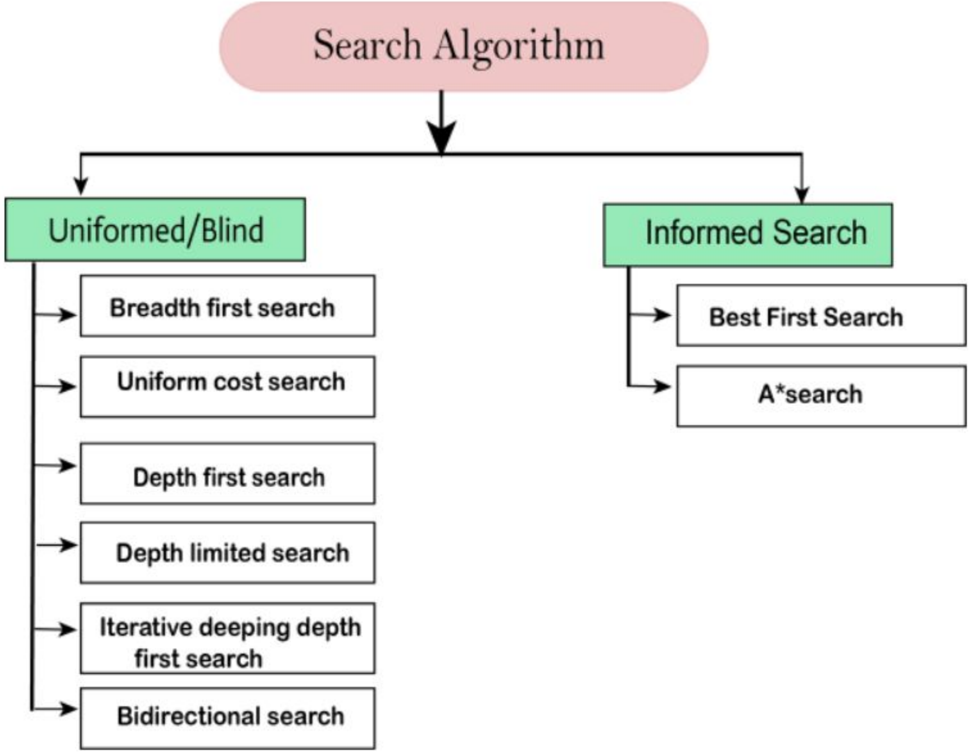
Searching Algorithms



Search Algorithm

Uniformed/Blind

- Breadth first search
- Uniform cost search
- Depth first search
- Depth limited search
- Iterative deeping depth first search
- Bidirectional search

Informed Search

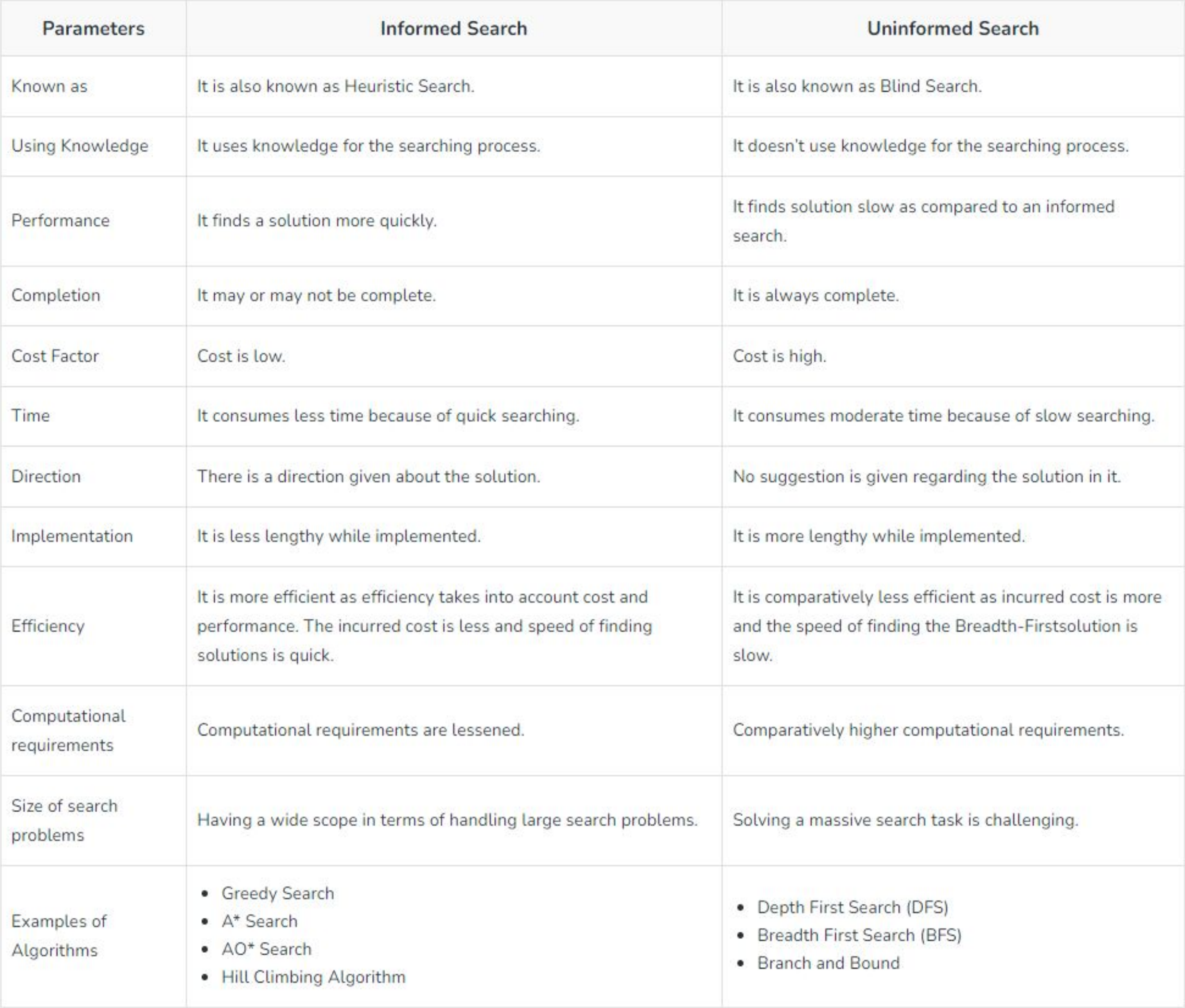
- Best First Search
- A*search

Uninformed/Blind Search

- The uninformed search have no information about the number of steps or the path cost from the current state to the goal—all they can do is distinguish a goal state from a non goal state.
- Uninformed search is also sometimes called blind search.
- It operates in a brute-force way as it only includes information about how to traverse the tree and how to identify leaf and goal nodes.
- It examines each node of the tree until it achieves the goal node.
- Types:
    - Breadth-first search
    - Uniform cost search
    - Depth-first search
    - Iterative deepening depth-first search
    - Bidirectional Search

Informed Search

- Informed search algorithms use domain knowledge.
- In an informed search, problem information is available which can guide the search.
- Informed search strategies can find a solution more efficiently than an uninformed search strategy.
- Informed search is also called a Heuristic search.
- A heuristic is a way which might not always be guaranteed for best solutions but guaranteed to find a good solution in reasonable time.
- Informed search can solve much complex problem which could not be solved in another way.
- An example of informed search algorithms is a traveling salesman problem.
  1. Greedy Search
  2. A* Search



Parameters	Informed Search	Uninformed Search
Known as	It is also known as Heuristic Search.	It is also known as Blind Search.
Using Knowledge	It uses knowledge for the searching process.	It doesn't use knowledge for the searching process.
Performance	It finds a solution more quickly.	It finds solution slow as compared to an informed search.
Completion	It may or may not be complete.	It is always complete.
Cost Factor	Cost is low.	Cost is high.
Time	It consumes less time because of quick searching.	It consumes moderate time because of slow searching.
Direction	There is a direction given about the solution.	No suggestion is given regarding the solution in it.
Implementation	It is less lengthy while implemented.	It is more lengthy while implemented.
Efficiency	It is more efficient as efficiency takes into account cost and performance. The incurred cost is less and speed of finding solutions is quick.	It is comparatively less efficient as incurred cost is more and the speed of finding the Breadth-Firstsolution is slow.
Computational requirements	Computational requirements are lessened.	Comparatively higher computational requirements.
Size of search problems	Having a wide scope in terms of handling large search problems.	Solving a massive search task is challenging.
Examples of Algorithms	• Greedy Search • A* Search • AO* Search • Hill Climbing Algorithm	• Depth First Search (DFS) • Breadth First Search (BFS) • Branch and Bound