Strong	Yes

Question 4: First-Order Logic Translation

You are given the predicates Person(x), which is true if x is a person, Knows(x, y), which is true if person x knows person y, Likes(x, y), which is true if person x likes person y, Friend(x, y), which is true if person x is a friend of person y.

a. Use the predicates to translate the following sentences into first-order logic:

1. People like those who know them.
2. Everyone has at least one friend.
3. People who know each other are friends.

b. Translate the following sentences from first-order logic into English:

1. ∀x (Person(x) ⇒ ∃y (Person(y) ∧ Knows(x, y) ∧ Likes(x, y)))
2. ∃x (Person(x) ∧ ∀y (Person(y) ∧ Knows(x, y) ⇒ Likes(y, x)))
3. ∀x ∀y (Person(x) ∧ Person(y) ∧ Friend(x, y) ⇒ ∃z (Person(z) ∧ Knows(x, z) ∧ Knows(y, z)))

Question 1: Show the operation of A* search.

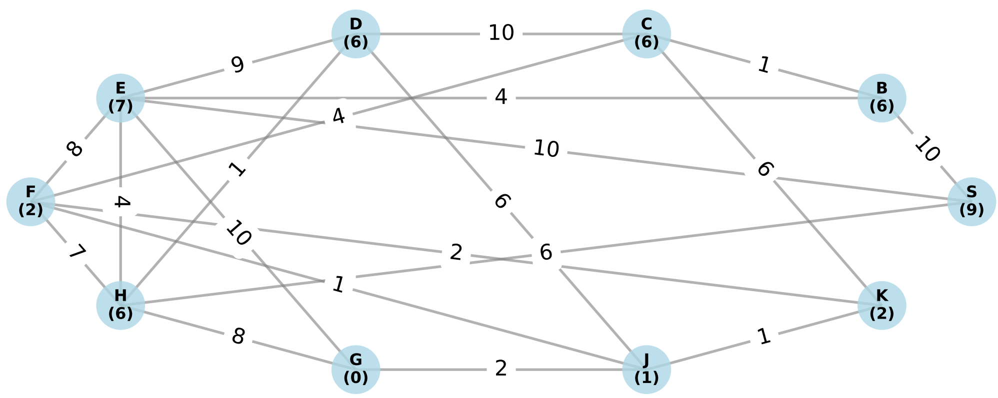
In the graph below you must show the operation of a search algorithm, starting at node S and searching for node G.

You must show all work in a table showing the node expanded and the search fringe.

In case of a tie between nodes, choose the one that comes earlier in the alphabet.

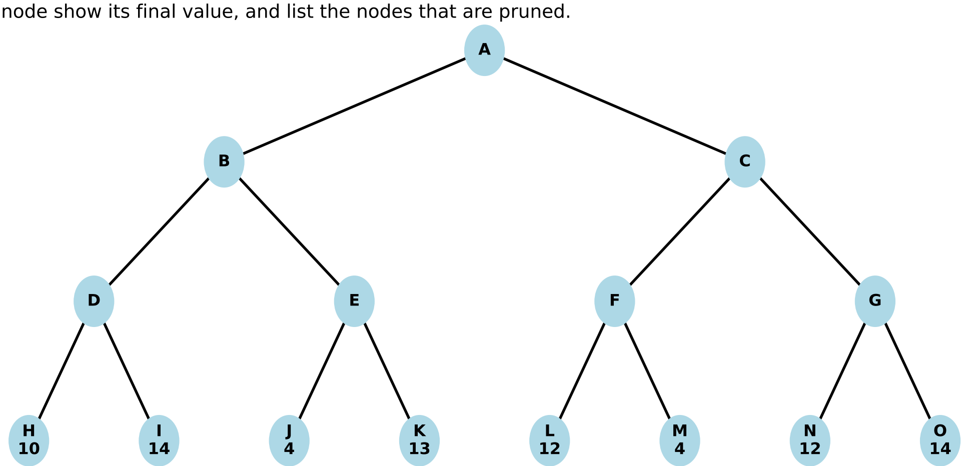
The heuristic value (the estimated distance to the goal) is the integer in each node.

Graph:



Question 2: Perform alpha-beta minimax on the following tree.

For each node show its final value, and list the nodes that are pruned.



Question 3: Draw a decision tree for the problem of deciding whether to play tennis or not.

You are given the following table of data, which has three features: the Outlook, the Temperature and the Wind, and must draw a decision tree that classifies the positive and negative instances.

Show your computations for the information gain for each node in the tree.

Day	Outlook	Temp	Wind	Play Tennis
1	Overcast	Mild	Strong	No
2	Rain	Hot	Strong	No
3	Overcast	Cool	Strong	Yes
4	Sunny	Hot	Strong	Yes
5	Sunny	Mild	Strong	Yes
6	Overcast	Hot	Weak	Yes
7	Overcast	Cool	Weak	No
8	Rain	Hot	Weak	No

Question 4: First-Order Logic Translation

You are given the predicates Book(x), which is true if x is a book, Author(x, y), which is true if book x is written by author y, Popular(x), which is true if x is a popular book, Reads(x, y), which is true if person x reads book y.

a. Use the predicates to translate the following sentences into first-order logic:

1. Popular books are read by many people.
2. People who read books by the same author like similar books.
3. Every book has an author.

b. Translate the following sentences from first-order logic into English:

1. $\forall x\ (Book(x) \Rightarrow \exists y\ (Author(x, y) \land Popular(y)))$
2. $\exists x\ (Book(x) \land \forall y\ (Reads(y, x) \Rightarrow Popular(x)))$
3. $\forall x\ \forall y\ (Book(x) \land Author(x, y) \Rightarrow \exists z\ (Reads(z, x) \land \exists w\ (Book(w) \land Author(w, y) \land Reads(z, w))))$

Question 1: Show the operation of A* search.

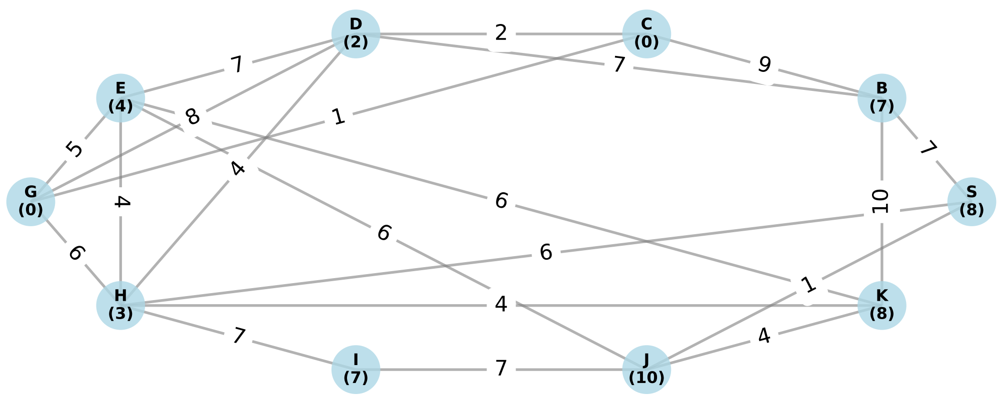
In the graph below you must show the operation of a search algorithm, starting at node S and searching for node G.

You must show all work in a table showing the node expanded and the search fringe.

In case of a tie between nodes, choose the one that comes earlier in the alphabet.

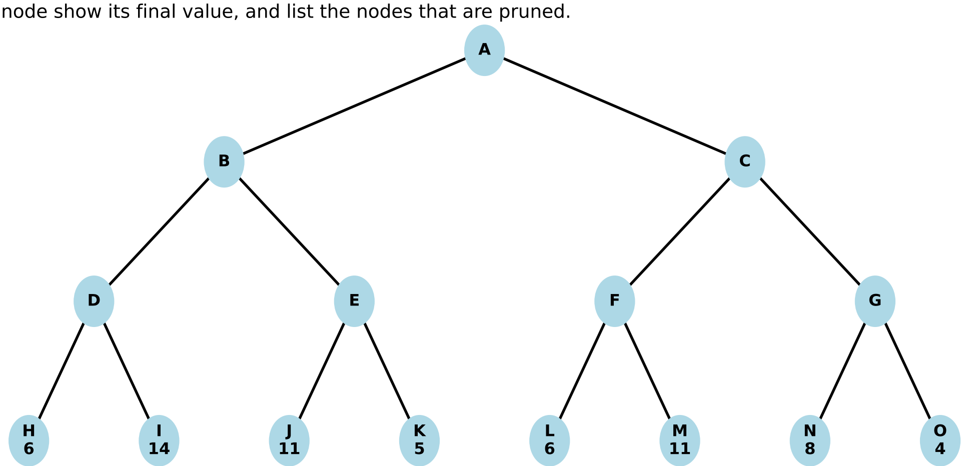
The heuristic value (the estimated distance to the goal) is the integer in each node.

Graph:



Question 2: Perform alpha-beta minimax on the following tree.

For each node show its final value, and list the nodes that are pruned.



Question 3: Draw a decision tree for the problem of deciding whether to play tennis or not.

You are given the following table of data, which has three features: the Outlook, the Temperature and the Wind, and must draw a decision tree that classifies the positive and negative instances.

Show your computations for the information gain for each node in the tree.

Day	Outlook	Temp	Wind	Play Tennis
1	Rain	Hot	Weak	Yes
2	Sunny	Cool	Strong	Yes
3	Sunny	Hot	Strong	No
4	Rain	Cool	Weak	Yes
5	Overcast	Mild	Strong	No
6	Overcast	Hot	Weak	No
7	Sunny	Cool	Weak	Yes
8	Rain	Mild	Strong	Yes

Question 4: First-Order Logic Translation

You are given the predicates Hospital(x), which is true if x is a hospital, Doctor(x), which is true if x is a doctor, WorksAt(x, y), which is true if doctor x works at hospital y, Specialist(x), which is true if x is a specialist.

a. Use the predicates to translate the following sentences into first-order logic:

1. Every hospital has doctors.
2. Specialists work at hospitals that have other specialists.
3. There is a hospital where all doctors are specialists.

b. Translate the following sentences from first-order logic into English:

1. $\forall x\ (Hospital(x) \Rightarrow \exists y\ (Doctor(y) \land WorksAt(y, x) \land Specialist(y)))$
2. $\exists x\ (Doctor(x) \land Specialist(x) \land \forall y\ (Hospital(y) \land WorksAt(x, y) \Rightarrow Specialist(x)))$
3. $\forall x\ \forall y\ (Doctor(x) \land Hospital(y) \land WorksAt(x, y) \Rightarrow \exists z\ (Doctor(z) \land WorksAt(z, y) \land \neg(x=z)))$

Question 1: Show the operation of A* search.

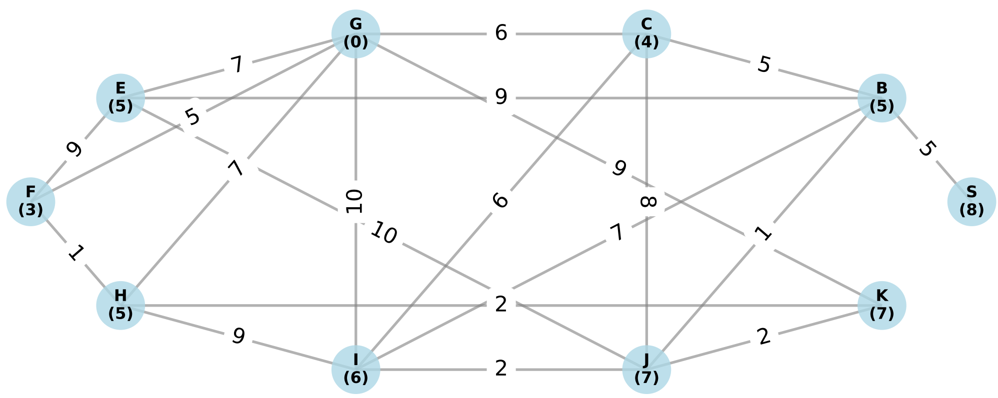
In the graph below you must show the operation of a search algorithm, starting at node S and searching for node G.

You must show all work in a table showing the node expanded and the search fringe.

In case of a tie between nodes, choose the one that comes earlier in the alphabet.

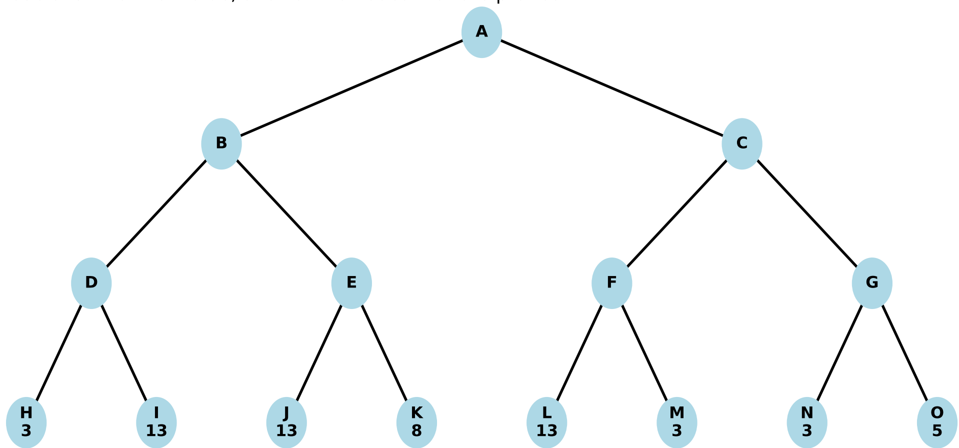
The heuristic value (the estimated distance to the goal) is the integer in each node.

Graph:



Question 2: Perform alpha-beta minimax on the following tree.

For each node show its final value, and list the nodes that are pruned.



Question 3: Draw a decision tree for the problem of deciding whether to play tennis or not.

You are given the following table of data, which has three features: the Outlook, the Temperature and the Wind, and must draw a decision tree that classifies the positive and negative instances.

Show your computations for the information gain for each node in the tree.

Day	Outlook	Temp	Wind	Play Tennis
1	Rain	Mild	Strong	Yes
2	Sunny	Cool	Strong	Yes
3	Rain	Hot	Weak	Yes
4	Rain	Cool	Strong	No
5	Overcast	Mild	Weak	Yes
6	Sunny	Hot	Weak	Yes
7	Overcast	Cool	Strong	No
8	Overcast	Hot	Strong	No

Question 4: First-Order Logic Translation

You are given the predicates Market(x), which is true if x is a market, Vendor(x), which is true if x is a vendor, SellsAt(x, y), which is true if vendor x sells at market y, Local(x), which is true if x is local.

a. Use the predicates to translate the following sentences into first-order logic:

1. Every market has vendors.
2. Local vendors sell at markets that have other local vendors.
3. There is a market where all vendors are local.

b. Translate the following sentences from first-order logic into English:

1. $\forall x\ (Market(x) \Rightarrow \exists y\ (Vendor(y) \land SellsAt(y, x) \land Local(y)))$
2. $\exists x\ (Vendor(x) \land Local(x) \land \forall y\ (Market(y) \land SellsAt(x, y) \Rightarrow Local(x)))$
3. $\forall x\ \forall y\ (Vendor(x) \land Market(y) \land SellsAt(x, y) \Rightarrow \exists z\ (Vendor(z) \land SellsAt(z, y) \land \neg(x=z)))$

Question 1: Show the operation of A* search.

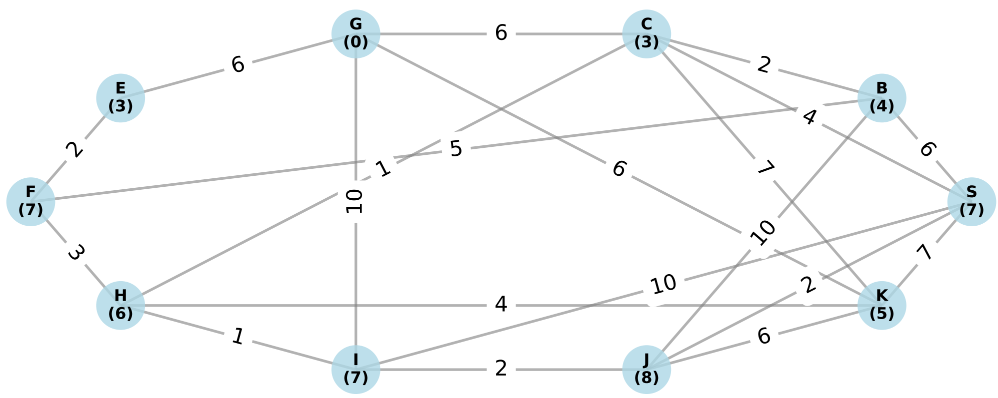
In the graph below you must show the operation of a search algorithm, starting at node S and searching for node G.

You must show all work in a table showing the node expanded and the search fringe.

In case of a tie between nodes, choose the one that comes earlier in the alphabet.

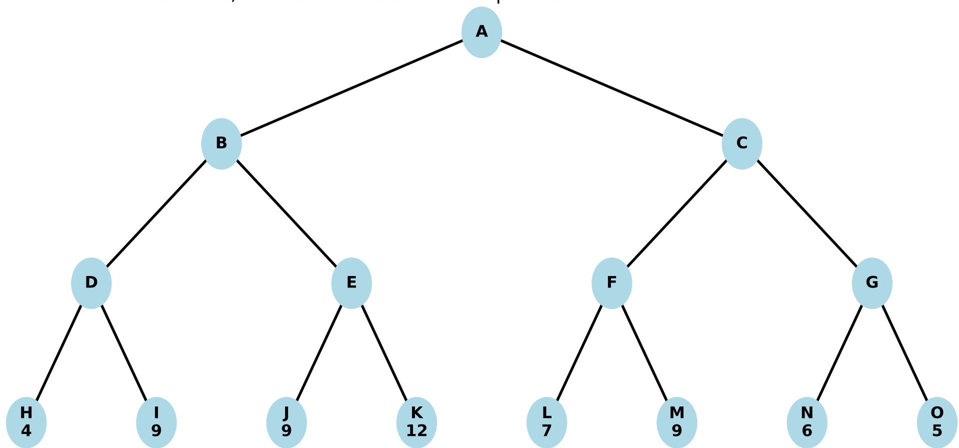
The heuristic value (the estimated distance to the goal) is the integer in each node.

Graph:



Question 2: Perform alpha-beta minimax on the following tree.

For each node show its final value, and list the nodes that are pruned.



Question 3: Draw a decision tree for the problem of deciding whether to play tennis or not.

You are given the following table of data, which has three features: the Outlook, the Temperature and the Wind,

and must draw a decision tree that classifies the positive and negative instances.

Show your computations for the information gain for each node in the tree.

Day	Outlook	Temp	Wind	Play Tennis
1	Rain	Mild	Strong	Yes
2	Rain	Mild	Weak	No
3	Sunny	Cool	Weak	No
4	Rain	Cool	Strong	No
5	Sunny	Hot	Weak	No
6	Rain	Hot	Strong	Yes
7	Overcast	Mild	Strong	Yes
8	Overcast	Mild	Weak	Yes

Question 4: First-Order Logic Translation

You are given the predicates Company(x), which is true if x is a company, Employee(x), which is true if x is an employee, WorksFor(x, y), which is true if employee x works for company y, Manager(x), which is true if x is a manager.

a. Use the predicates to translate the following sentences into first-order logic:

1. Every company has at least one employee.
2. Managers work for companies that have other managers.
3. There is a company where all employees are managers.

b. Translate the following sentences from first-order logic into English:

1. $\forall x\ (Company(x) \Rightarrow \exists y\ (Employee(y) \land WorksFor(y, x) \land Manager(y)))$
2. $\exists x\ (Employee(x) \land Manager(x) \land \forall y\ (Company(y) \land WorksFor(x, y) \Rightarrow Manager(x)))$
3. $\forall x\ \forall y\ (Employee(x) \land Company(y) \land WorksFor(x, y) \Rightarrow \exists z\ (Employee(z) \land WorksFor(z, y) \land \neg(x=z)))$

Question 1: Show the operation of A* search.

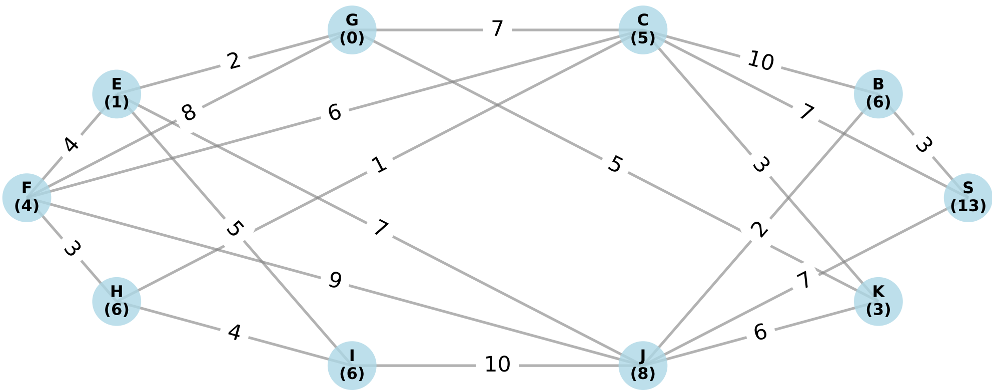
In the graph below you must show the operation of a search algorithm, starting at node S and searching for node G.

You must show all work in a table showing the node expanded and the search fringe.

In case of a tie between nodes, choose the one that comes earlier in the alphabet.

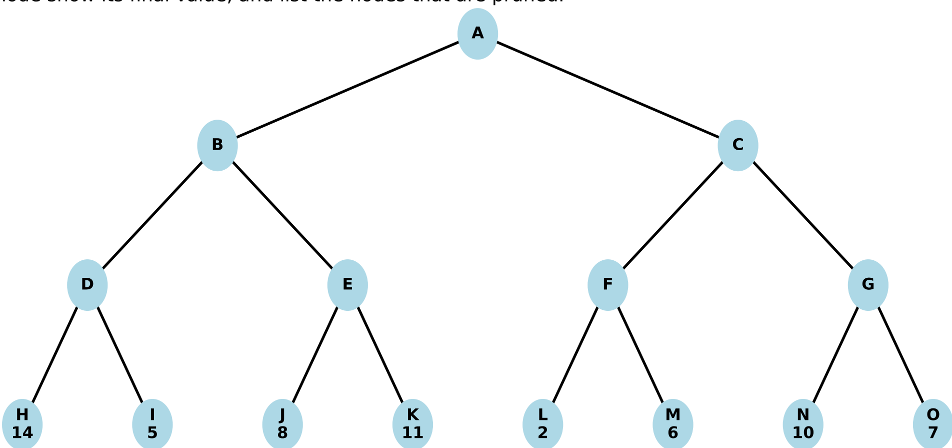
The heuristic value (the estimated distance to the goal) is the integer in each node.

Graph:



Question 2: Perform alpha-beta minimax on the following tree.

For each node show its final value, and list the nodes that are pruned.



Question 3: Draw a decision tree for the problem of deciding whether to play tennis or not.

You are given the following table of data, which has three features: the Outlook, the Temperature and the Wind, and must draw a decision tree that classifies the positive and negative instances.

Show your computations for the information gain for each node in the tree.

Day	Outlook	Temp	Wind	Play Tennis
1	Overcast	Mild	Weak	No
2	Sunny	Cool	Strong	Yes
3	Overcast	Cool	Weak	Yes
4	Sunny	Cool	Weak	Yes
5	Rain	Mild	Weak	Yes
6	Rain	Hot	Weak	No
7	Rain	Mild	Strong	Yes
8	Rain	Hot	Strong	No

Question 4: First-Order Logic Translation

You are given the predicates Gym(x), which is true if x is a gym, Member(x), which is true if x is a member, BelongsTo(x, y), which is true if member x belongs to gym y, Active(x), which is true if x is active.

a. Use the predicates to translate the following sentences into first-order logic:

1. Every gym has members.
2. Active members belong to gyms that have other active members.
3. There is a gym where all members are active.

b. Translate the following sentences from first-order logic into English:

1. $\forall x\ (Gym(x) \Rightarrow \exists y\ (Member(y) \land BelongsTo(y, x) \land Active(y)))$
2. $\exists x\ (Member(x) \land Active(x) \land \forall y\ (Gym(y) \land BelongsTo(x, y) \Rightarrow Active(x)))$
3. $\forall x\ \forall y\ (Member(x) \land Gym(y) \land BelongsTo(x, y) \Rightarrow \exists z\ (Member(z) \land BelongsTo(z, y) \land \neg(x=z)))$