A* Search and Best-First Search

Understanding Heuristics and Search Algorithms

A* Search Algorithm

- A* (A-star) Search is a pathfinding and graph traversal algorithm. It is widely used due to its efficiency and accuracy in finding the shortest path.

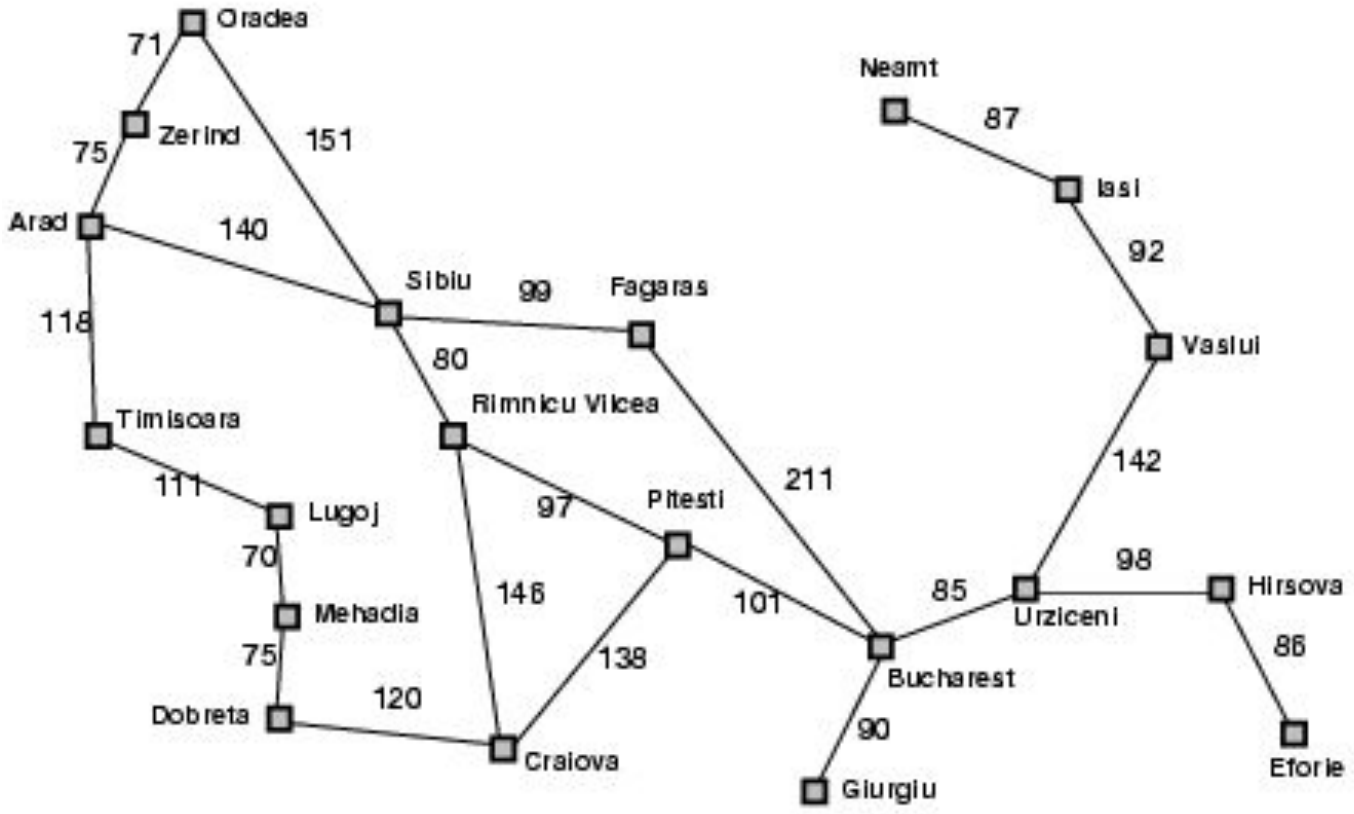
- Key Characteristics:
- - Combines both cost to reach a node (g) and a heuristic estimate to the goal (h).
- - f(n) = g(n) + h(n), where f(n) is the total estimated cost of the cheapest solution through node n.
- - Uses a priority queue to explore paths with the lowest f(n) values first.

Best-First Search Algorithm

- Best-First Search is an algorithm that explores a graph by expanding the most promising node chosen according to a specified rule.

- Key Characteristics:
- - Uses only the heuristic value (h) to choose the next node to explore.
- - Does not consider the path cost (g), which can lead to suboptimal solutions.
- - Faster than A* but not guaranteed to find the shortest path.

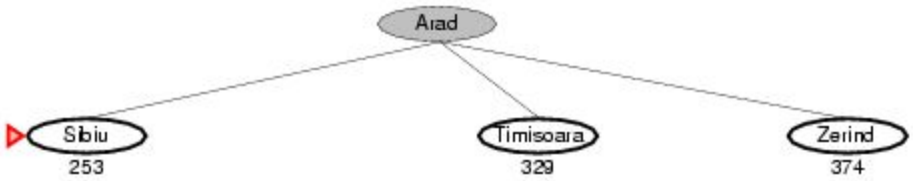
Romania with step costs in km



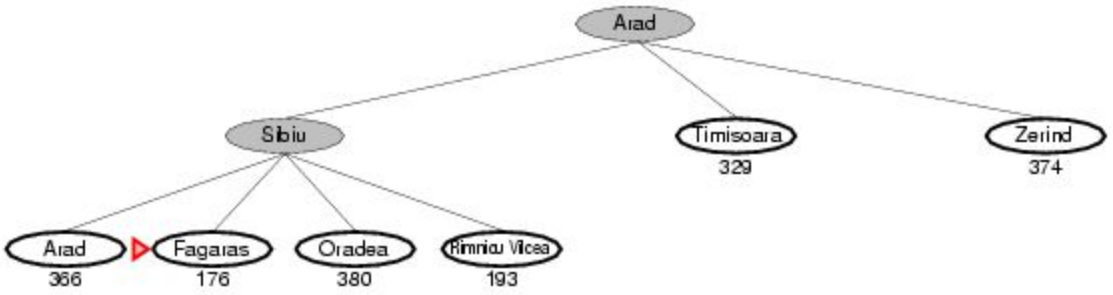
Straight–line distance to Bucharest	
Arad	366
Bucharest	0
Craiova	160
Dobreta	242
Eforie	161
Fagaras	176
Giurgiu	77
Hirsova	151
Iasi	226
Lugoj	244
Mehadia	241
Neamt	234
Oradea	380
Pitesti	10
Rimnicu Vilcea	193
Sibiu	253
Timisoara	329
Urziceni	80
Vaslui	199
Zerind	374

Greedy best-first search example

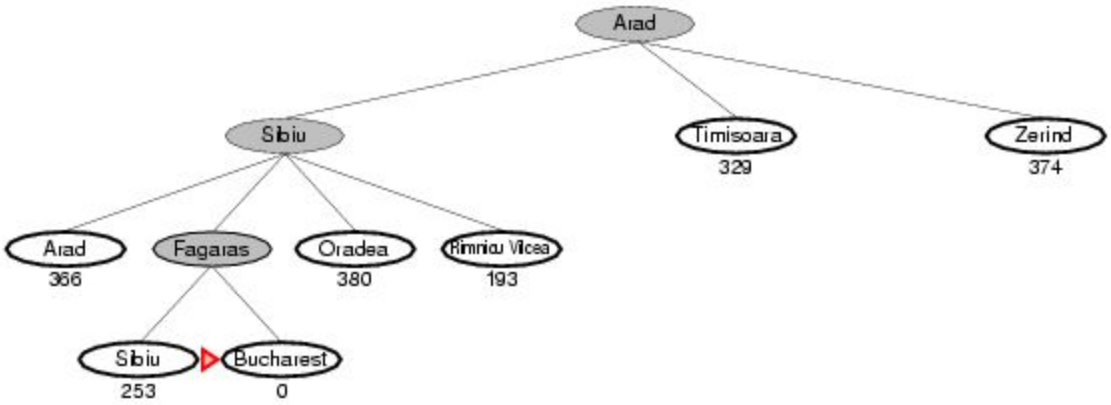
Greedy best-first search example



Greedy best-first search example



Greedy best-first search example



Heuristics in Search Algorithms

- A heuristic is a problem-solving technique used to estimate the cost of reaching the goal from a given node.

- Key Concepts:
- - Helps in making informed guesses about the best path to take.
- - The quality of the heuristic function determines the efficiency of the search algorithm.
- - Example: In pathfinding on a map, the straight-line distance (Euclidean distance) between the current location and the goal can be used as a heuristic.

A* vs. Best-First Search

- Comparison between A* Search and Best-First Search:

- - A*: Uses both path cost (g) and heuristic (h) → more accurate but slower.
- - Best-First: Uses only heuristic (h) → faster but may not find the optimal path.
- - A* is guaranteed to find the shortest path if the heuristic is admissible (never overestimates the true cost).

Uniform Cost Search

- Uniform Cost Search is a variant of Dijkstra's algorithm and is used to find the least-cost path in a graph.

- Key Characteristics:
- - Expands the node with the lowest path cost (g) regardless of the heuristic.
- - Suitable for finding the optimal solution when all actions have different costs.
- - Unlike A* or Best-First, it does not use a heuristic function.
- - Guaranteed to find the least-cost solution but can be slow if the graph has many nodes.

Comparison: A*, Best-First, Uniform Cost Search

- Comparison of different search algorithms:

- - A* Search: Uses both path cost (g) and heuristic (h). Guaranteed to find the shortest path.
- - Best-First Search: Uses only heuristic (h). Faster but may not find the optimal path.
- - Uniform Cost Search: Uses only path cost (g). Guaranteed to find the least-cost path but slower.
- - A* combines the strengths of Uniform Cost Search and Best-First Search.