

Name: Wenhao Li ID:#14328112

1.

	Sudoku Puzzle	Autonomous vacuum cleaner
Observable?	Fully	Partially
Deterministic?	Deterministic	Deterministic
Episodic?	Sequential	Episodic
Static?	Static	Static
Discrete?	Discrete	Discrete
Single-agent?	Single-agent	Single-agent

2.

a) Performance: number of steps taken

Environment: the location of the tiles

Actuator: tile mover

Sensors: camera

b) Performance: dirt cleaned, movements, energy consumption

Environment: rooms, dirt

Actuator: wheels, vacuum motor

Sensors: dirt sensor, location sensor

3.

a) Yes. It maximize the performance measure in this circumstance.

b) Yes. It also maximize the performance measure.

c) Yes.

d) function VACUUM-AGENT([loc, status]) -> action

if loc==A or C

do{

if status==dirty , suck

turn right }

if loc==D

do{

if status==dirty, suck

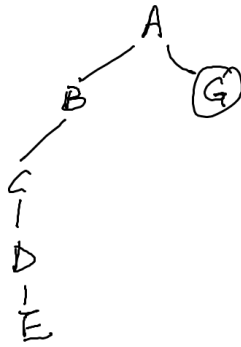
turn left }

4. Each room can be dirty or cleaned, so there are a total of 2^{24} combinations.

A vacuum cleaner can be in any of the 24 rooms, which counts for 24 location possibilities.

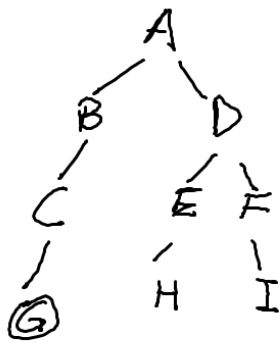
So, there are a total of $24 \cdot 2^{24}$ states in total.

5. Because breadth-first search always expand the most shallow path while depth-first always expand one path, we can easily construct such a tree that the goal state is shallow and there is one path which is very deep.



For this particular search tree, the order of breadth-first search is A-B-G, therefore it takes a space of 3 nodes, while depth-first search goes A-B-C-D-E-G, taking a space of 6 nodes.

6. Interactive deepening search is basically depth-first search within a depth limit at one time. So we can construct search a goal state where it is deep down the first path that interactive deepening search generates more nodes than a depth-first search.



In this example, a depth-first search will search through A-B-C-G, but an interactive deepening search will go through A-B-D-C-E-F then finally G.

7.

a)

1)S-C-E-F-A

2)S-F-D

3)5

b)

1)S-C-F-B-E

2)S-F-B-D

3) 4

c)

1)S-C-E-F-A

2)S-F-D

3)5

8.

a) No-op. Cost=0

- b) Move left-suck or move right-suck. Cost=-1.1
- c) Move left-suck-noop or move right-suck-noop, cost=-1.1
- d) Move right-suck-move right-suck. Cost=-2.2