Local beam search with k = 1 is hill-climbing search.
Local beam search with k = ∞. The idea is that if every successor is retained (because k is unbounded), then the search resembles breadth-first search in that it adds one complete layer of nodes before adding the next layer. Starting from one state, the algorithm would be essentially identical to breadth-first search except that each layer is generated all at once.
Simulated annealing with T = 0 at all times: ignoring the fact that the termination step would be triggered immediately, the search would be identical to first-choice hill climb ing because every downward successor would be rejected with probability 1.
Genetic algorithm with population size N = 1: if the population size is 1, then the two selected parents will be the same individual; crossover yields an exact copy of the individual; then there is a small chance of mutation. Thus, the algorithm executes a random walk in the space of individuals.

Percept: the agent's perceptual inputs
Percept sequence: the complete history of everything the agent has perceived
Agent function: maps any given percept sequence to action [f:p* -> a]
Agent program: runs on physical architecture to produce f (architecture + program)
Rational Agent: Agent = something that acts. Rational: doing the right thing
An agent is anything that can be viewed as perceiving its environment through sensors and acting upon that environment through actuators

Fully observable (vs. partially observable): An agent's sensors give it access to the complete state of the environment at each point in time
Deterministic (vs. stochastic): next state of env. determined by current state of agent action. if deterministic except for the actions of other agent then the env is strategic
Episodic (vs. sequential): Agent's experience is divided into atomic "episodes". Choice of action in each episode depends only on the episode itself
Static (vs. dynamic): the environment is unchanged while an agent is deliberating. Semidynamic if the environment itself doesn't change with time but the agent's performance score does
Discrete (vs. continuous): A limited number of distinct, clearly defined percepts and actions
Single agent (vs. multiagent): An agent operating by itself in an environment. Competitive vs. cooperative

Task Environment	Observable	deterministics	episodic	static	discrete	agents
crossword	fully	deterministics	sequential	static	discrete	single
chess w/clock	fully	strategic	sequential	semi	discrete	multi
taxi driver	partially	stochastic	sequential	dynamic	conti.	multi
mushroom picking	partially	stochastic	episodic	dynamic	conti.	single

•Playing soccer. P- Win/Lose E- Soccer field A- Legs,Head,Upper body S- Eyes,Ears. partially observable, multiagent, stochastic, sequential, dynamic, continuous, unknown
•Exploring the subsurface oceans of Titan. P- Surface area mapped, extraterrestrial life found E- subsurface oceans of Titan A- steering, accelerator, break, probe arm, S- camera, sonar, probe sensors. partially observable, single agent, stochastic, sequential, dynamic, continuous, unknown
• Shopping for used AI books on the Internet.P- Cost of book, quality/relevance/correct edition E- Internet's used book shops A- key entry, cursor S- website interfaces, browser. partially observable, multiagent, stochastic, sequential, dynamic, continuous, unknown
• Playing a tennis match. P- Win/Lose E- Tennis court A- Tennis racquet, Legs S- Eyes, Ears. partially observable, multiagent, stochastic, sequential, dynamic, continuous, unknown
• Practicing tennis against a wall. P- Improved performance in future tennis matches E- Near a wall A- Tennis racquet, Legs S- Eyes, Ears. observable, single agent, stochastic, sequential, dynamic, continuous, unknown
• Performing a high jump. P- Clearing the jump or not E- Track A- Legs, Body S- Eyes. observable, single agent, stochastic, sequential, dynamic, continuous,unknown
• Knitting a sweater. P- Quality of resulting sweater E- Rocking chair A- Hands,Needles S- Eyes. observable, single agent, stochastic, sequential, dynamic, continuous, unknown
• Bidding on an item at an auction. P- Item acquired, Final price paid for item E- Auction House (or online) A- Bidding S- Eyes, Ears. Partially observable, multiagent, stochastic (tie-breaking for two simultaneous bids), episodic, dynamic, continuous, know

Goal formulation: based on current situation and agent's performance measure
Problem formulation: deciding what actions and states to consider, given a goal
Problem can be defined into 5 components: Initial State, Actions, Transition model (state space), Goal Test, Path Cost

Search strategy: picking the order of node expansion

Search strategies evalution:
- Completeness: Does it always find a solution if one exists?
- Optimality: Does it always find a least-cost solution?
- Time complexity: number of nodes generated
- Space complexity: maximum number of nodes in memeory

where
- b: max branching factor of the search tree
- d: depth of the least cost solution
- m: maximum depth of the state space

Search cost (time), total cost (time + space)

General uninformed serach strategies:
- breadth first search; fifo queue frontier
- uniform cost search; priority queue frontier ordered by path cost g(n)
- depth first search; lifo queue / recursive frontier
- depth limited search
- iterative deepening search

Analysis
========

	BFS	Uniform Cost Search	Tree Search	DFS	DFS Graph Search	Depth Limited	IDS	Search
Complete	Yes	Yes, cost >= E	No	No(infinite)	Yes(finite)	Yes	Yes	no
Time	b ^ (d+1)	b ceiling (C*/E)	b ^ m	b ^ m		b ^ d	b ^ l	
Space	b ^ (d+1)	b ceiling (C*/E)	bm	not linear		b d	bl	
Optimal	Yes	yes, incrse g(n)	No	No		Yes	not	

BFS.space. keep every node in memory
BFS.optimal.if step costs are identical and in nondecreasing function

Uniform cost search. C* = cost of optimal solution

Backtracking search is a variant of DFS.
Each partially expanded nodes remember sucessors
Memory requirement is O(m) vs O(bm)

Depth Limited Search is DFS with limit l

IDS (Interative deeping search) = DFS + BFS
IDS.space. b d. tree search version
IDS.optimal. if step cost are identical and non decreasing function

Informed Search Analysis
========================
Best First search. Expand most desirable unexpanded node. frontier data structure in a decreasing order of desirability
h(n) = estimated cheapest path from node n to goal node. if 0 then goal. fn = hn
greedy best first search = hsld = straight line distance

	Greedy Best First	A*
Complete	no, get stuck	yes
Time	b ^ m	exponential
Space	b ^ m	all in ram
Optimal	no	yes

A heuristic is consistent if for every node n, every successor n' of n generated by any action a, h(n) ≤ c(n,a,n') + h(n')
f(n) is non-decreasing along any path

Variant of Hill Climbing
- stochastic hill climbing. random. uphill moves. converges slowly. better solution
- first choice hill climbing. successor random until better than current.
- random restart hill climbing. complete with probability approaching 1. series of hill climbing until goal found
  - very effective for n queen problem

Genetic Algorithm. start with population. evaluation function(fitness function). Higher values for better states
- produce next generation by selection, crossover, mutation