Problem-solving Agents

AI+X
INSTITUTE FOR ARTIFICIAL INTELLIGENCE

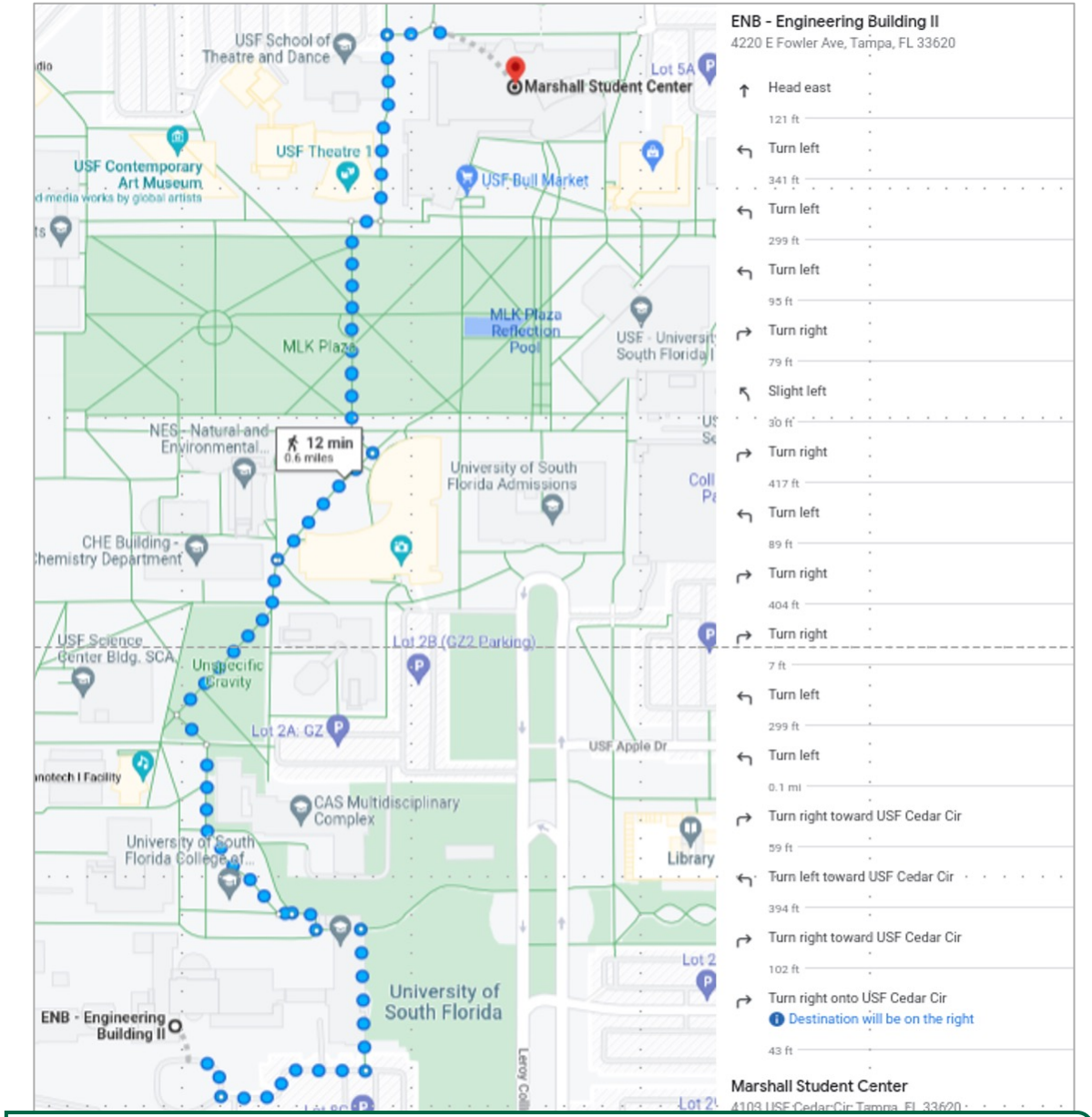


Figure 1. Sequence of action that will take you from the CSE department to Marshal Student Center at USF.

Problem-solving Agents

Problem-solving Agent: Agents that plan ahead to decide which action to take.

Search: The process of finding the most suitable action.

The next action depends on finding a sequence of actions that achieve a **goal**.

Search Problems

A search problem consists of:

- a state space [figure on the side]
- a successor function (with actions, costs) [arrows between states]
- a start state [top-left state] and a goal test [reach any state in the bottom of the graph]
- A solution is a sequence of actions (a plan) which transforms the start state to a goal state [S,R,S]

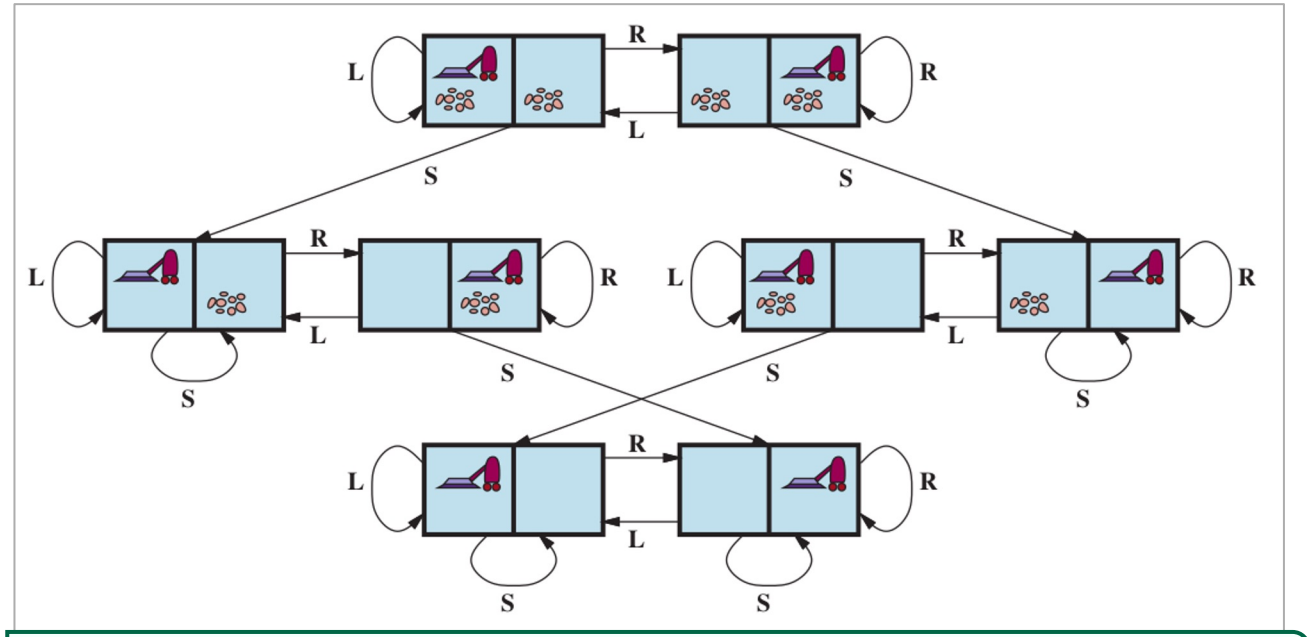


Figure 2. State space of the vacuum-cleaner world. Actions are suck dirt (S), move to the left (L) and move to the right (R).

Knowledge Check 1

Which plan of actions will reach a goal state (any state bottom in the bottom of the graph on the side) for any initial state?

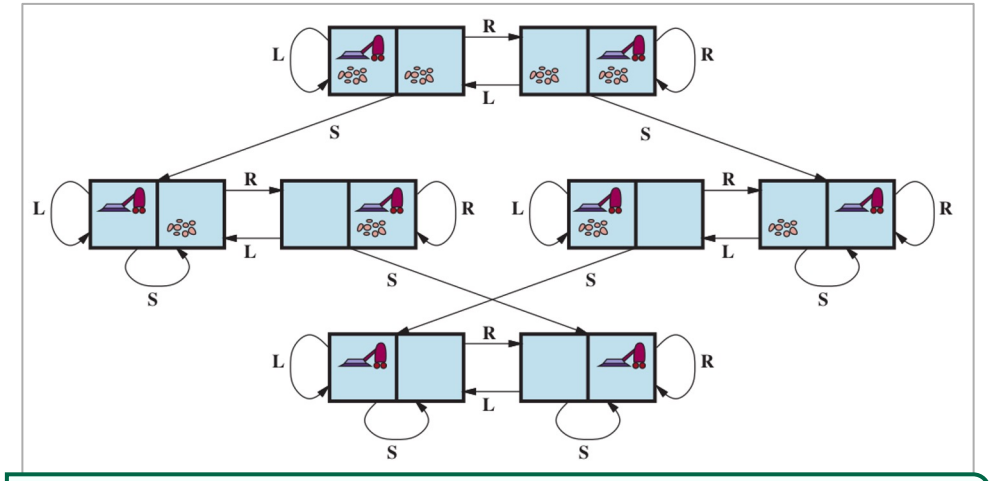


Figure 2. State space of the vacuum-cleaner world. Actions are suck dirt (S), move to the left (L) and move to the right (R).

A. S, R, S

B. S, R, S, L

C. L, S, R, S

D. S, L, R, S

Search Problems

A search problem consists of:

- State space graph
  - Mathematical representation of a search problem
- Nodes represent world configurations
- Arcs represent action results (successors)
- The goal test is a set of goal nodes
- No duplicate states
- For most problems, we cannot build the full graph in memory

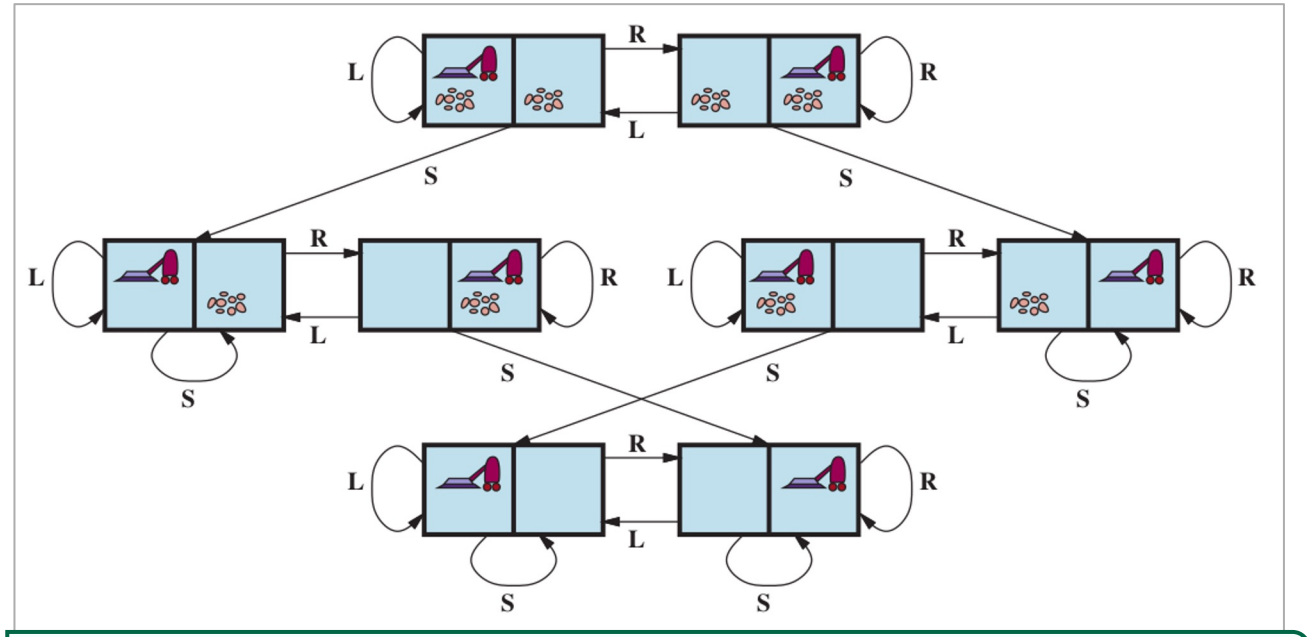


Figure 2. State space of the vacuum-cleaner world. Actions are suck dirt (S), move to the left (L) and move to the right (R).

Search Trees

- A search tree represents plans and their outcomes
- The start state is the root node
- Child nodes corresponds to successors
- Multiple tree nodes can represent the same state in the graph, as they are part of different plans.

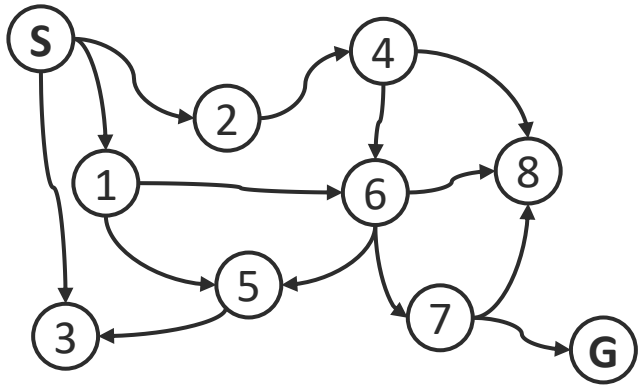


Figure 3. Small state space graph

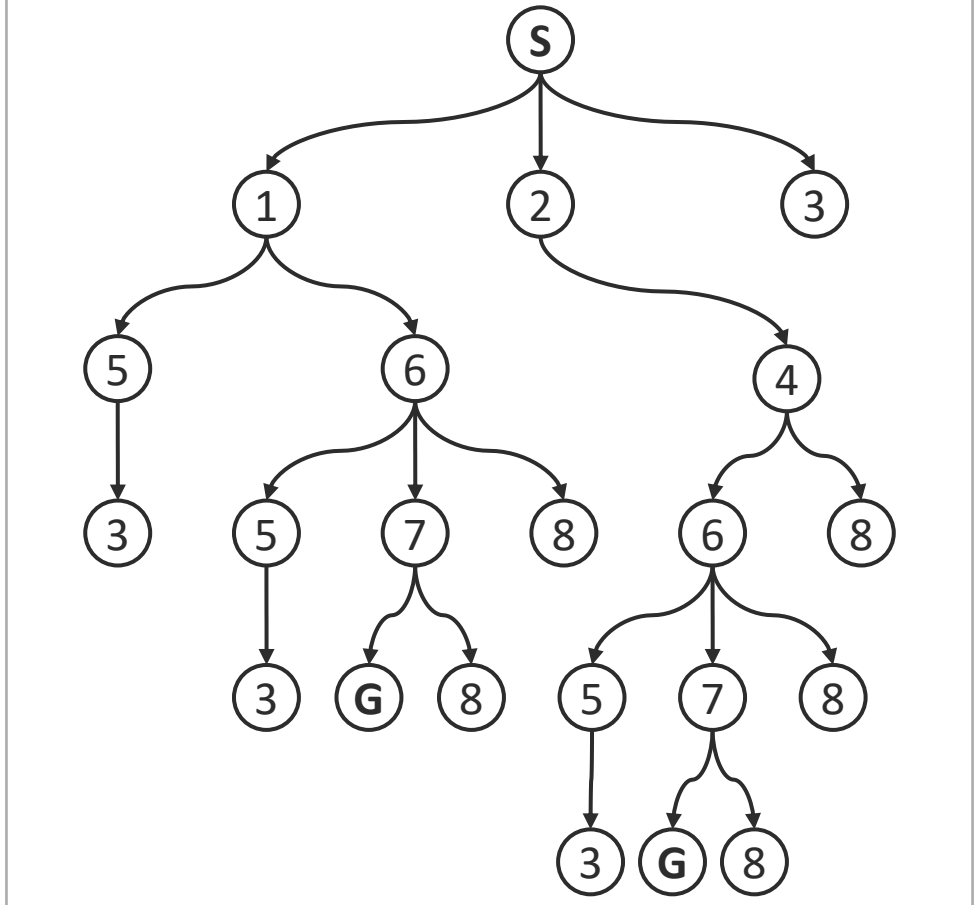


Figure 4. Search tree for the graph on the left

Search Trees

- A search tree represents plans and their outcomes
- The start state is the root node
- Child nodes corresponds to successors
- Multiple tree nodes can represent the same state in the graph, as they are part of different plans.

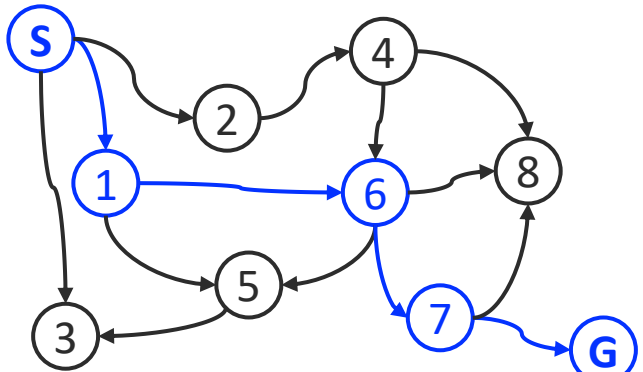


Figure 5. Small state space graph

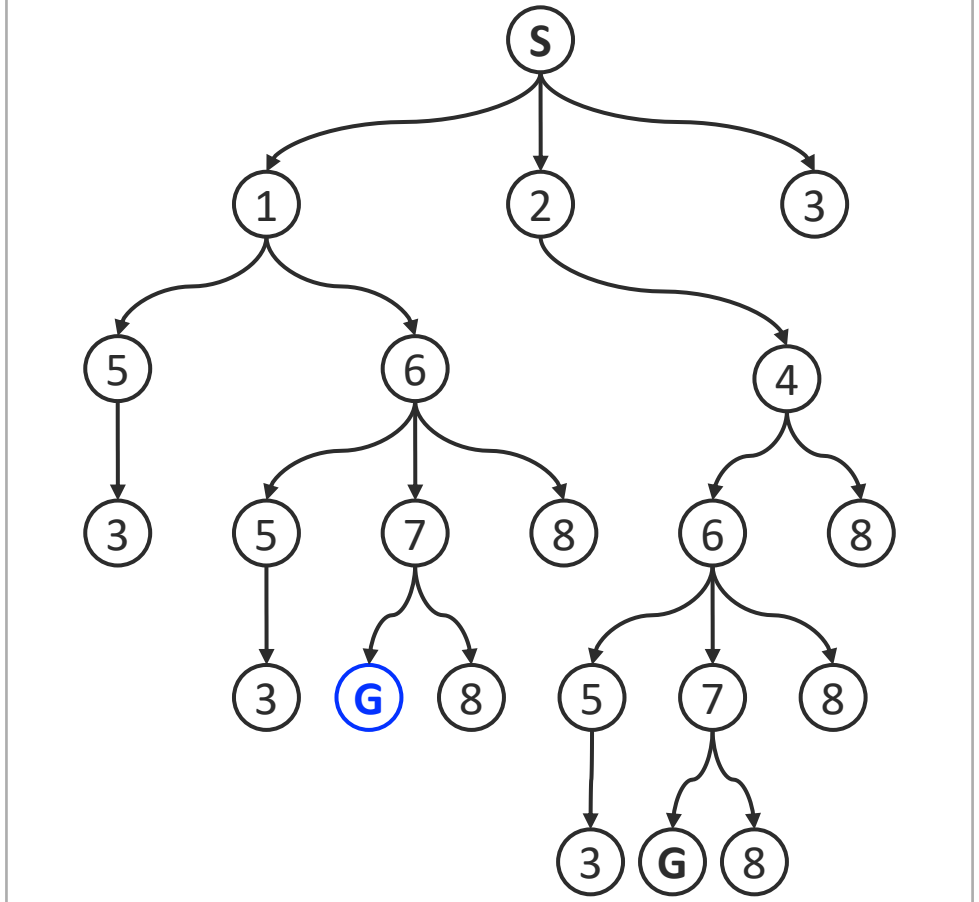


Figure 6. Search tree for the graph on the left

Search Trees

- A search tree represents plans and their outcomes
- The start state is the root node
- Child nodes corresponds to successors
- Multiple tree nodes can represent the same state in the graph, as they are part of different plans.

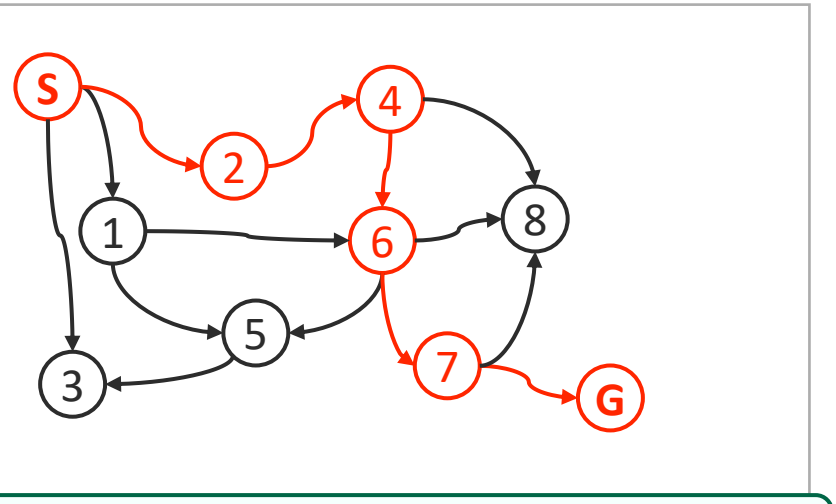


Figure 7. Small state space graph

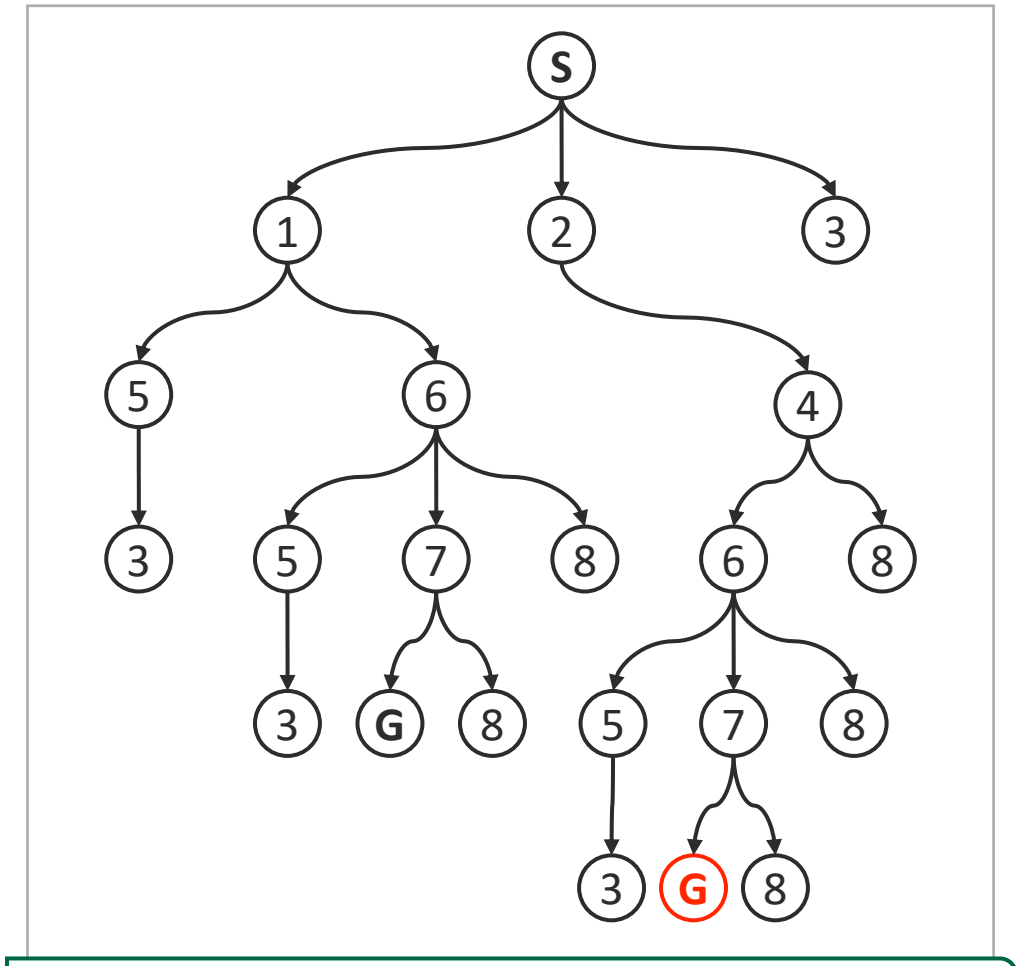


Figure 8. Search tree for the graph on the left

Searching with a Search Tree

- Expand out potential plans (tree nodes)
- Maintain a fringe of partial plans under consideration
- Try to expand as few tree nodes as possible

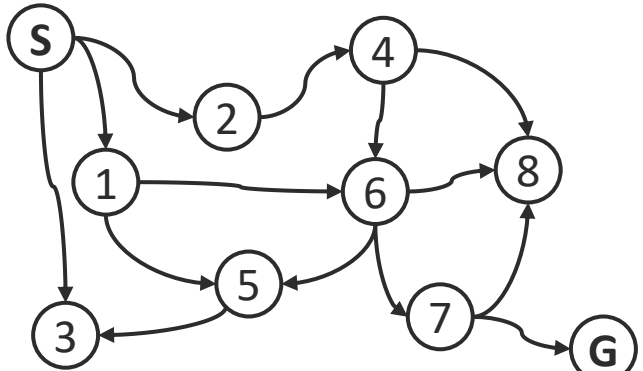


Figure 3. Small state space graph

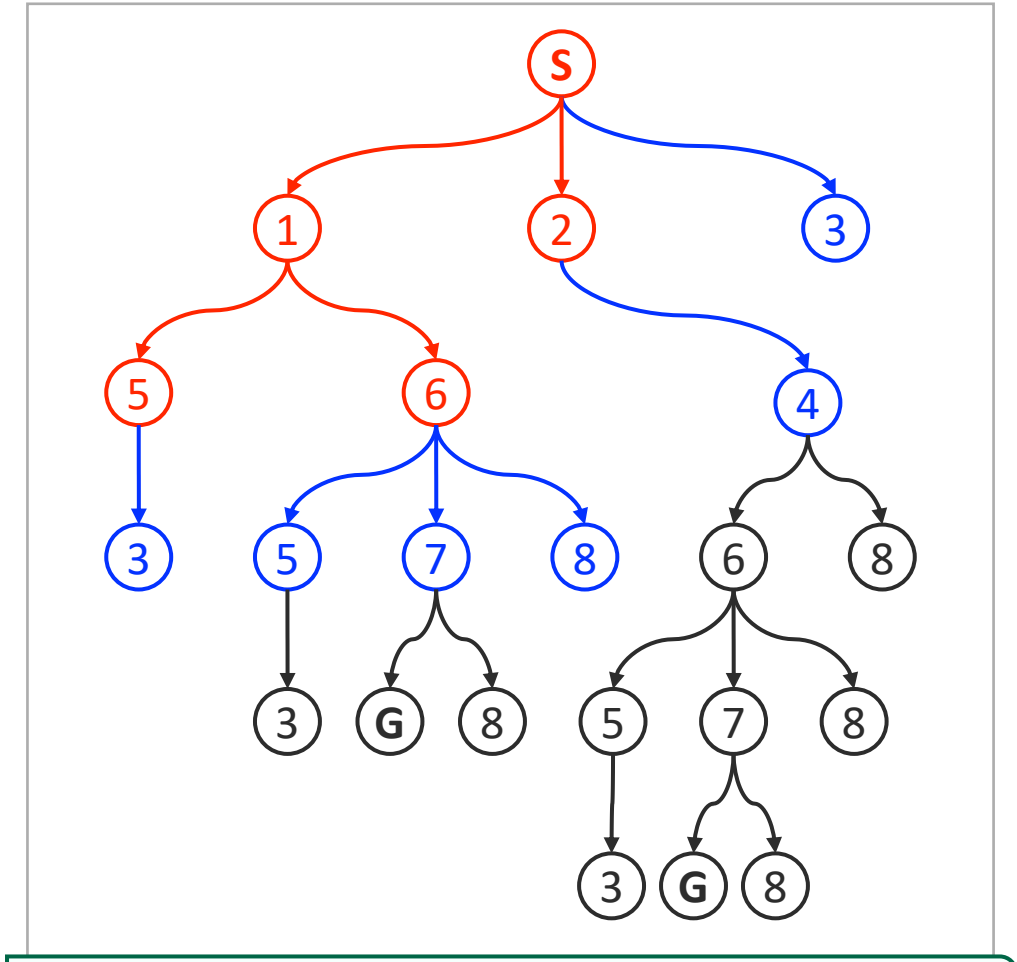


Figure 9. Search tree for the graph on the left

Searching with a Search Tree

```
set initial state of problem as the tree root
while True:
  choose leaf node for expansion
  if there are no candidates for expansion
    return failure
  if chosen node is a goal state
    return path from root to node
  else
    expand node and add neighbors to the tree
```

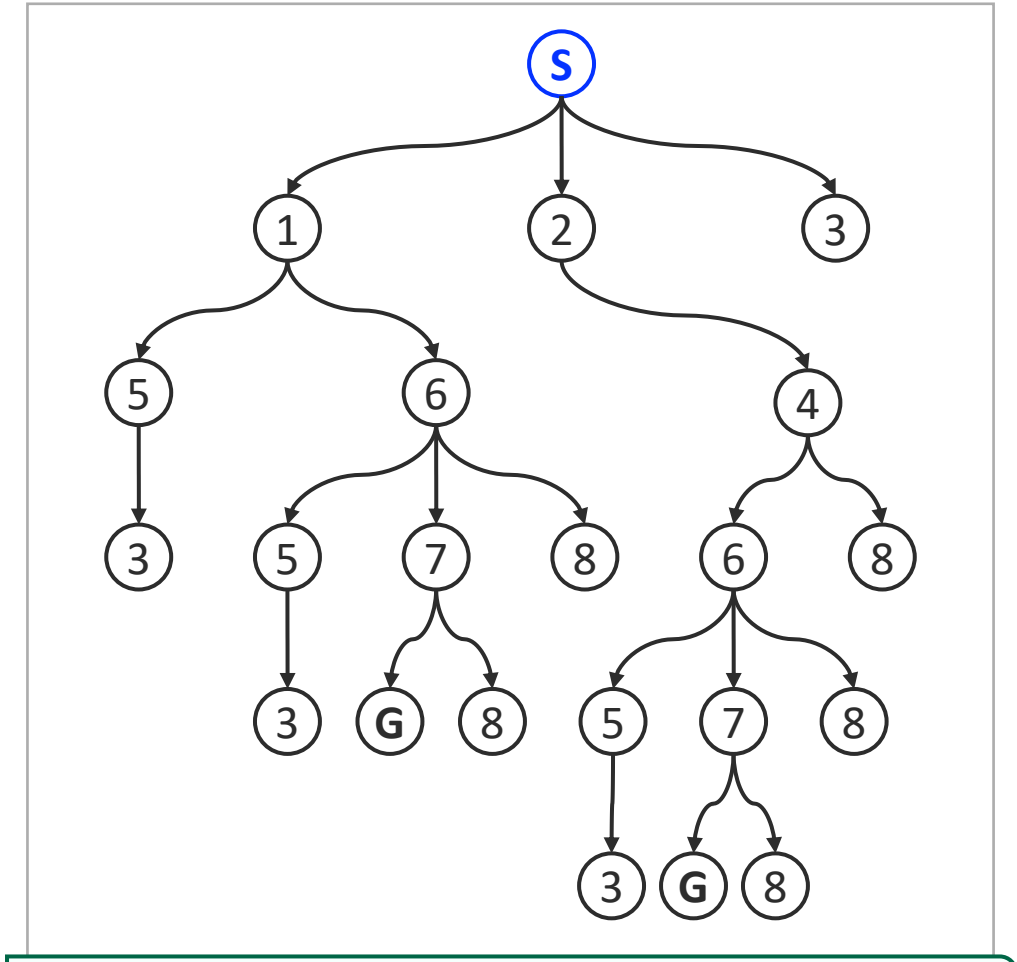


Figure 10. Search Tree

Searching with a Search Tree

```
set initial state of problem as the tree root
while True:
  choose leaf node for expansion
  if there are no candidates for expansion
    return failure
  if chosen node is a goal state
    return path from root to node
  else
    expand node and add neighbors to the tree
```

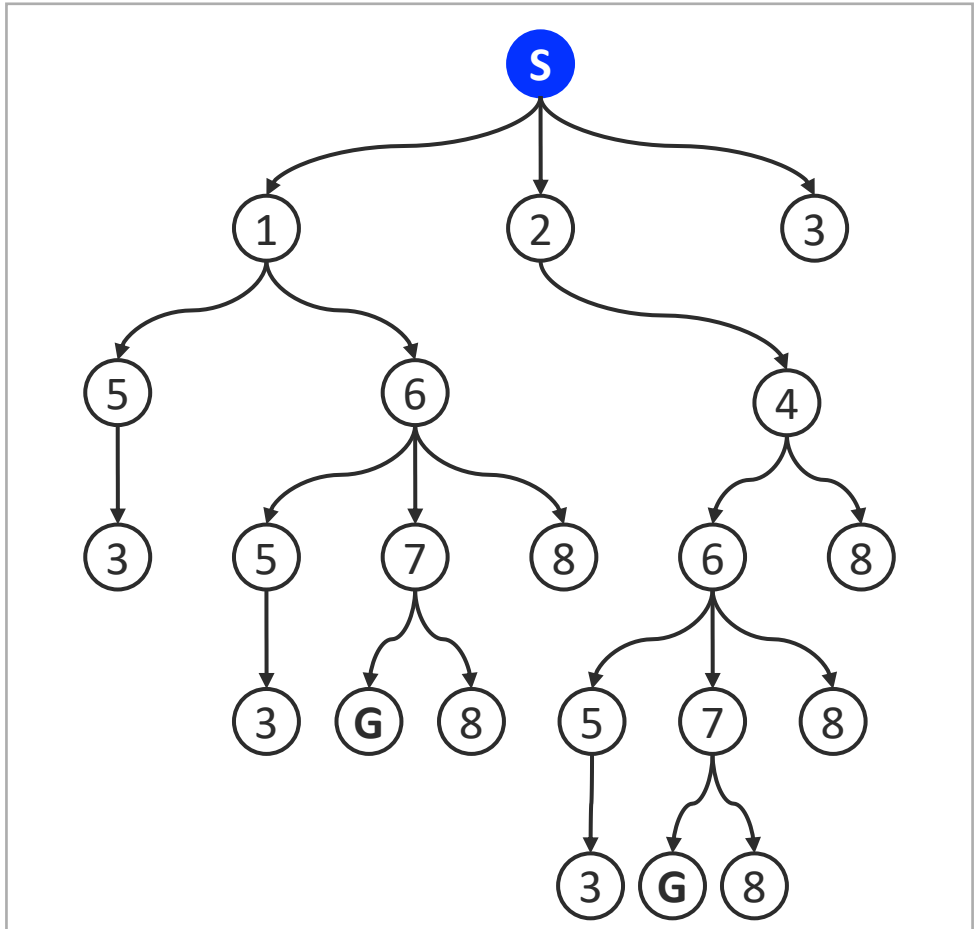


Figure 10. Search Tree

Searching with a Search Tree

```
set initial state of problem as the tree root
while True:
  choose leaf node for expansion
  if there are no candidates for expansion
    return failure
  if chosen node is a goal state
    return path from root to node
  else
    expand node and add neighbors to the tree
```

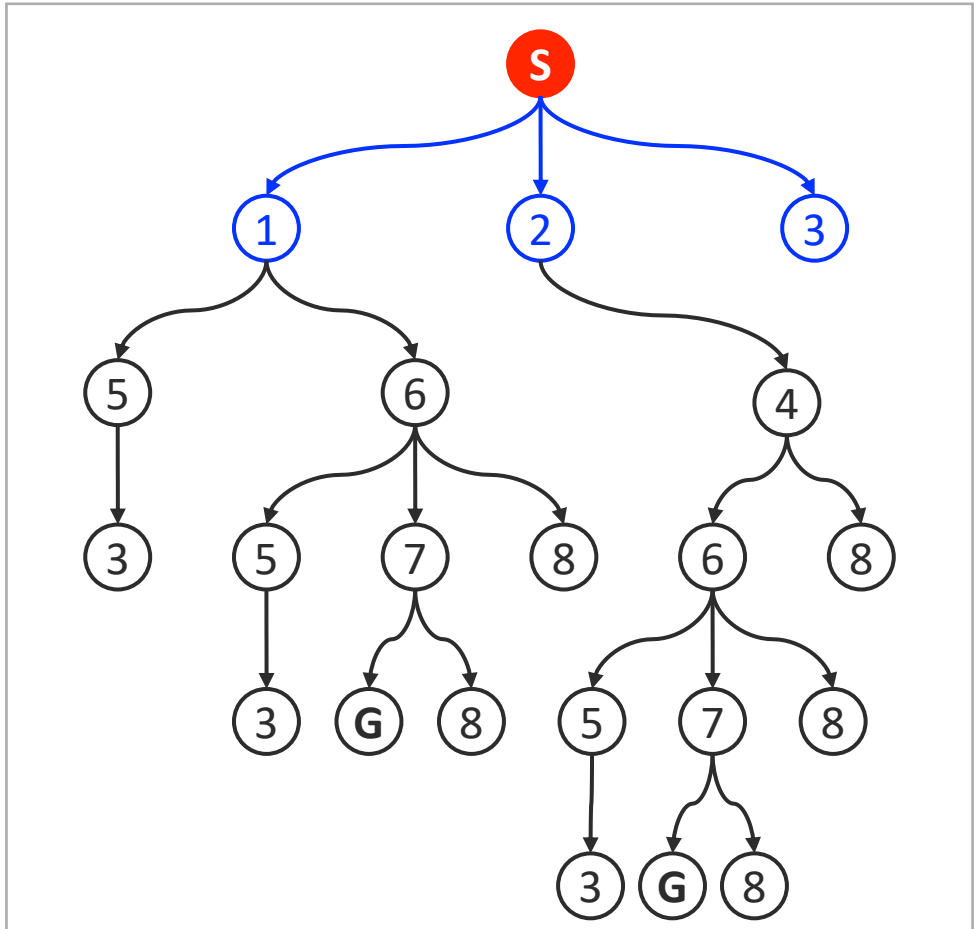


Figure 10. Search Tree

Searching with a Search Tree

```
set initial state of problem as the tree root
while True:
  choose leaf node for expansion
  if there are no candidates for expansion
    return failure
  if chosen node is a goal state
    return path from root to node
  else
    expand node and add neighbors to the tree
```

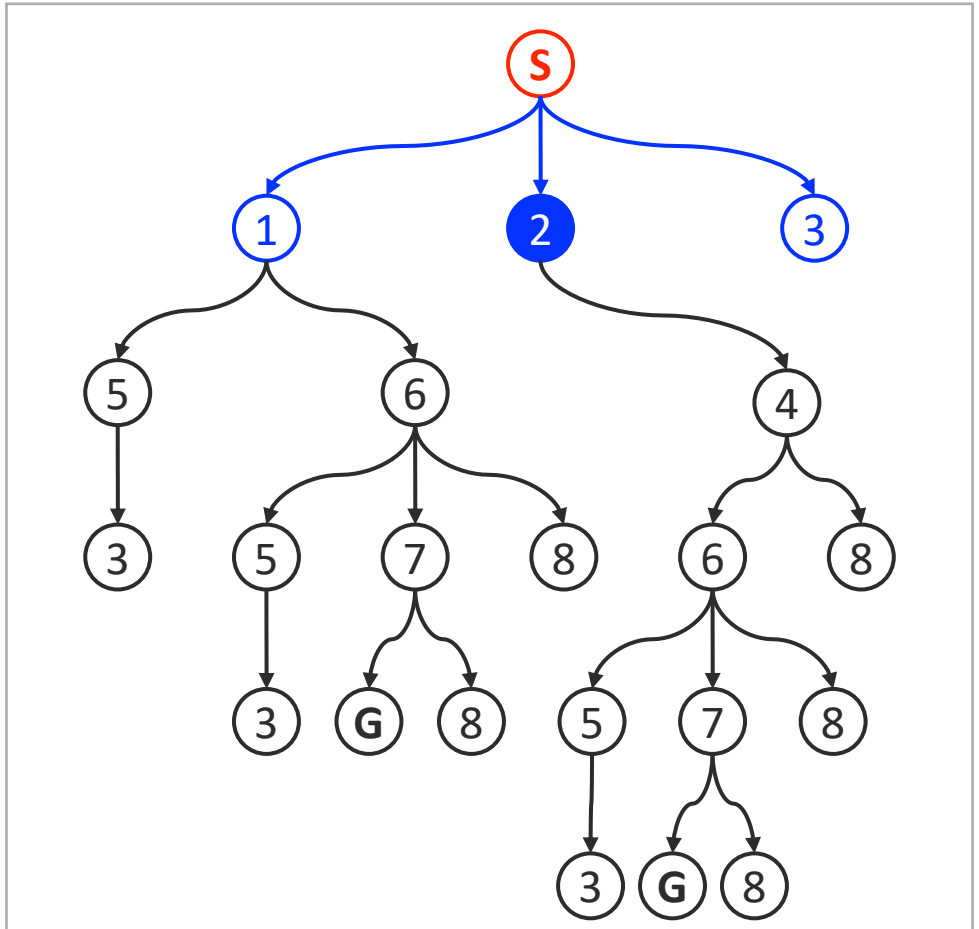


Figure 10. Search Tree

Searching with a Search Tree

```
set initial state of problem as the tree root
while True:
  choose leaf node for expansion
  if there are no candidates for expansion
    return failure
  if chosen node is a goal state
    return path from root to node
  else
    expand node and add neighbors to the tree
```

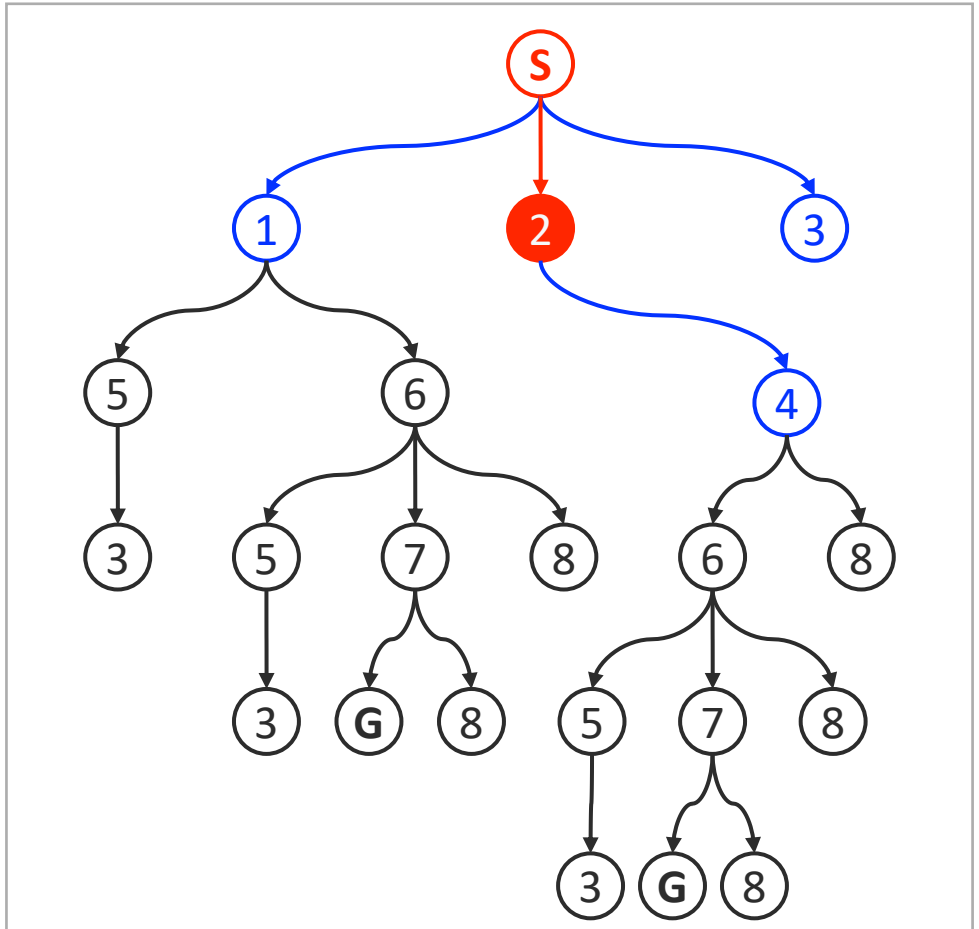


Figure 11. Search Tree

Searching with a Search Tree

```
set initial state of problem as the tree root
while True: (after a few iterations!!!)
  choose leaf node for expansion
  if there are no candidates for expansion
    return failure
  if chosen node is a goal state
    return path from root to node
  else
    expand node and add neighbors to the tree
```

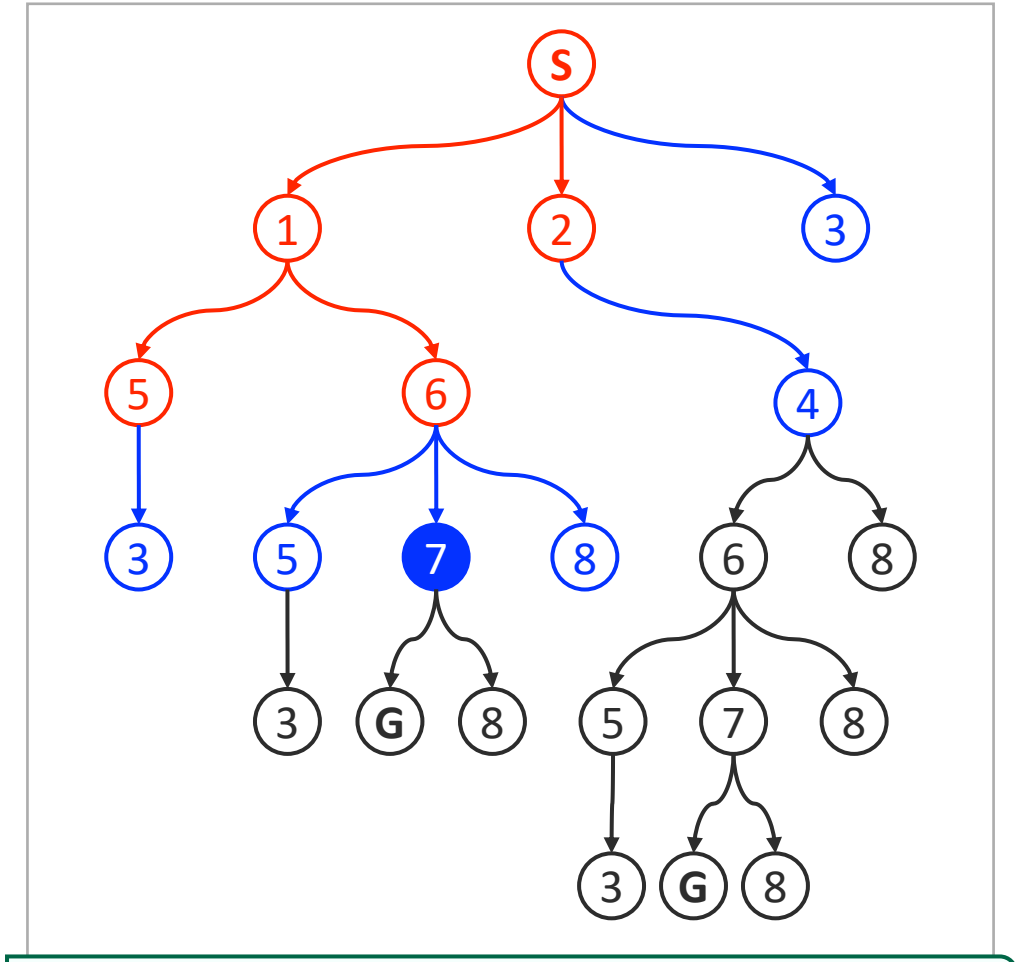


Figure 12. Search Tree

Searching with a Search Tree

```
set initial state of problem as the tree root
while True:
  choose leaf node for expansion
  if there are no candidates for expansion
    return failure
  if chosen node is a goal state
    return path from root to node
  else
    expand node and add neighbors to the tree
```

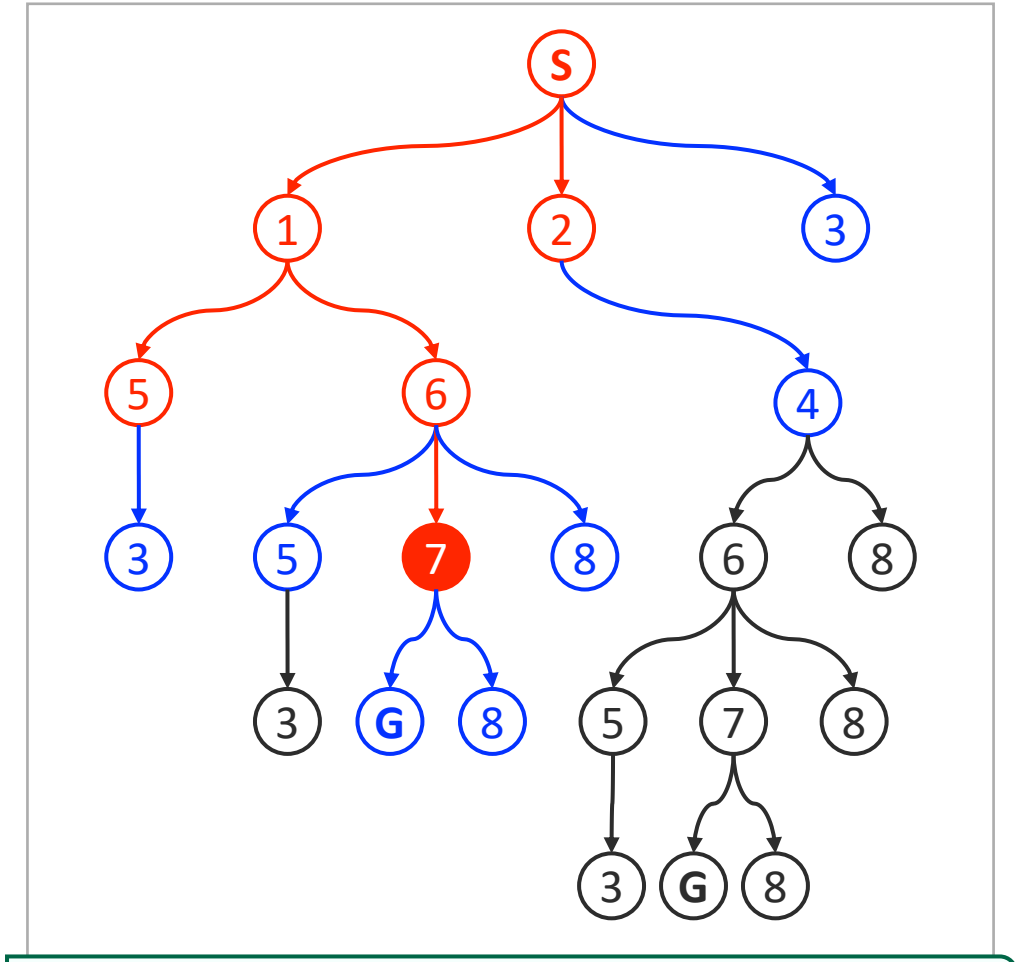


Figure 13. Search Tree

Searching with a Search Tree

```
set initial state of problem as the tree root
while True:
  choose leaf node for expansion
  if there are no candidates for expansion
    return failure
  if chosen node is a goal state
    return path from root to node
  else
    expand node and add neighbors to the tree
```

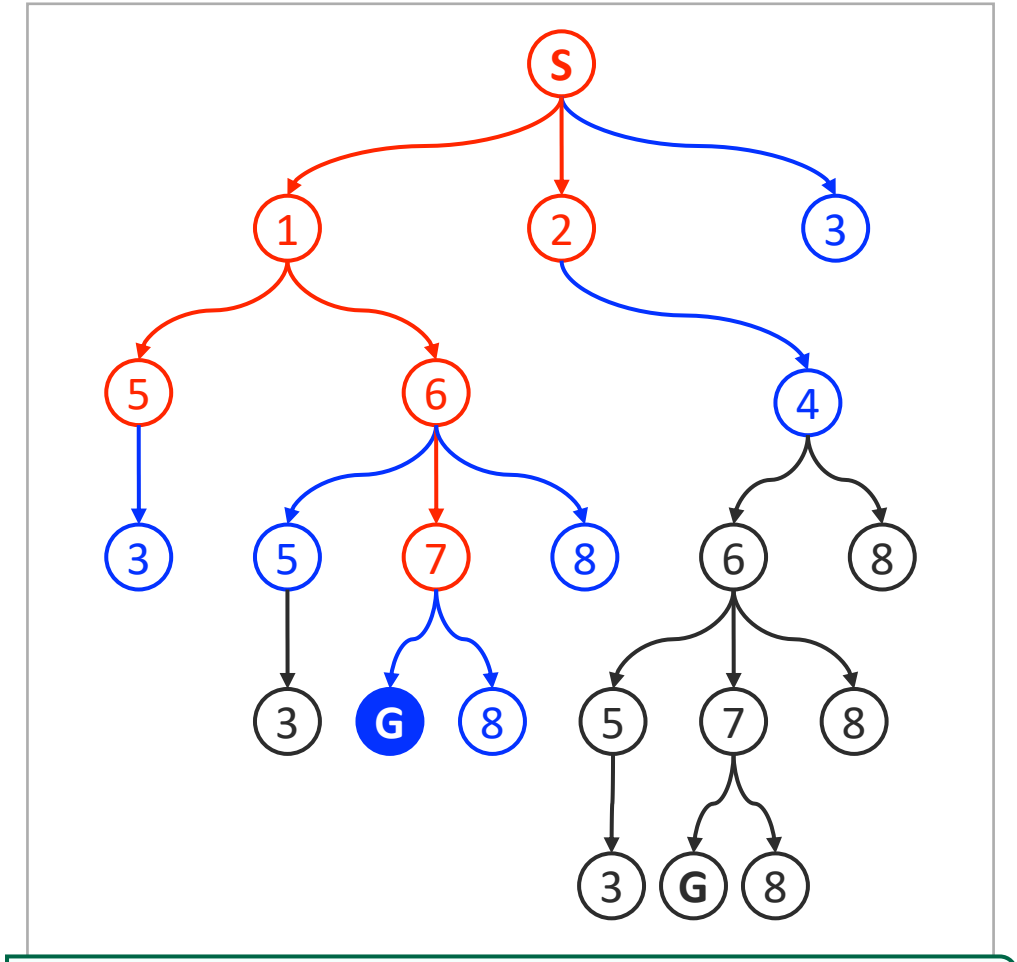


Figure 14. Search Tree

Searching with a Search Tree

```
set initial state of problem as the tree root
while True:
  choose leaf node for expansion
  if there are no candidates for expansion
    return failure
  if chosen node is a goal state
    return path from root to node
  else
    expand node and add neighbors to the tree
```

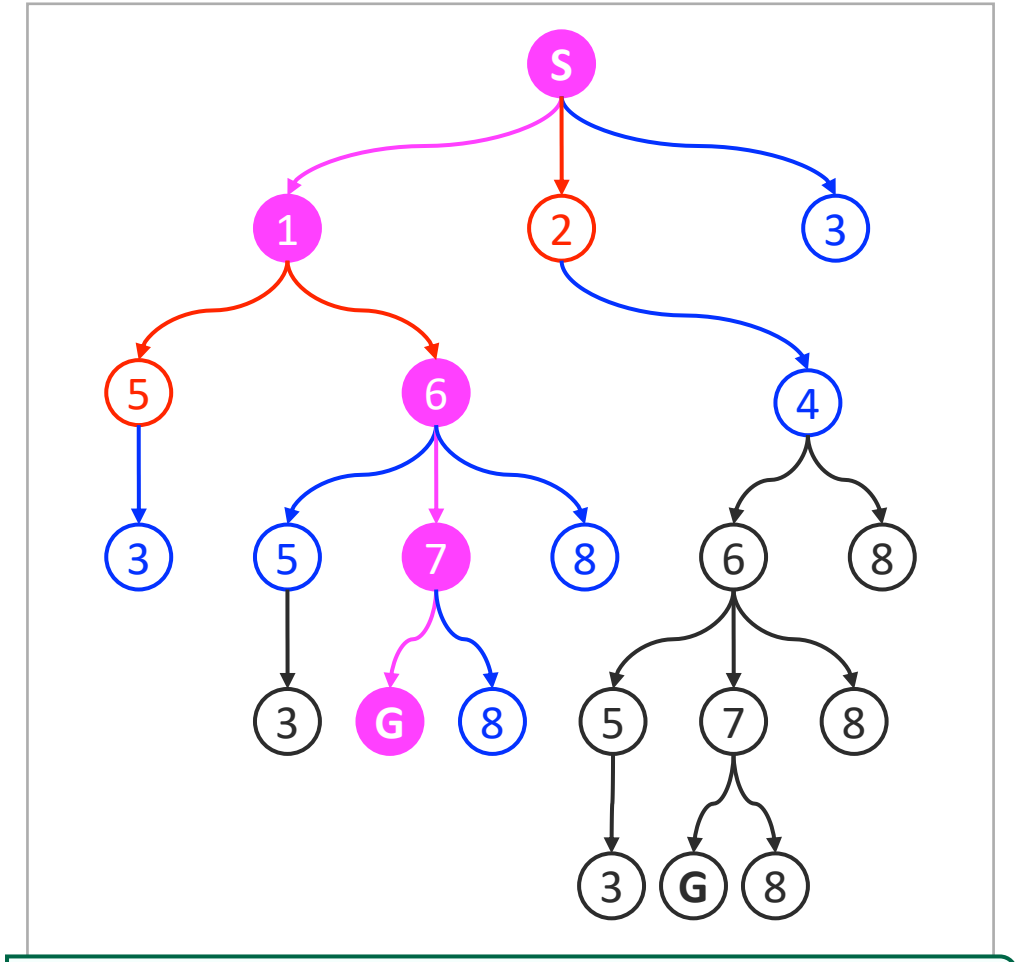


Figure 15. Search Tree

Search Trees

- For most problems, we cannot build the full search tree in memory.
- Cycles in the state space graph result in infinitely big search trees
  - A search can get stuck in an infinite loop
- Different problems might require different search strategies.

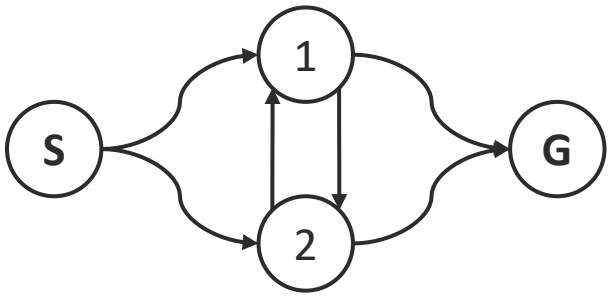


Figure 16. Small state space graph with cycle

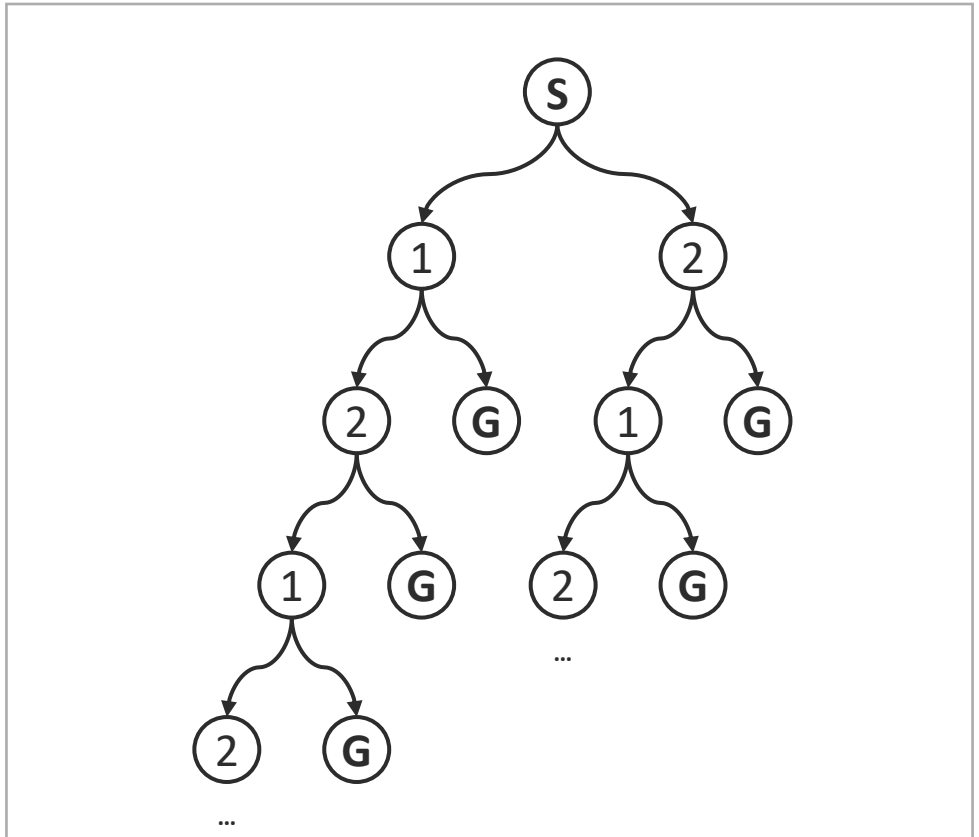


Figure 17. Search tree for the graph on the left

Evaluating Search Strategies

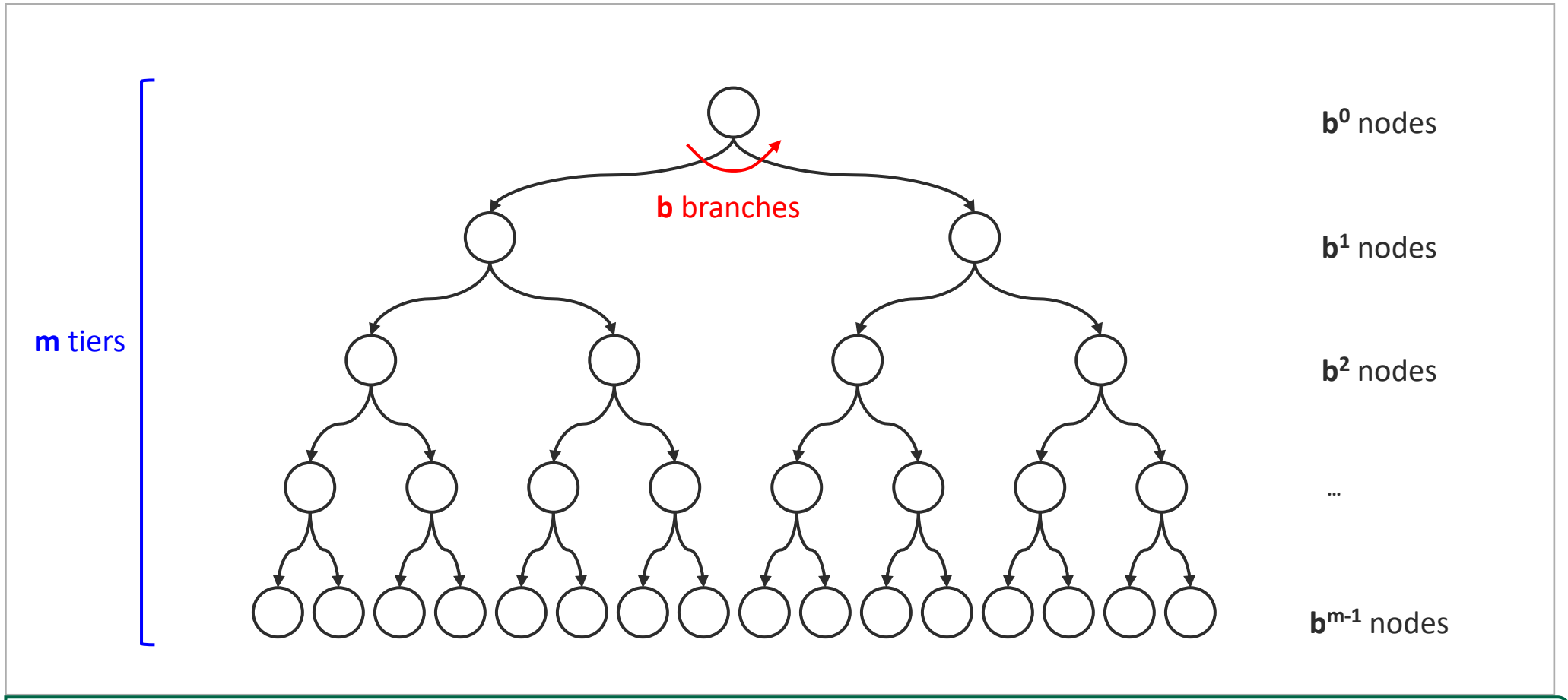


Figure 18. Generic search tree

Knowledge Check 2

What is the total number of nodes in a tree with **m** tiers and branching factor **b**?

A $m \times b^{m-1}$

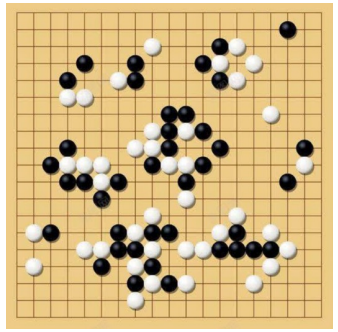
B b^{2m-1}

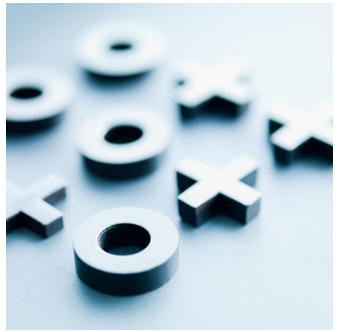
C $(b^m-1) \div (b-1)$

D $b^0 \times b^1 \times b^2 \times \ldots \times b^{m-1}$

Knowledge Check 3

When using search to play games, the state of the game can be seen as a node in the search tree, and the average number of valid moves corresponds to the branching factor. Can you correctly guess the approximate branching factor of Chess, Checkers, Go, and Tic-Tac-Toe?





A. 35, 6, 250, and 5

B. 250, 6, 35, and 5

C. 35, 5, 250, and 6

D. 250, 5, 35, and 6

You have reached the end of the lecture.

Image/Figure References

Figure 1. Sequence of action that will take you from the CSE department to Marshal Student Center at USF. Google.maps
Figure 2. State space of the vacuum-cleaner world. Actions are suck dirt (S), move to the left (L) and move to the right (R). Source: Russell & Norvig, Artificial Intelligence: A Modern Approach, 4th edition, Pearson, 2021.
Figure 3. Small state space graph. Source: Russell & Norvig, Artificial Intelligence: A Modern Approach, 4th edition, Pearson, 2021.
Figures 4-17. Search tree for the graph on the left. Source: Russell & Norvig, Artificial Intelligence: A Modern Approach, 4th edition, Pearson, 2021.
Figure 18. Generic search tree. Source: Russell & Norvig, Artificial Intelligence: A Modern Approach, 4th edition, Pearson, 2021.
Other images were purchased from Getty Images and with permission to use.