CS4246 / CS5446

Tutorial Week 6

Muhammad **Rizki** Maulana
rizki@u.nus.edu

First

Modeling with MDPs. Specify each of the following problems as a Markov decision process, i.e. specify the state space, the actions, the transition functions, and the reward function. What is the (approximate) size of the state space and the action space?

(c) Atari games. Atari games have 128 bytes of RAM, 18 actions, and 33,728 screen pixels taking values from 0-127.

Question

Modeling with MDPs. Specify each of the following problems as a Markov decision process, i.e. specify the state space, the actions, the transition functions, and the reward function. What is the (approximate) size of the state space and the action space?

(c) Atari games. Atari games have 128 bytes of RAM, 18 actions, and 33,728 screen pixels taking values from 0-127.

1 byte = 256 values (-128 … 127)

State:

Ram 256^128

Modeling with MDPs. Specify each of the following problems as a Markov decision process, i.e. specify the state space, the actions, the transition functions, and the reward function. What is the (approximate) size of the state space and the action space?

(c) Atari games. Atari games have 128 bytes of RAM, 18 actions, and 33,728 screen pixels taking values from 0-127.

State:

Ram	256^128
Pixels	not MDP

Only contains position information, no velocity and acceleration!

Image credit: ATARI Games, breakout

Modeling with MDPs. Specify each of the following problems as a Markov decision process, i.e. specify the state space, the actions, the transition functions, and the reward function. What is the (approximate) size of the state space and the action space?

(c) Atari games. Atari games have 128 bytes of RAM, 18 actions, and 33,728 screen pixels taking values from 0-127.

State:

Ram	256^128
Pixels	not MDP, might need to consider more than one frames

2 frames can capture velocity:

$v_t = pos_t - pos_{t-1}$

4 frames can capture acceleration:

$a_t = v_t - v_{t-1}$

Image credit: ATARI Games, breakout

Modeling with MDPs. Specify each of the following problems as a Markov decision process, i.e. specify the state space, the actions, the transition functions, and the reward function. What is the (approximate) size of the state space and the action space?

(c) Atari games. Atari games have 128 bytes of RAM, 18 actions, and 33,728 screen pixels taking values from 0-127.

State:

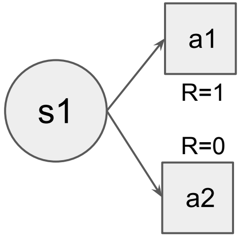
Ram	256^128
Pixels	not MDP, might need to consider more than one frames

Actions: 18

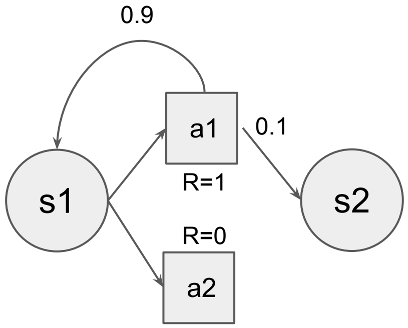
Transitions & Rewards: depends on the game

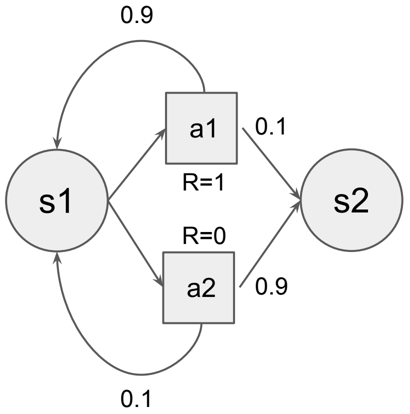
Image credit: ATARI Games, breakout

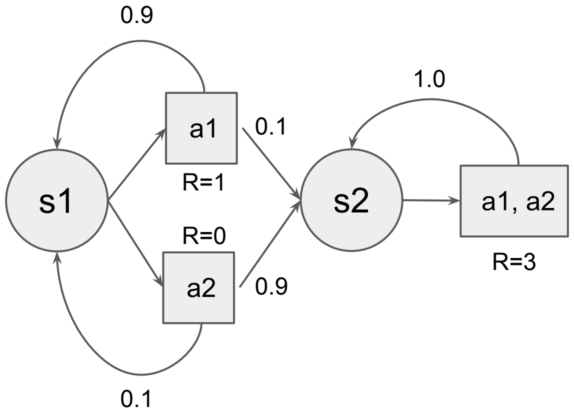
Second



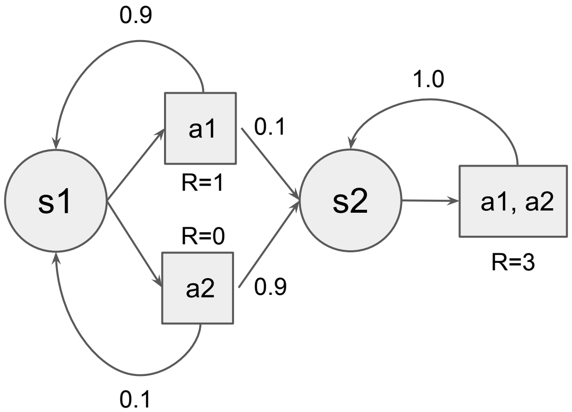
s1
a1
R=1
R=0
a2


0.9
a1
0.1
s1
R=1
s2
R=0
a2

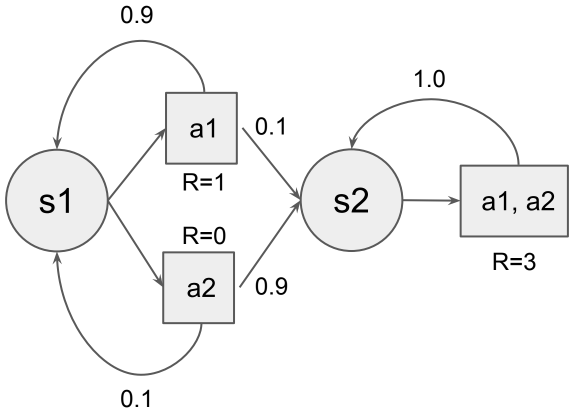

0.9
a1
0.1
s1
R=1
s2
R=0
a2
0.9
0.1



0.9
a1
0.1
R=1
s1
s2
a1, a2
1.0
R=0
a2
0.9
R=3
0.1



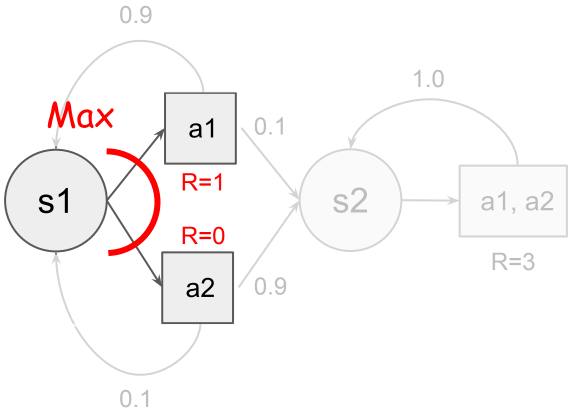
Discount factor : 0.9



Discount factor : 0.9

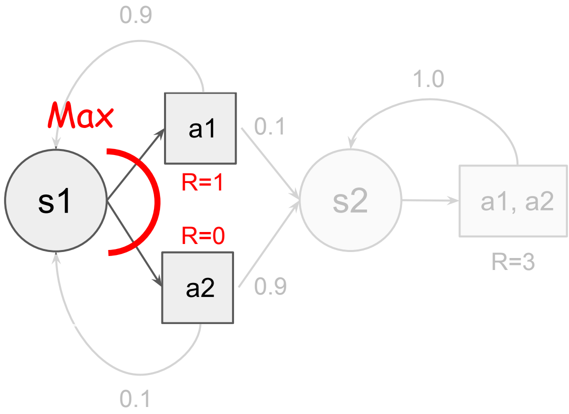
(a) Assume a finite horizon problem with horizon 1 (only 1 action is to be taken). What is the value function and the optimal action in each state?

Question



Discount factor : 0.9

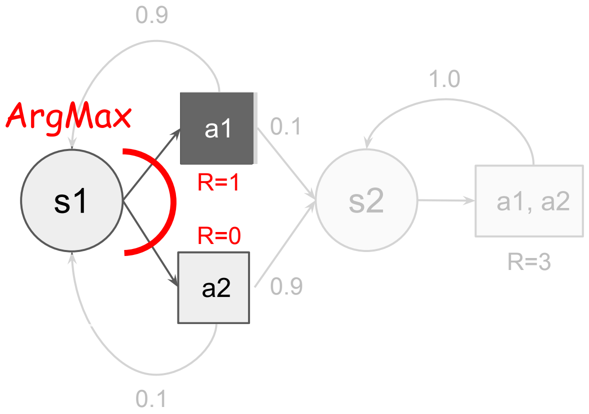
(a) Assume a finite horizon problem with horizon 1 (only 1 action is to be taken). What is the value function and the optimal action in each state?



Discount factor : 0.9

(a) Assume a finite horizon problem with horizon 1 (only 1 action is to be taken). What is the value function and the optimal action in each state?

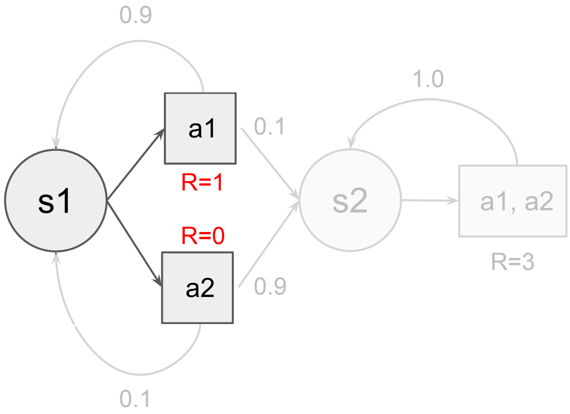
$$V_1(s_1) = 1$$

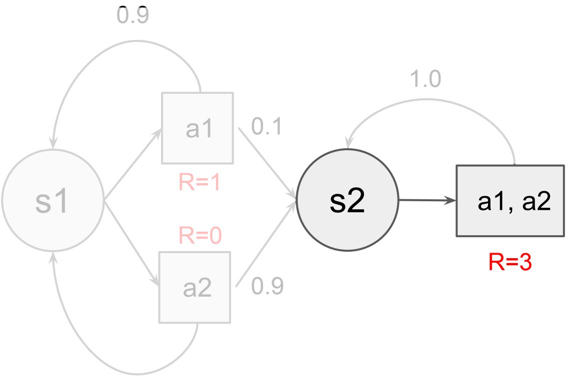


Discount factor : 0.9

(a) Assume a finite horizon problem with horizon 1 (only 1 action is to be taken). What is the value function and the optimal action in each state?

$$V_1(s_1) = 1 \qquad a^*(s_1) = a_1$$

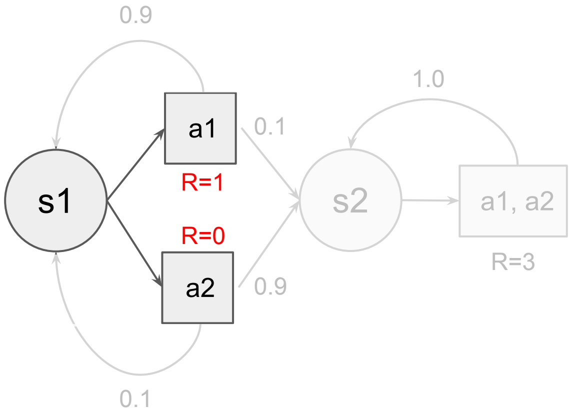


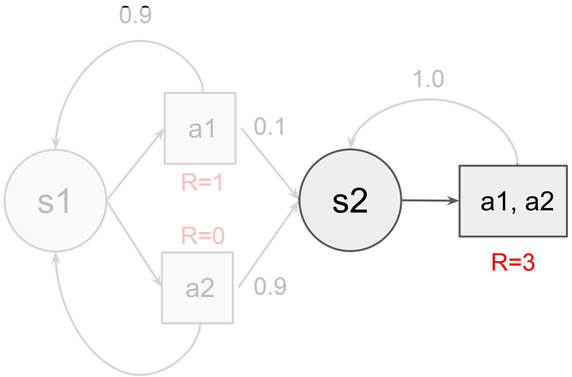


Discount factor : 0.9

(a) Assume a finite horizon problem with horizon 1 (only 1 action is to be taken). What is the value function and the optimal action in each state?

$$V_1(s_1) = 1 \qquad a^*(s_1) = a_1$$



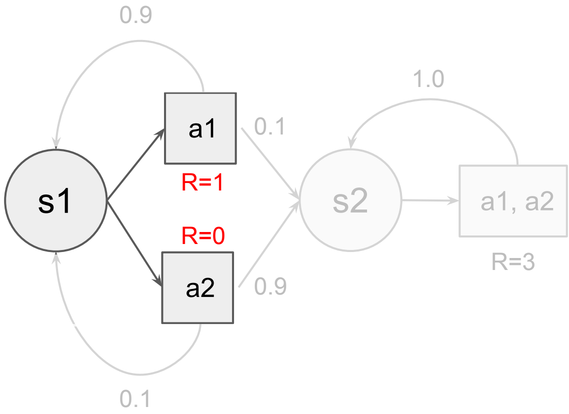


Discount factor : 0.9

(a) Assume a finite horizon problem with horizon 1 (only 1 action is to be taken). What is the value function and the optimal action in each state?

$$V_1(s_1) = 1 \qquad a^*(s_1) = a_1$$

$$V_1(s_2) = 3$$

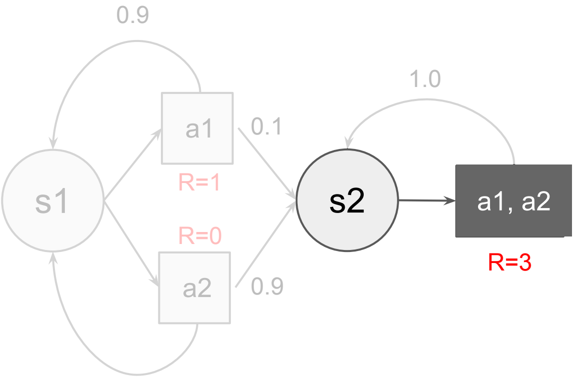


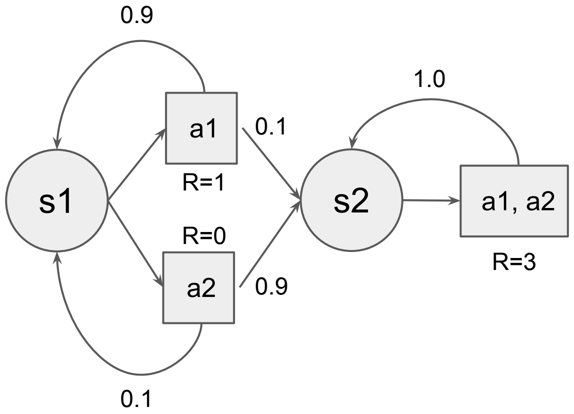
Discount factor : 0.9

(a) Assume a finite horizon problem with horizon 1 (only 1 action is to be taken). What is the value function and the optimal action in each state?

$$V_1(s_1) = 1 \qquad a^*(s_1) = a_1$$

$$V_1(s_2) = 3 \qquad a^*(s_2) = a_1 \text{ or } a_2$$

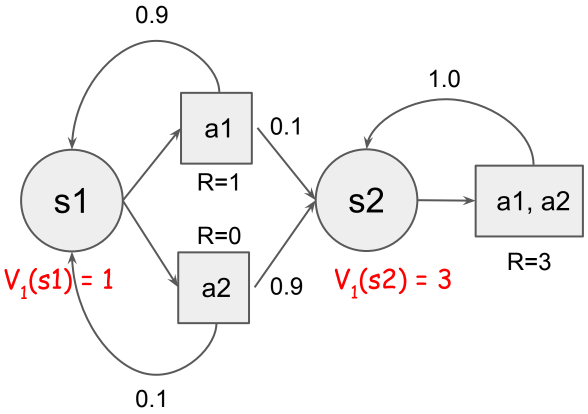




Discount factor : 0.9

(b) Assume a finite horizon problem with horizon 2 (2 actions is to be taken). What is the value function and the optimal action in each state?

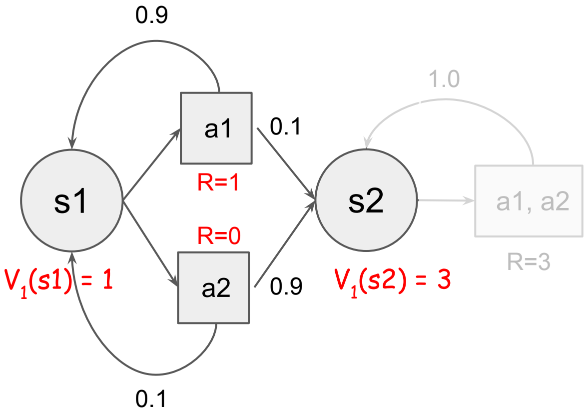
Question



Discount factor : 0.9

(b) Assume a finite horizon problem with horizon 2 (2 actions is to be taken). What is the value function and the optimal action in each state?

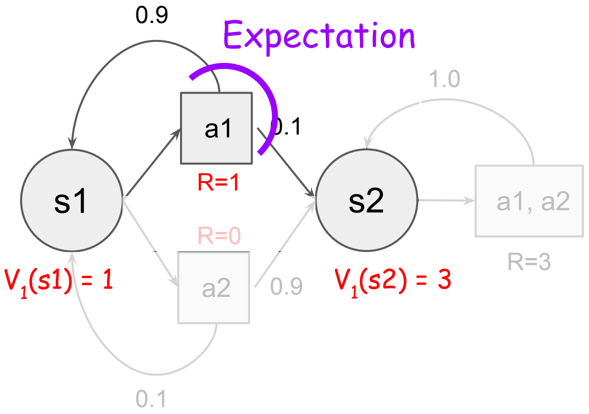
$$V_2(s_i) = \max_a \left(R(s_i, a) + \gamma \sum_{j=1}^{2} P(s_j|s_i, a) V_1(s_j)\right).$$



Discount factor : 0.9

(b) Assume a finite horizon problem with horizon 2 (2 actions is to be taken). What is the value function and the optimal action in each state?

$$V_2(s_i) = \max_a \left(R(s_i, a) + \gamma \sum_{j=1}^{2} P(s_j | s_i, a) V_1(s_j)\right).$$



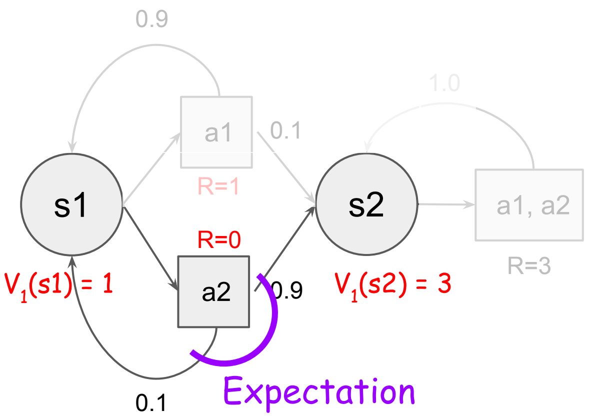
Discount factor : 0.9

(b) Assume a finite horizon problem with horizon 2 (2 actions is to be taken). What is the value function and the optimal action in each state?

$$V_2(s_i) = \max_a (R(s_i, a) + \gamma \sum_{j=1}^{2} P(s_j|s_i, a) V_1(s_j)).$$

For state 1 action 1

$$value = 1 + 0.9(0.9 * 1 + 0.1 * 3) = 2.08.$$



Discount factor : 0.9

(b) Assume a finite horizon problem with horizon 2 (2 actions is to be taken). What is the value function and the optimal action in each state?

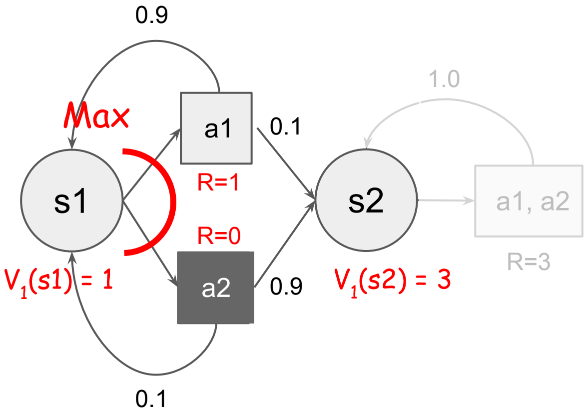
$$V_2(s_i) = \max_a (R(s_i, a) + \gamma \sum_{j=1}^{2} P(s_j|s_i, a) V_1(s_j)).$$

For state 1 action 1

$$value = 1 + 0.9(0.9 * 1 + 0.1 * 3) = 2.08.$$

For state 1 action 2

$$value = 0 + 0.9(0.9 * 3 + 0.1 * 1) = 2.52.$$



Discount factor : 0.9

(b) Assume a finite horizon problem with horizon 2 (2 actions is to be taken). What is the value function and the optimal action in each state?

$$V_2(s_i) = \max_a (R(s_i, a) + \gamma \sum_{j=1}^{2} P(s_j|s_i, a) V_1(s_j)).$$

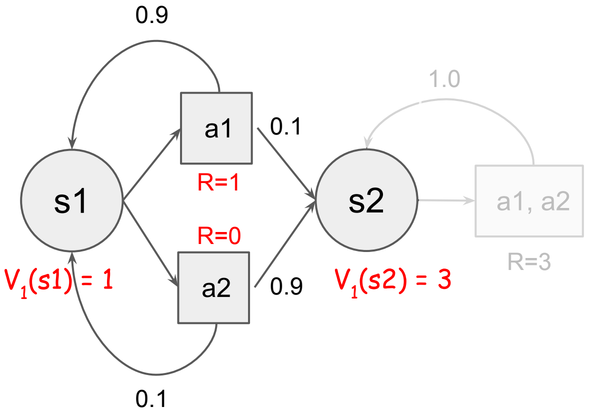
For state 1 action 1

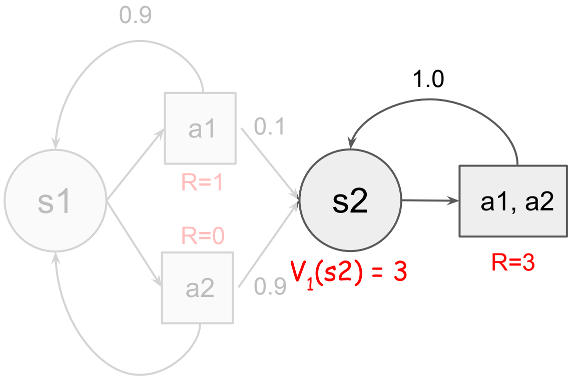
$$value = 1 + 0.9(0.9 * 1 + 0.1 * 3) = 2.08.$$

For state 1 action 2

$$value = 0 + 0.9(0.9 * 3 + 0.1 * 1) = 2.52.$$

Max = 2.52 (action 2)





Discount factor : 0.9

(b) Assume a finite horizon problem with horizon 2 (2 actions is to be taken). What is the value function and the optimal action in each state?

$$V_2(s_i) = \max_a (R(s_i, a) + \gamma \sum_{j=1}^{2} P(s_j | s_i, a) V_1(s_j)).$$

For state 1 action 1

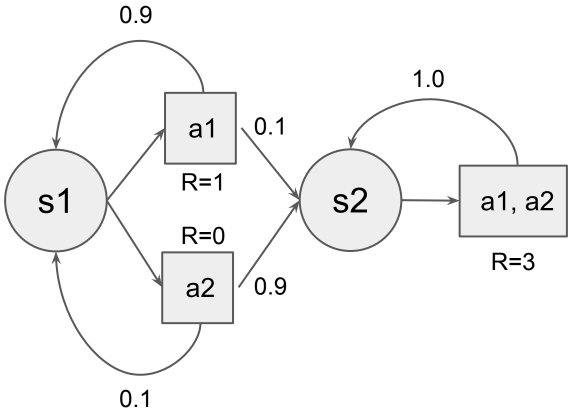
$$value = 1 + 0.9(0.9 * 1 + 0.1 * 3) = 2.08.$$

For state 1 action 2

$$value = 0 + 0.9(0.9 * 3 + 0.1 * 1) = 2.52.$$

Max = 2.52 (action 2)

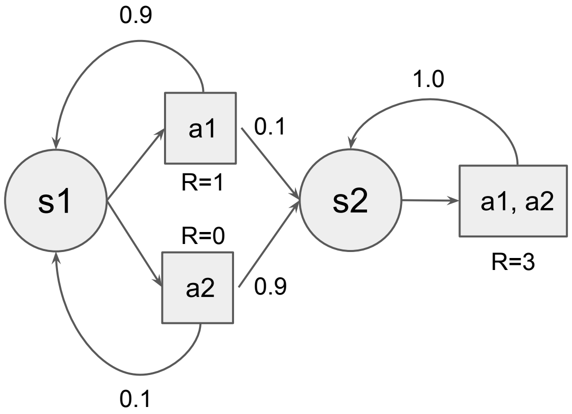
$$value = 3 + 0.9 * 3 = 5.7.$$



Discount factor : 0.9

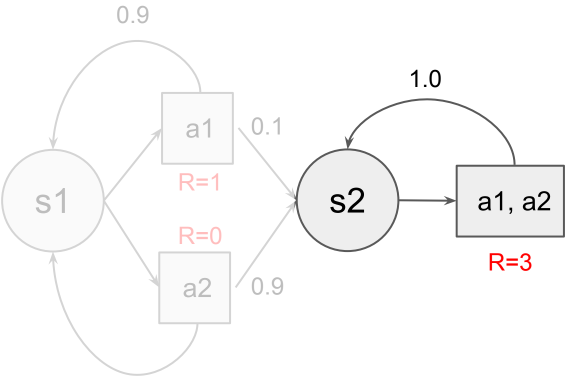
(c) What is the optimal infinite horizon policy?

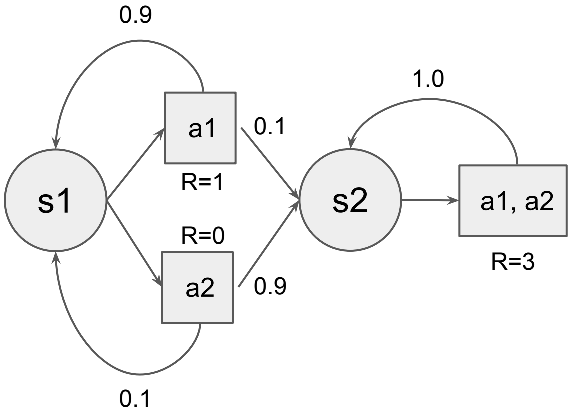
Question



Discount factor : 0.9

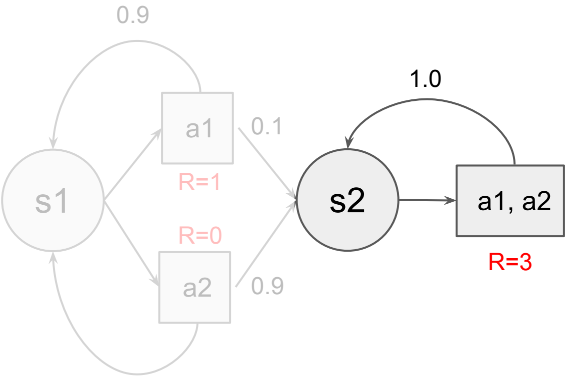
(c) What is the optimal infinite horizon policy?



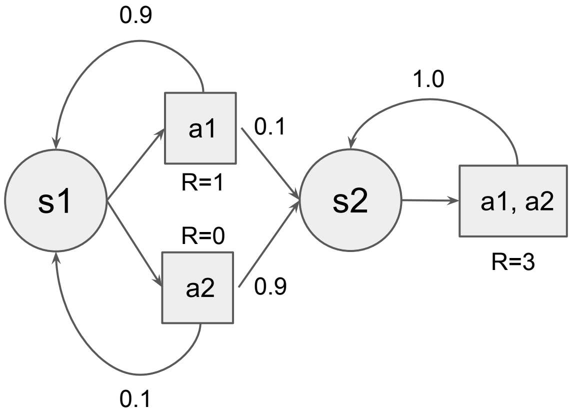


Discount factor : 0.9

(c) What is the optimal infinite horizon policy?

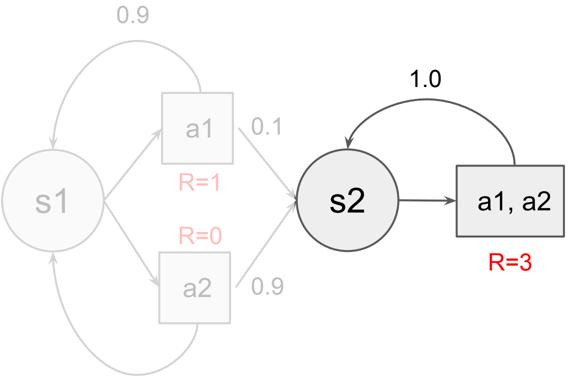


$$V(s_2) = 3 + 0.9(3 + 0.9(\ldots))$$

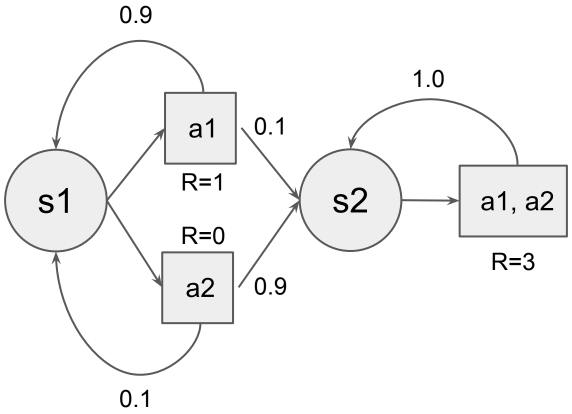


Discount factor : 0.9

(c) What is the optimal infinite horizon policy?

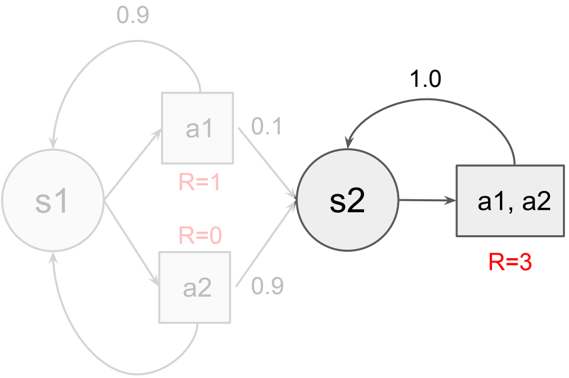


$$V(s_2) = 3 + \underbrace{0.9(3 + 0.9(...))}_{\text{Geometric series}}$$

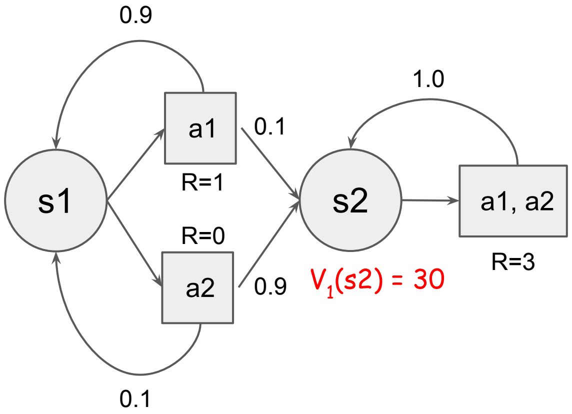


Discount factor : 0.9

(c) What is the optimal infinite horizon policy?

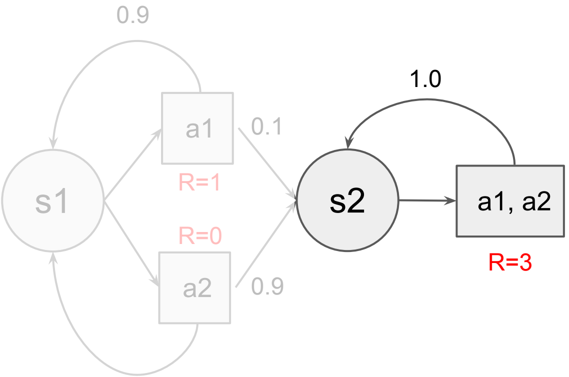


$$V(s_2) = \underbrace{3 + 0.9(3 + 0.9(\ldots))}_{\text{Geometric series}} = \frac{3}{1-0.9} = \frac{3}{0.1} = 30$$

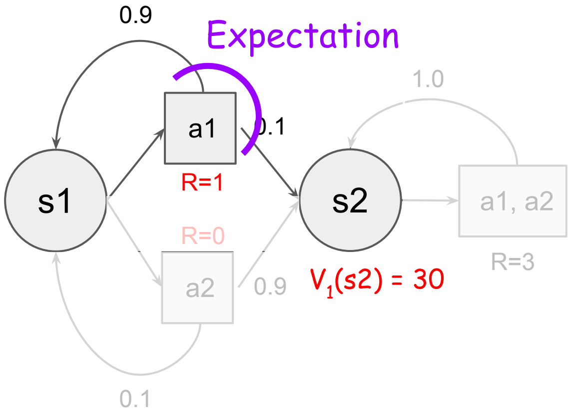


Discount factor : 0.9

(c) What is the optimal infinite horizon policy?



$$V(s_2) = \underbrace{3 + 0.9(3 + 0.9(\ldots))}_{\text{Geometric series}} = \frac{3}{1-0.9} = \frac{3}{0.1} = 30$$



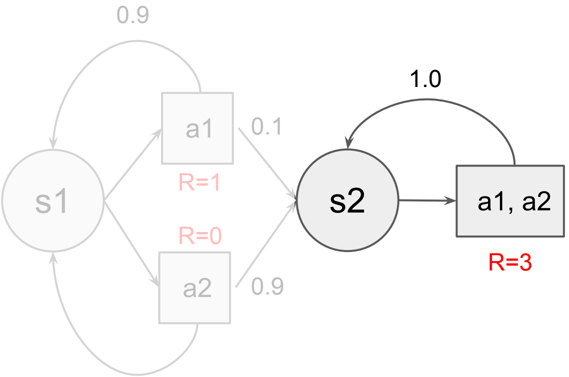
Discount factor : 0.9

(c) What is the optimal infinite horizon policy?

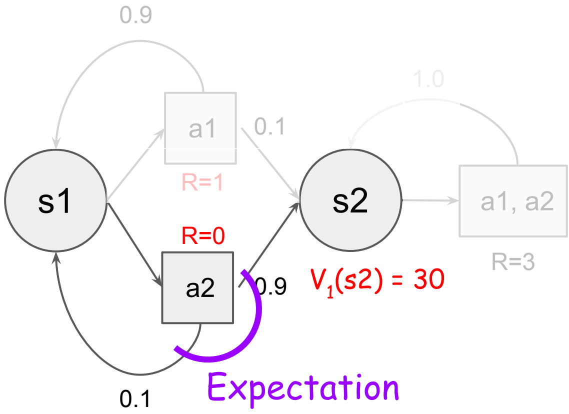
If action a_1 is taken, the value of the policy must satisfy

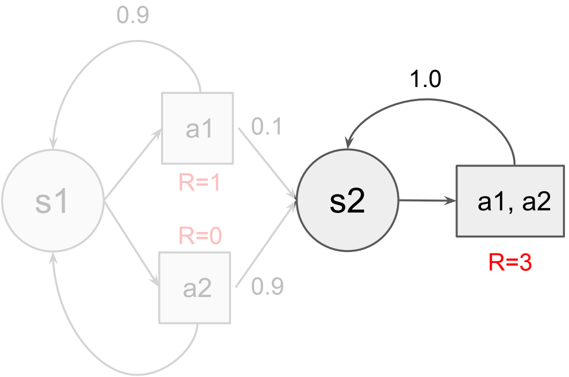
$$V(s_1) = 1 + 0.9(0.9V(s_1) + 0.1 * 30)$$

giving $V(s_1) = 19.47$.



$$V(s_2) = \underbrace{3 + 0.9(3 + 0.9(...))}_{\text{Geometric series}} = \frac{3}{1-0.9} = \frac{3}{0.1} = 30$$





Discount factor : 0.9

(c) What is the optimal infinite horizon policy?

If action a_1 is taken, the value of the policy must satisfy

$$V(s_1) = 1 + 0.9(0.9V(s_1) + 0.1 * 30)$$

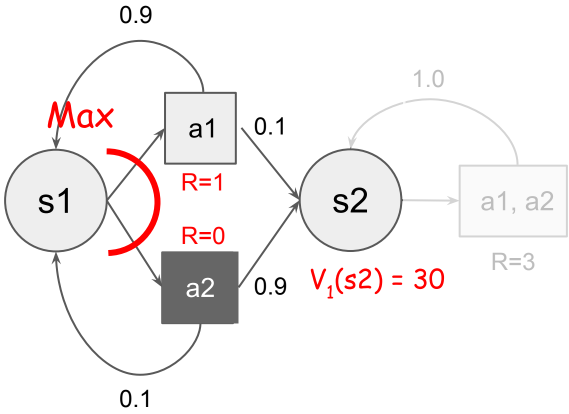
giving $V(s_1) = 19.47$.

If action a_2 is taken, the value of the policy must satisfy

$$V(s_1) = 0 + 0.9(0.9 * 30 + 0.1V(s_1))$$

giving $V(s_1) = 26.7$.

$$V(s_2) = 3 + \underbrace{0.9(3 + 0.9(...))}_{\text{Geometric series}} = \frac{3}{1-0.9} = \frac{3}{0.1} = 30$$



Discount factor : 0.9

(c) What is the optimal infinite horizon policy?

If action a_1 is taken, the value of the policy must satisfy

$$V(s_1) = 1 + 0.9(0.9V(s_1) + 0.1 * 30)$$

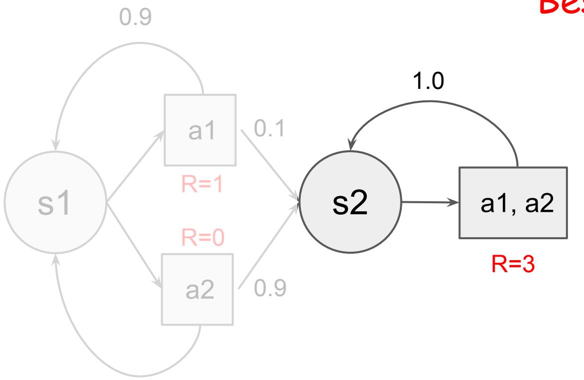
giving $V(s_1) = 19.47$.

Best

If action a_2 is taken, the value of the policy must satisfy

$$V(s_1) = 0 + 0.9(0.9 * 30 + 0.1V(s_1))$$

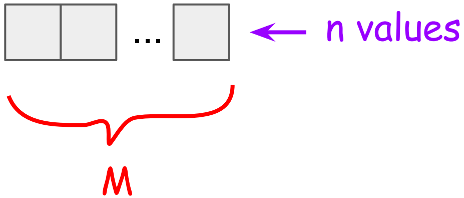
giving $V(s_1) = 26.7$.



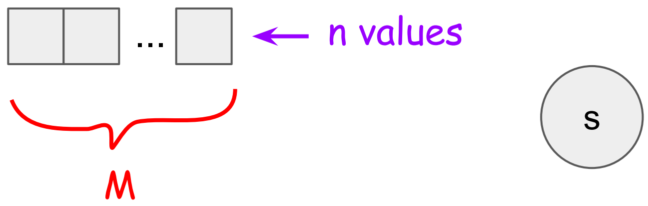
$$V(s_2) = \underbrace{3 + 0.9(3 + 0.9(...))}_{\text{Geometric series}} = \frac{3}{1 - 0.9} = \frac{3}{0.1} = 30$$

Third

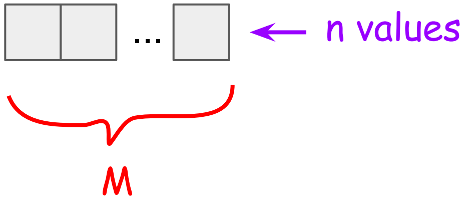
State:

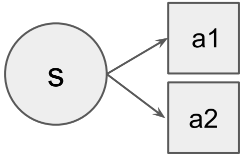


State:

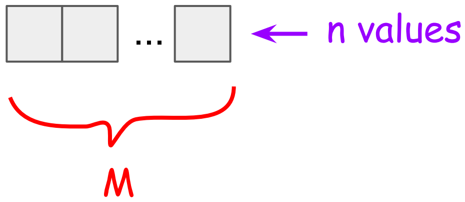


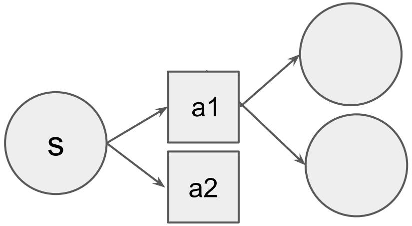
State:

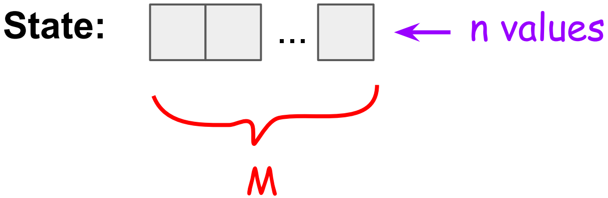


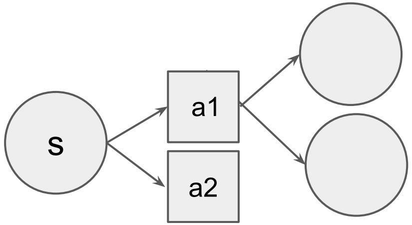


State:



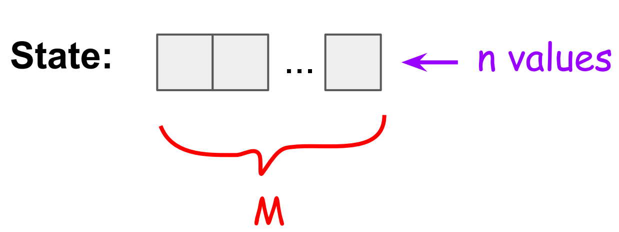


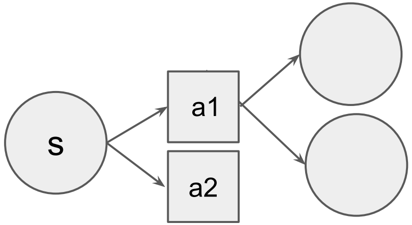




a) What is the size of the state space of this MDP? Can this MDP be efficiently solvable with value iteration as M grows?

Question

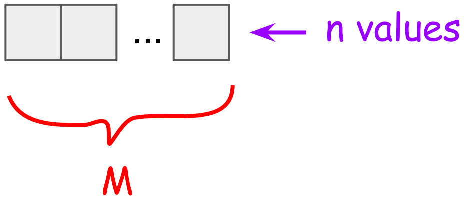


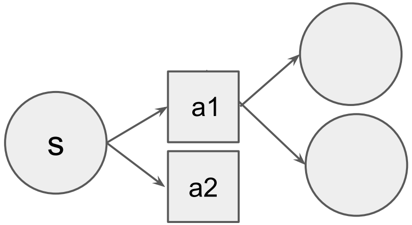


a) What is the size of the state space of this MDP? Can this MDP be efficiently solvable with value iteration as M grows?

$$n^M$$

State:



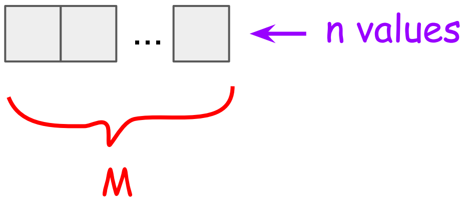


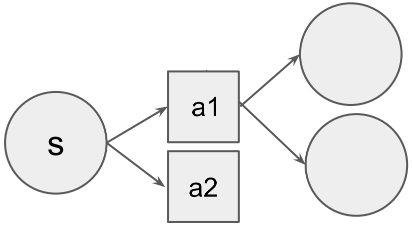
a) What is the size of the state space of this MDP? Can this MDP be efficiently solvable with value iteration as M grows?

n^M

Value iteration: runtime exponential in M (not good!)

State:

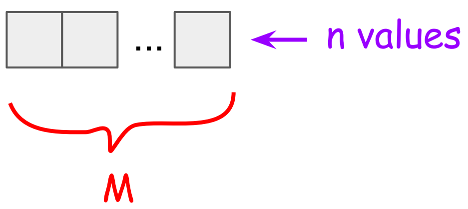


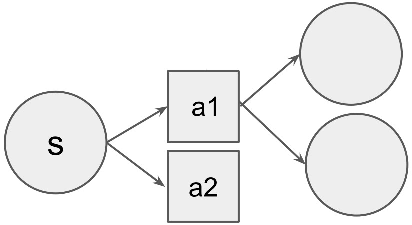


b) A search tree of depth D (number of actions from the root to any leaf is D) is constructed from an initial state s. What is the size of the search tree (the number of nodes and edges) as a function of M and D, in O-notation? Can online search be done efficiently as M grows if D is a fixed small constant?

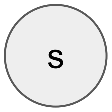
Question

State:

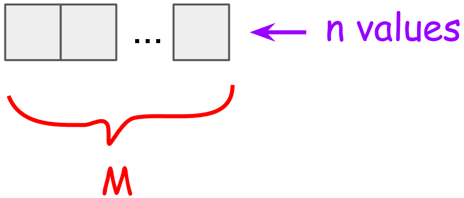


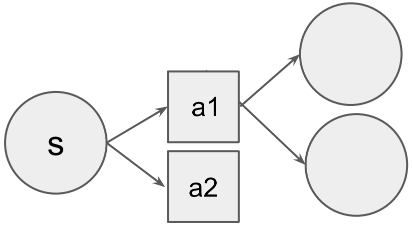


b) A search tree of depth D (number of actions from the root to any leaf is D) is constructed from an initial state s. What is the size of the search tree (the number of nodes and edges) as a function of M and D, in O-notation? Can online search be done efficiently as M grows if D is a fixed small constant?

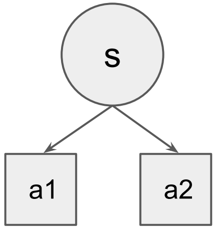


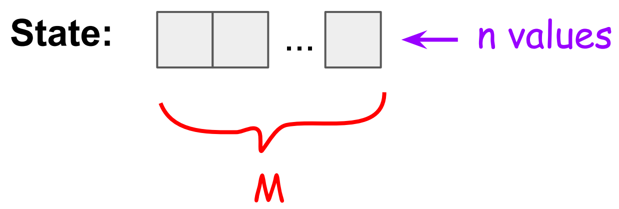
State:

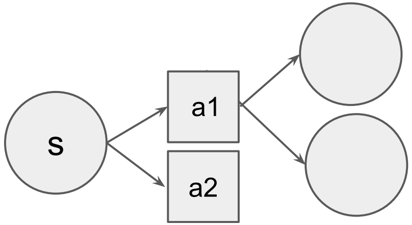




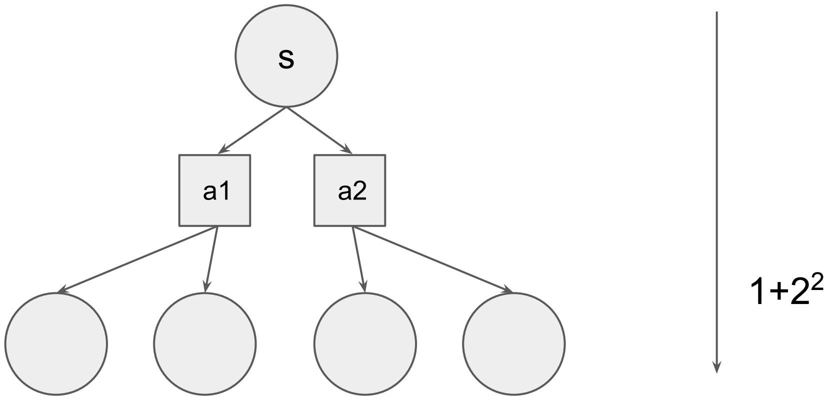
b) A search tree of depth D (number of actions from the root to any leaf is D) is constructed from an initial state s. What is the size of the search tree (the number of nodes and edges) as a function of M and D, in O-notation? Can online search be done efficiently as M grows if D is a fixed small constant?



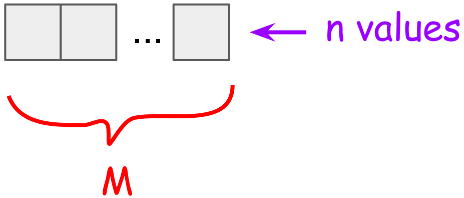


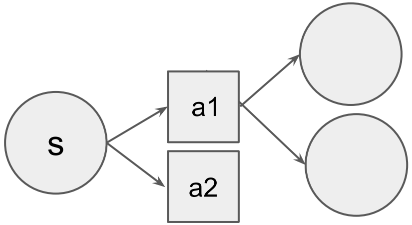


b) A search tree of depth D (number of actions from the root to any leaf is D) is constructed from an initial state s. What is the size of the search tree (the number of nodes and edges) as a function of M and D, in O-notation? Can online search be done efficiently as M grows if D is a fixed small constant?

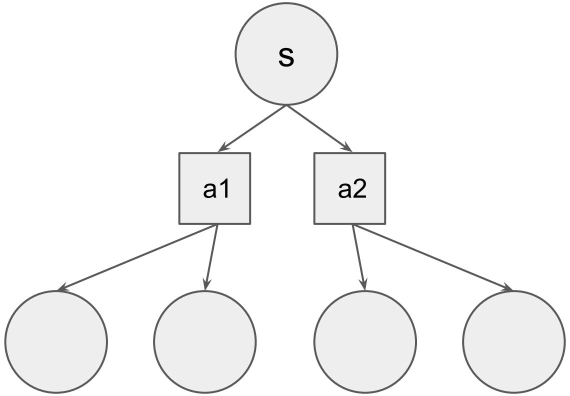


State:



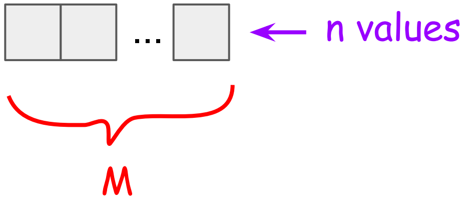


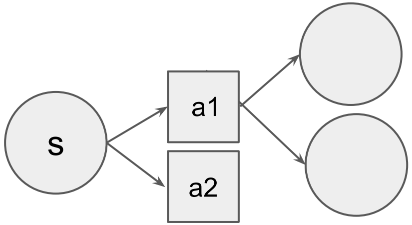
b) A search tree of depth D (number of actions from the root to any leaf is D) is constructed from an initial state s. What is the size of the search tree (the number of nodes and edges) as a function of M and D, in O-notation? Can online search be done efficiently as M grows if D is a fixed small constant?



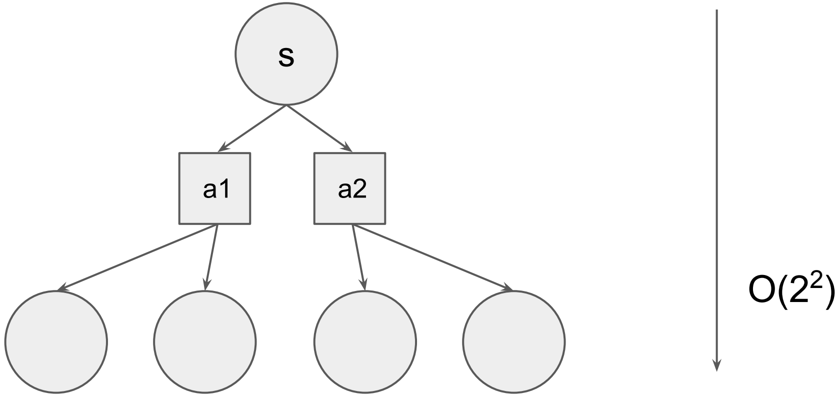
$1+2^2 = O(2^2)$

State:

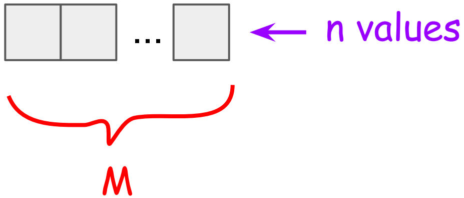


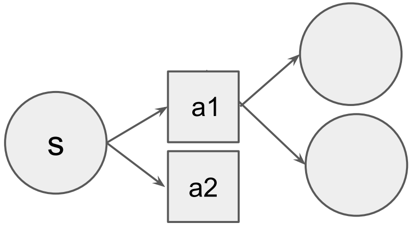


b) A search tree of depth D (number of actions from the root to any leaf is D) is constructed from an initial state s. What is the size of the search tree (the number of nodes and edges) as a function of M and D, in O-notation? Can online search be done efficiently as M grows if D is a fixed small constant?

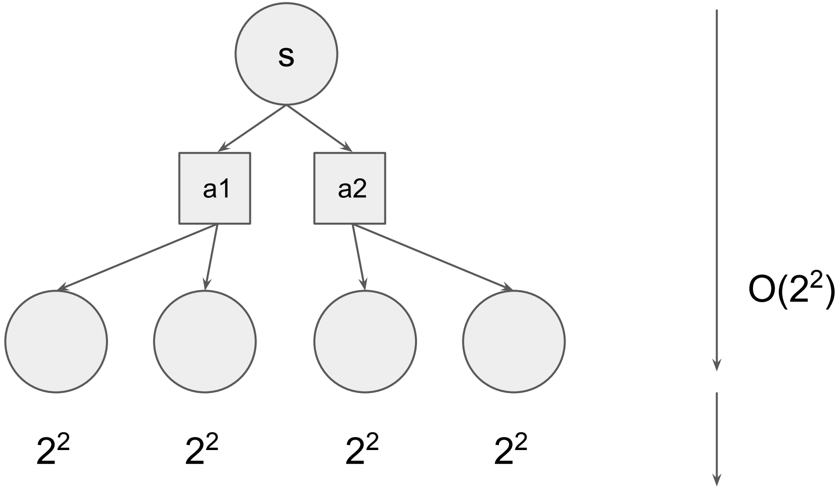


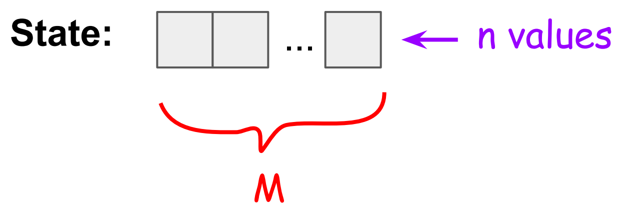
State:

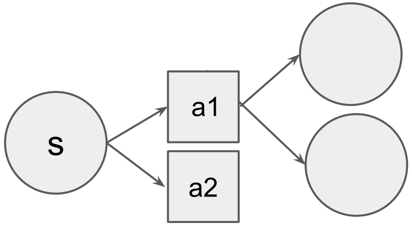




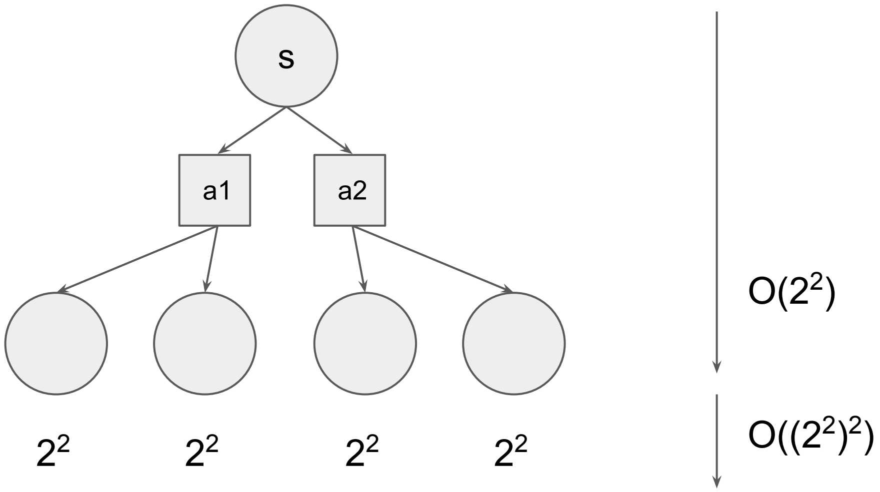
b) A search tree of depth D (number of actions from the root to any leaf is D) is constructed from an initial state s. What is the size of the search tree (the number of nodes and edges) as a function of M and D, in O-notation? Can online search be done efficiently as M grows if D is a fixed small constant?



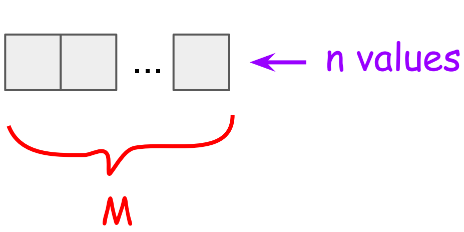


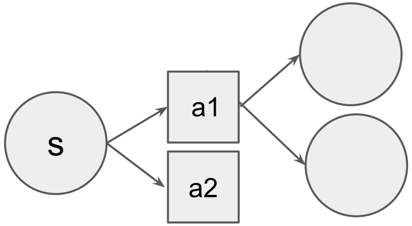


b) A search tree of depth D (number of actions from the root to any leaf is D) is constructed from an initial state s. What is the size of the search tree (the number of nodes and edges) as a function of M and D, in O-notation? Can online search be done efficiently as M grows if D is a fixed small constant?

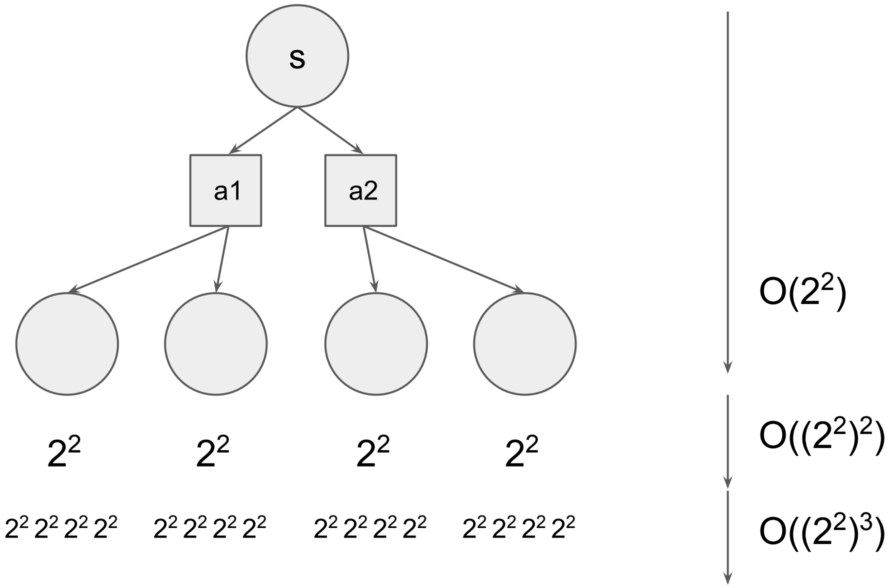


State:

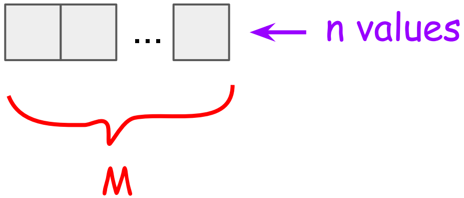


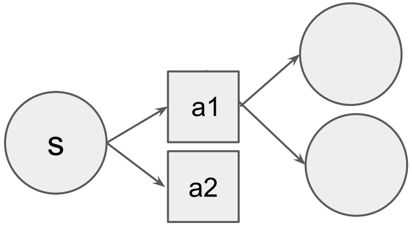


b) A search tree of depth D (number of actions from the root to any leaf is D) is constructed from an initial state s. What is the size of the search tree (the number of nodes and edges) as a function of M and D, in O-notation? Can online search be done efficiently as M grows if D is a fixed small constant?

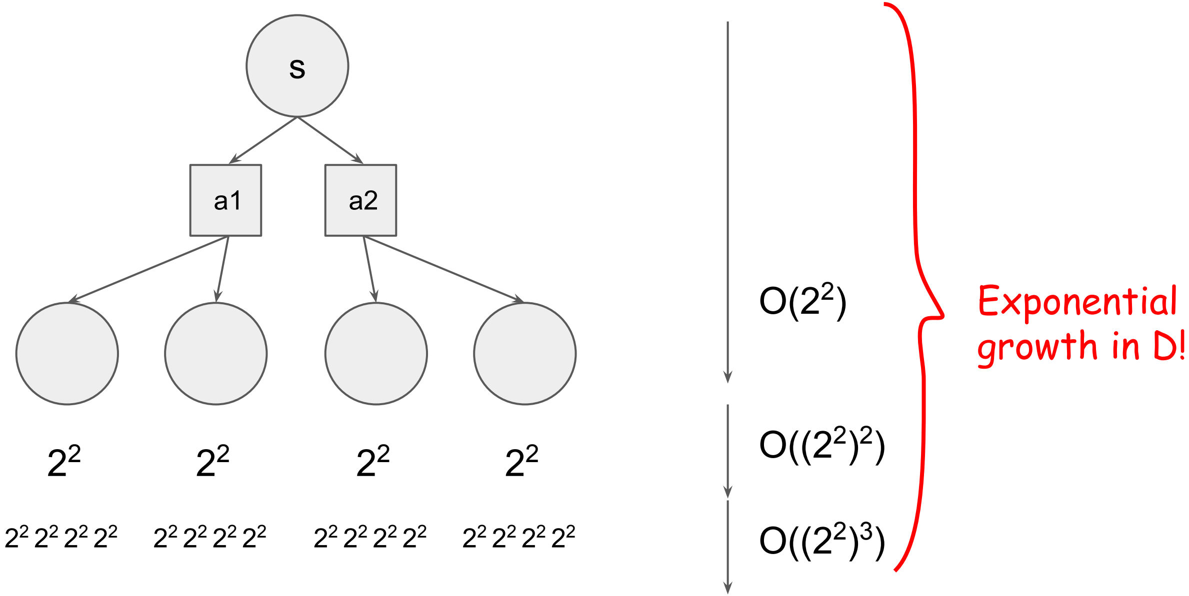


State:

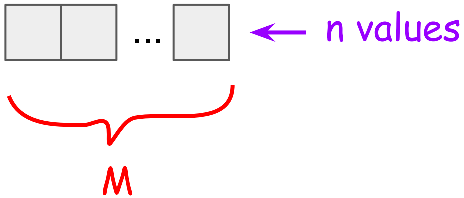


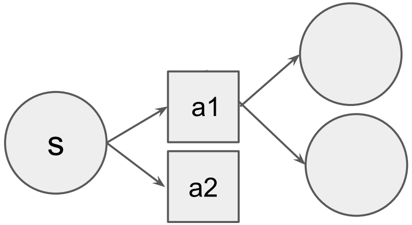


b) A search tree of depth D (number of actions from the root to any leaf is D) is constructed from an initial state s. What is the size of the search tree (the number of nodes and edges) as a function of M and D, in O-notation? Can online search be done efficiently as M grows if D is a fixed small constant?

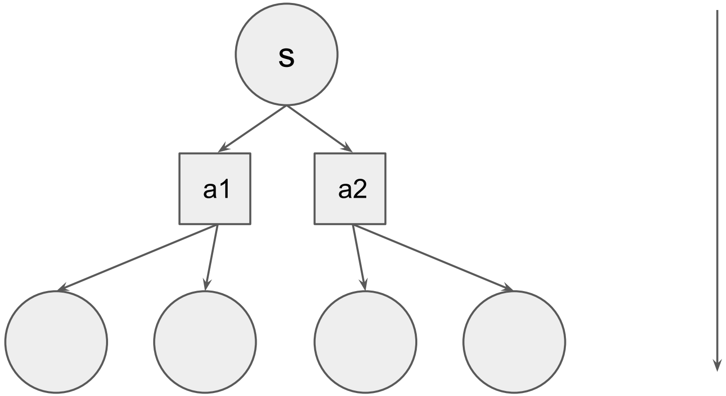


State:



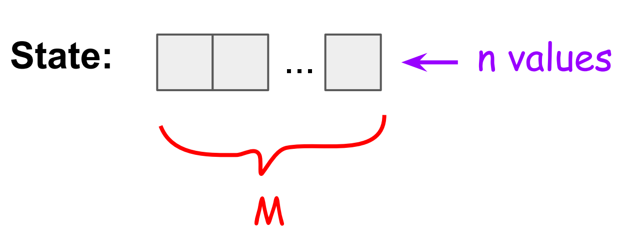


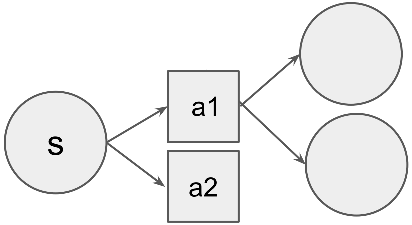
b) A search tree of depth D (number of actions from the root to any leaf is D) is constructed from an initial state s. What is the size of the search tree (the number of nodes and edges) as a function of M and D, in O-notation? Can online search be done efficiently as M grows if D is a fixed small constant?



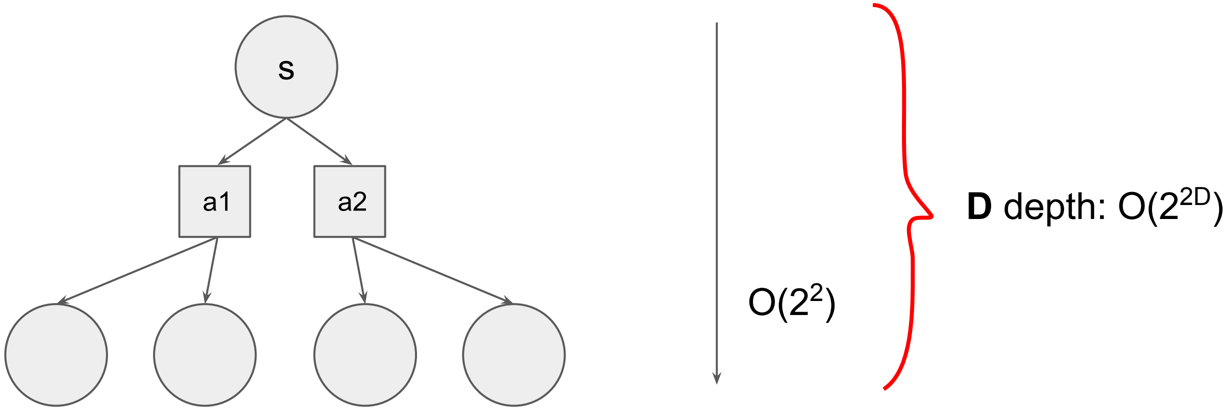
$O(2^2)$

D depth: $O((2^2)^D)$

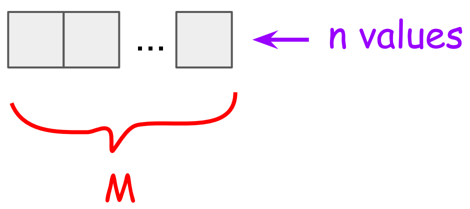


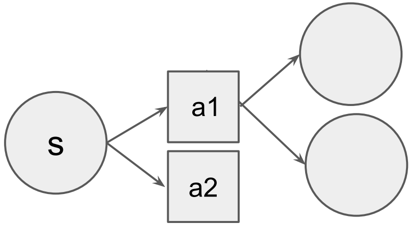


b) A search tree of depth D (number of actions from the root to any leaf is D) is constructed from an initial state s. What is the size of the search tree (the number of nodes and edges) as a function of M and D, in O-notation? Can online search be done efficiently as M grows if D is a fixed small constant?

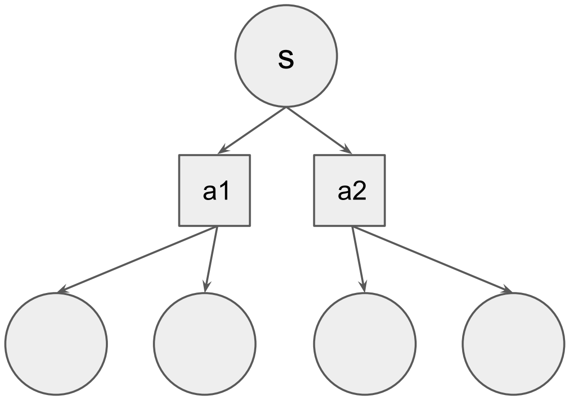


State:





b) A search tree of depth D (number of actions from the root to any leaf is D) is constructed from an initial state s. What is the size of the search tree (the number of nodes and edges) as a function of M and D, in O-notation? Can online search be done efficiently as M grows if D is a fixed small constant?

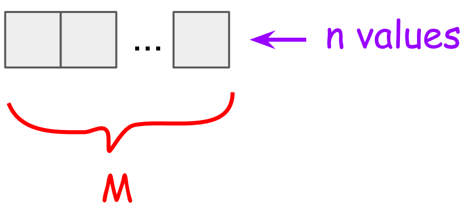


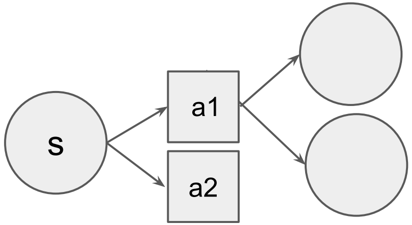
$O(2^2)$

D depth: $O(2^{2D})$

Doesn't depend on M, if
D is small then all good!

State:

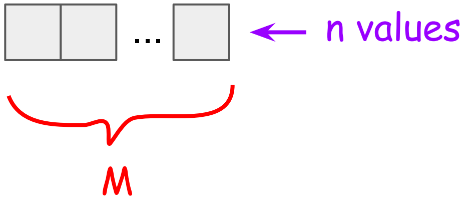


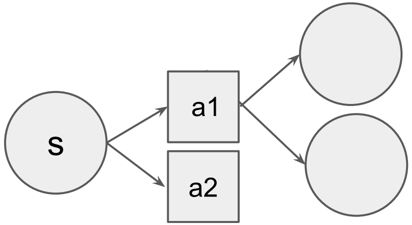


c) MCTS is used for solving this MDP. What is the size of the search tree if T trials of MTCS is performed up to a search depth of D, as a function of M, D and T in O-notation?

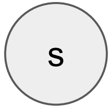
Question

State:

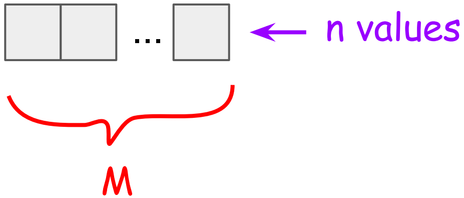


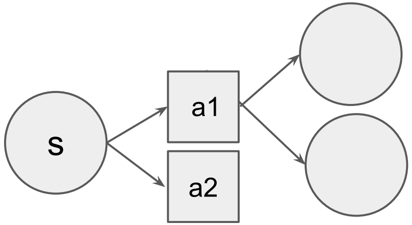


c) MCTS is used for solving this MDP. What is the size of the search tree if T trials of MTCS is performed up to a search depth of D, as a function of M, D and T in O-notation?

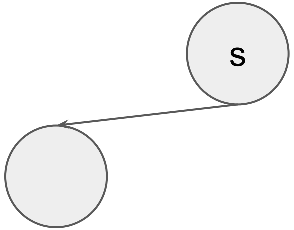


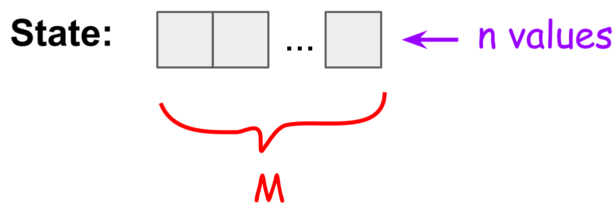
State:

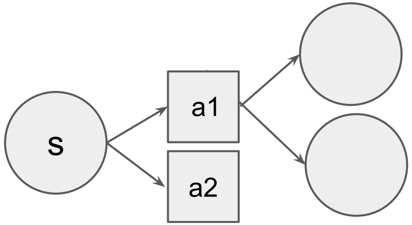




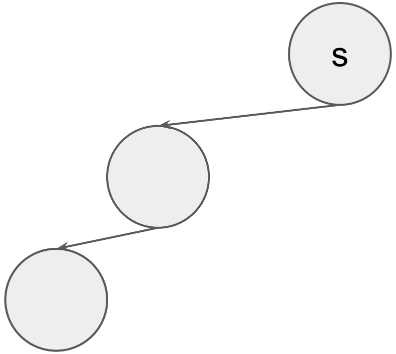
c) MCTS is used for solving this MDP. What is the size of the search tree if T trials of MTCS is performed up to a search depth of D, as a function of M, D and T in O-notation?

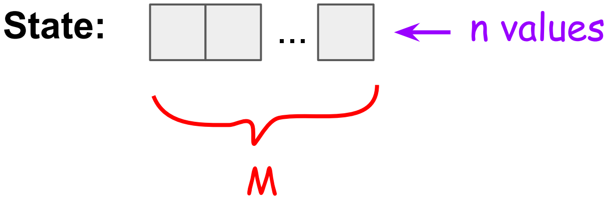


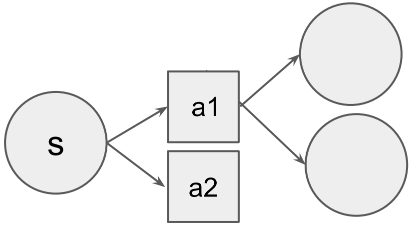




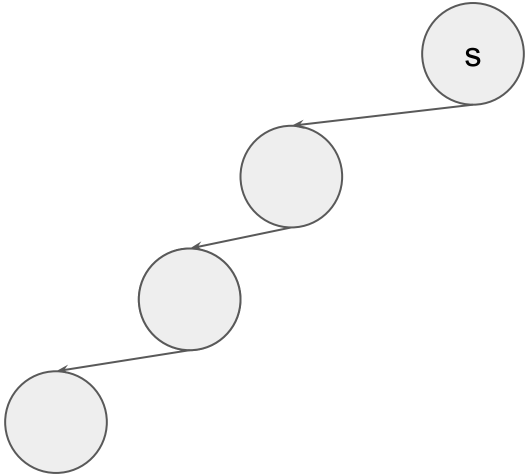
c) MCTS is used for solving this MDP. What is the size of the search tree if T trials of MTCS is performed up to a search depth of D, as a function of M, D and T in O-notation?

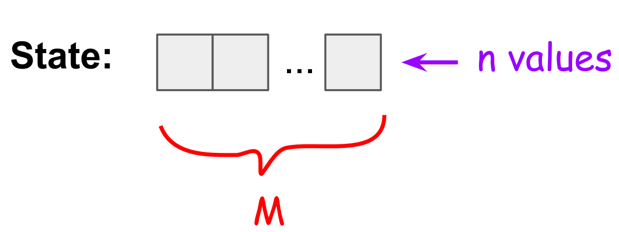


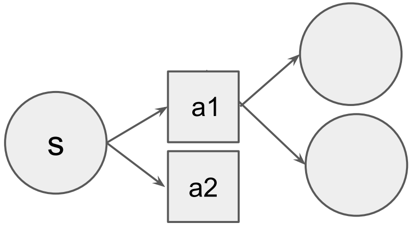




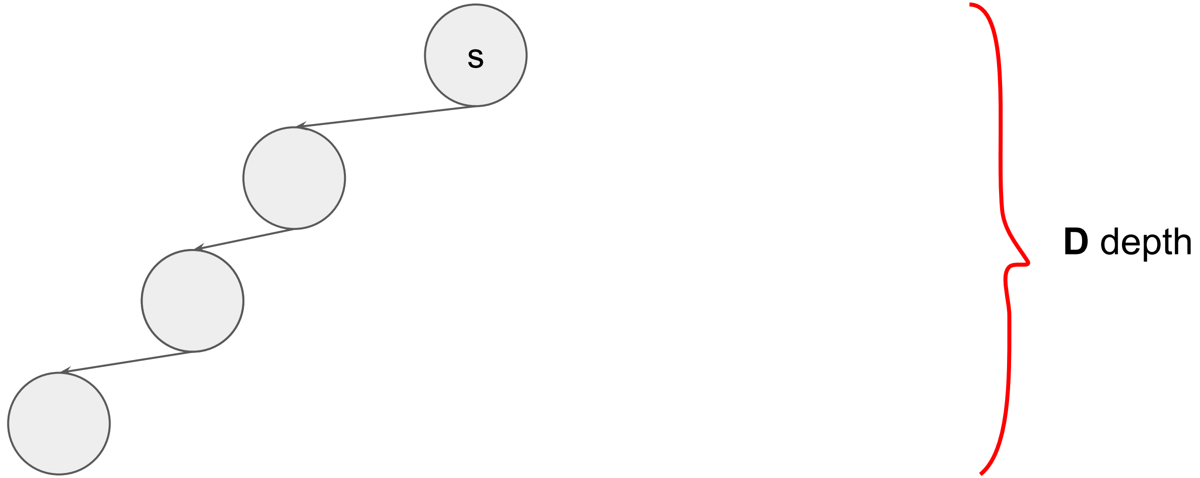
c) MCTS is used for solving this MDP. What is the size of the search tree if T trials of MTCS is performed up to a search depth of D, as a function of M, D and T in O-notation?

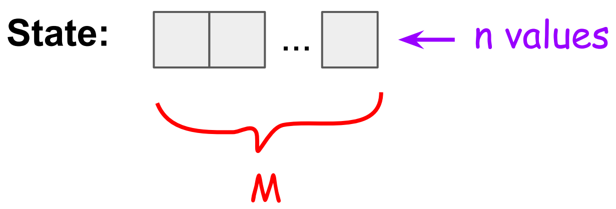


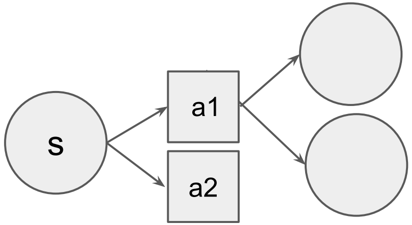




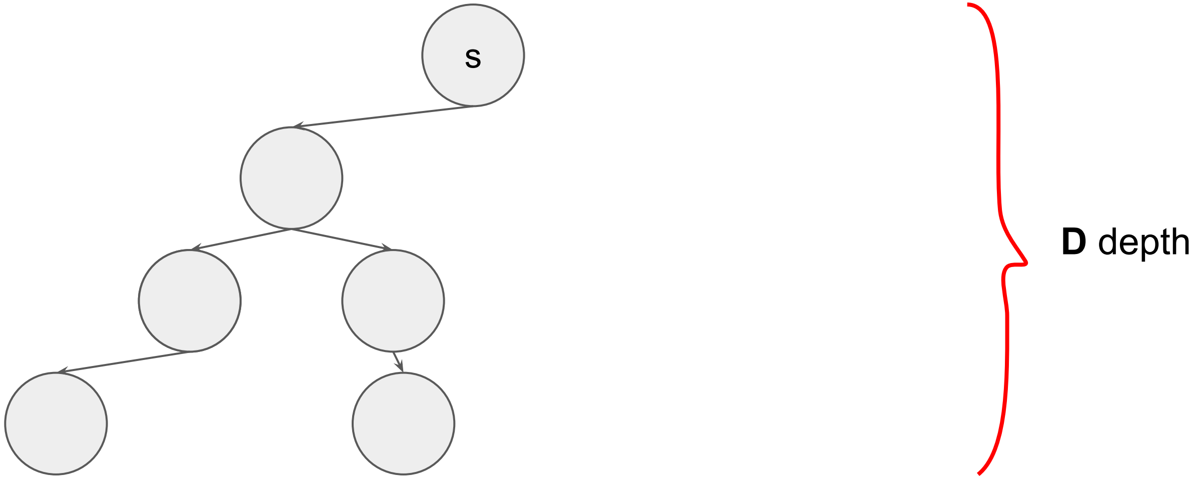
c) MCTS is used for solving this MDP. What is the size of the search tree if T trials of MTCS is performed up to a search depth of D, as a function of M, D and T in O-notation?



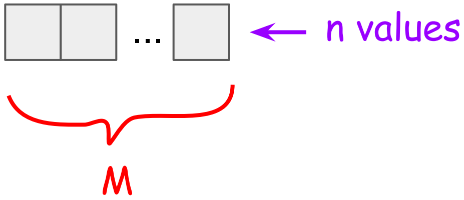


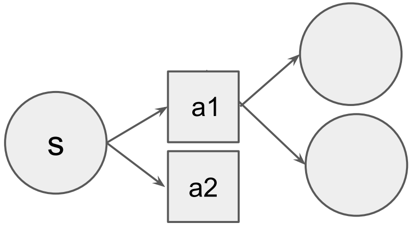


c) MCTS is used for solving this MDP. What is the size of the search tree if T trials of MTCS is performed up to a search depth of D, as a function of M, D and T in O-notation?

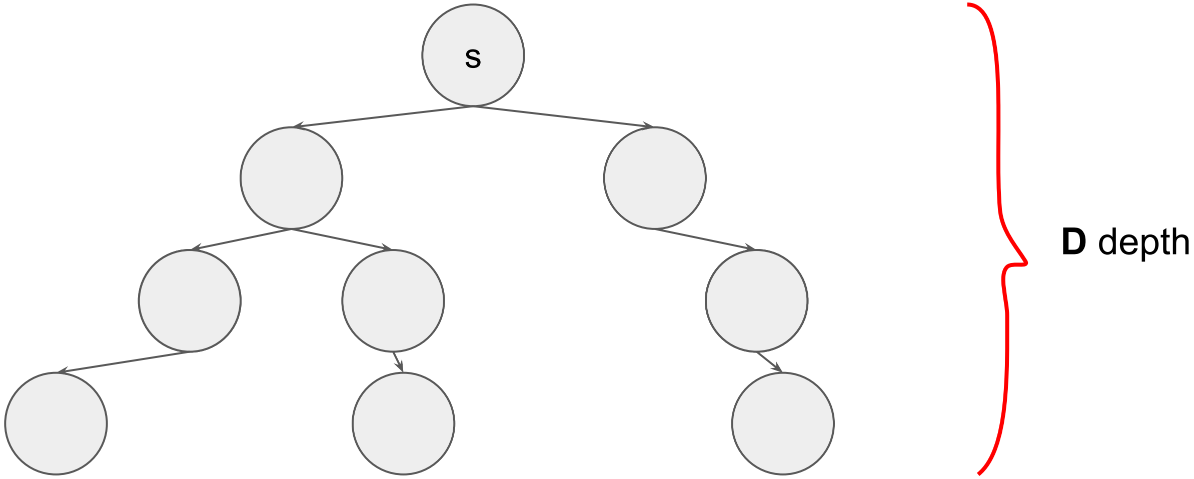


State:

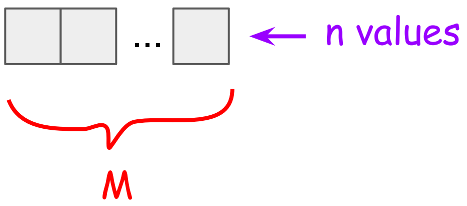


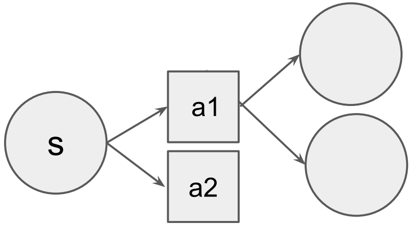


c) MCTS is used for solving this MDP. What is the size of the search tree if T trials of MTCS is performed up to a search depth of D, as a function of M, D and T in O-notation?

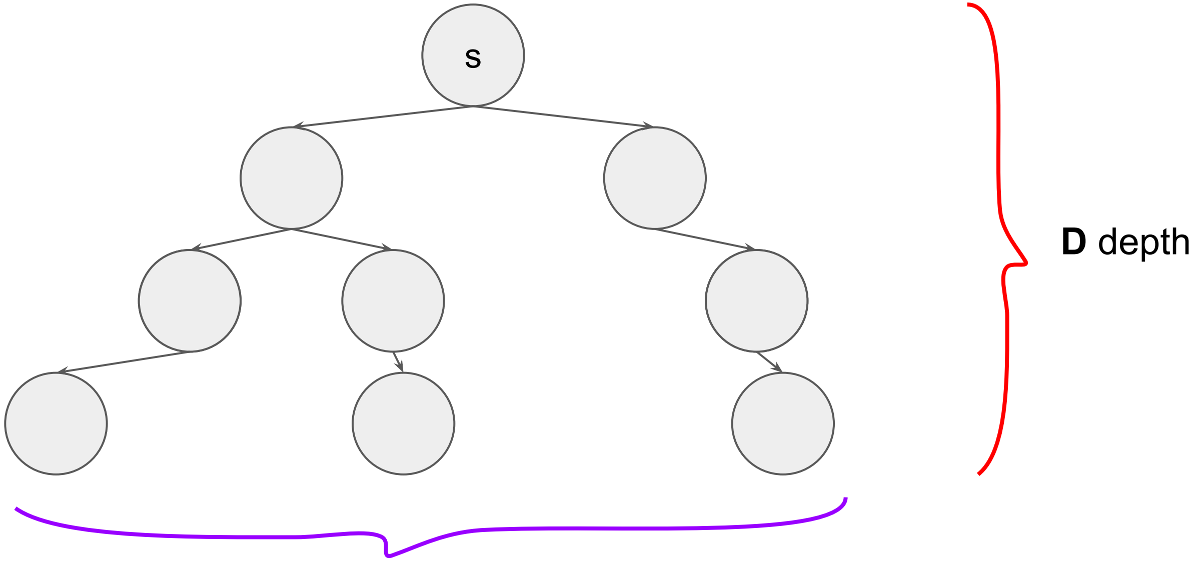


State:

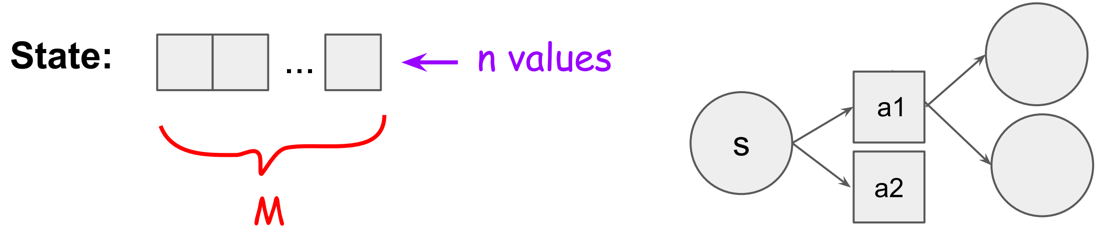




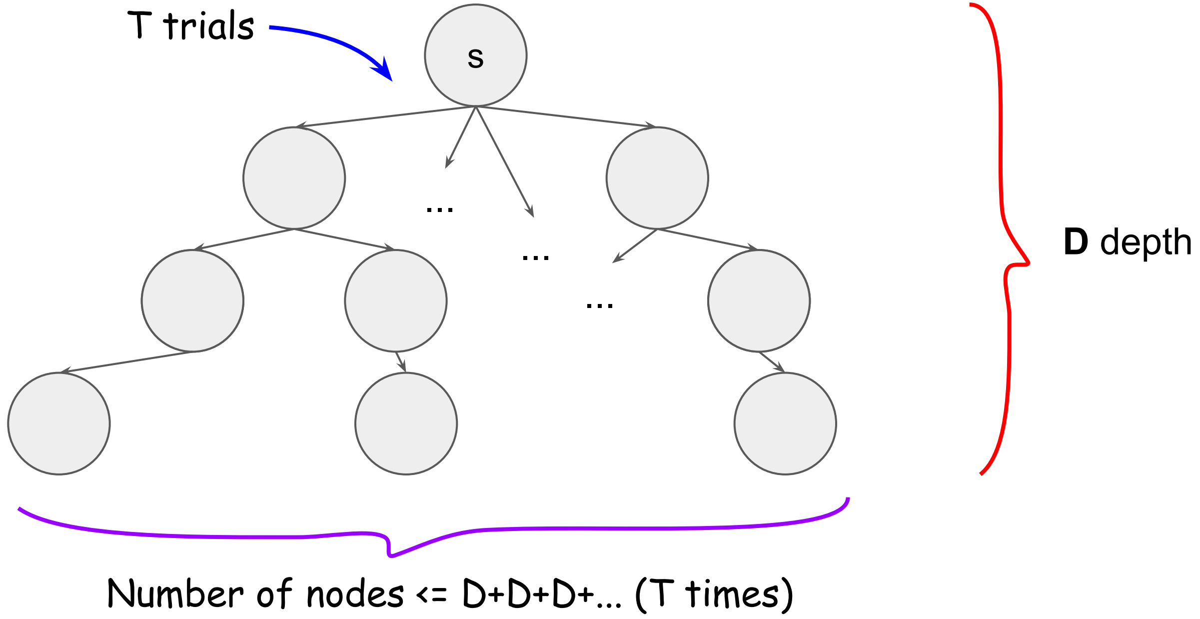
c) MCTS is used for solving this MDP. What is the size of the search tree if T trials of MTCS is performed up to a search depth of D, as a function of M, D and T in O-notation?



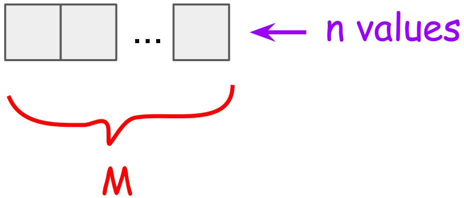
Number of nodes <= D+D+D

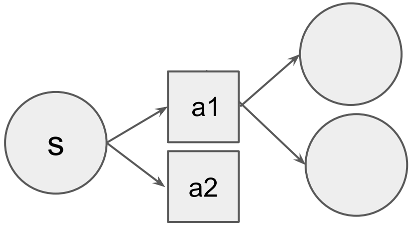


c) MCTS is used for solving this MDP. What is the size of the search tree if T trials of MTCS is performed up to a search depth of D, as a function of M, D and T in O-notation?



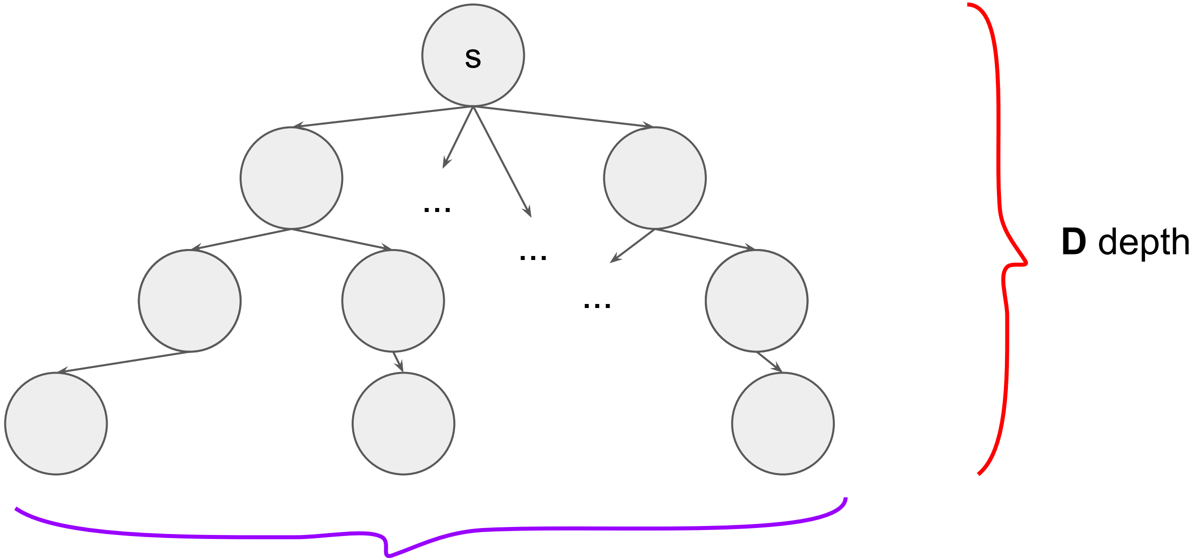
State:



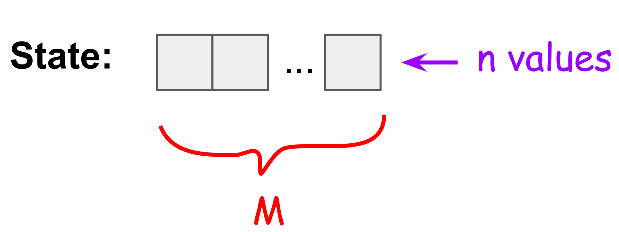


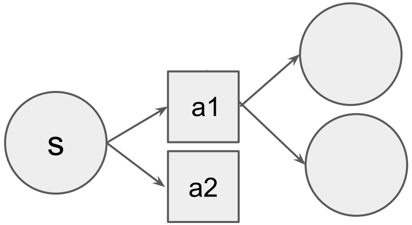
c) MCTS is used for solving this MDP. What is the size of the search tree if T trials of MTCS is performed up to a search depth of D, as a function of M, D and T in O-notation?

$$O(DT)$$



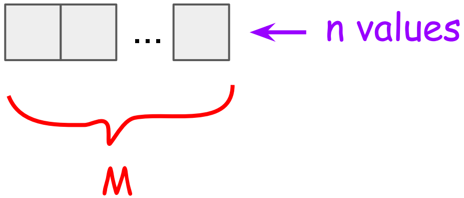
Number of nodes <= D+D+D+... (T times) = O(DT)

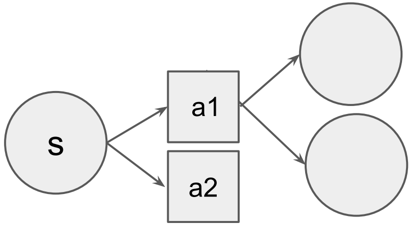




d) Consider a search tree where the reward is zero everywhere except at the leaves. When a MCTS trial goes through a node, we say that an action at the node wins if the trial ends in a leaf with reward 1. Consider an MCTS simulation where a node has been visited 16 times and has two actions, A and B. Action A has a won 2 out 4 times whereas action B has won 8 out of 12 times. Which action will the MCTS algorithm chose given the exploration parameter c is set to 1? Give the values of π_{UCT} for the node (consider log base 2 in UCT bound).

State:



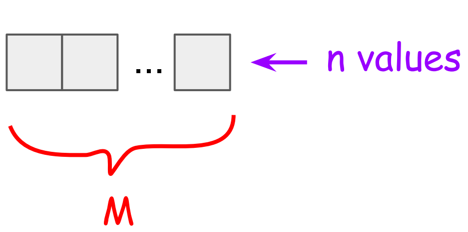


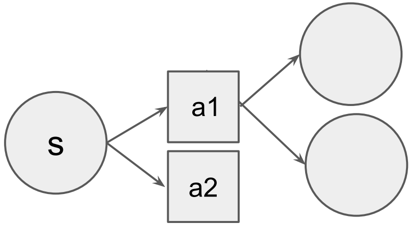
d) Consider a search tree where the reward is zero everywhere except at the leaves. When a MCTS trial goes through a node, we say that an action at the node wins if the trial ends in a leaf with reward 1. Consider an MCTS simulation where a node has been visited 16 times and has two actions, A and B. Action A has a won 2 out 4 times whereas action B has won 8 out of 12 times. Which action will the MCTS algorithm chose given the exploration parameter c is set to 1? Give the values of π_{UCT} for the node (consider log base 2 in UCT bound).

$$\pi_{UCT}(n) = \underset{a}{\operatorname{argmax}}\left(\hat{Q}(n,a) + c\sqrt{\frac{\log(N(n))}{N(n,a)}}\right)$$

Question

State:



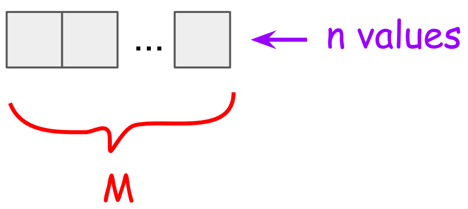


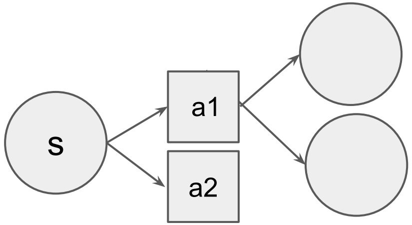
d) Consider a search tree where the reward is zero everywhere except at the leaves. When a MCTS trial goes through a node, we say that an action at the node wins if the trial ends in a leaf with reward 1. Consider an MCTS simulation where a node has been visited 16 times and has two actions, A and B. Action A has a won 2 out 4 times whereas action B has won 8 out of 12 times. Which action will the MCTS algorithm chose given the exploration parameter c is set to 1? Give the values of π_{UCT} for the node (consider log base 2 in UCT bound).

$$\pi_{UCT}(n) = \underset{a}{\operatorname{argmax}}\left(\hat{Q}(n,a) + c\sqrt{\frac{\log(N(n))}{N(n,a)}}\right)$$

$$\mathrm{A}: \quad \boxed{\frac{2}{4}} + \sqrt{\frac{\log \boxed{16}}{\boxed{4}}} = 1.5$$

State:





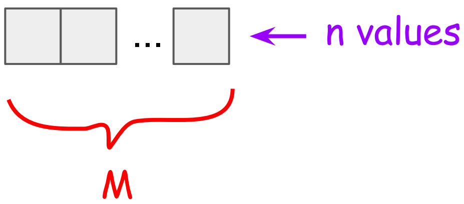
d) Consider a search tree where the reward is zero everywhere except at the leaves. When a MCTS trial goes through a node, we say that an action at the node wins if the trial ends in a leaf with reward 1. Consider an MCTS simulation where a node has been visited 16 times and has two actions, A and B. Action A has a won 2 out 4 times whereas action B has won 8 out of 12 times. Which action will the MCTS algorithm chose given the exploration parameter c is set to 1? Give the values of π_{UCT} for the node (consider log base 2 in UCT bound).

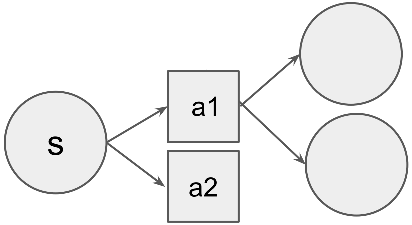
$$\pi_{UCT}(n) = \underset{a}{\operatorname{argmax}}\left(\hat{Q}(n,a) + c\sqrt{\frac{\log(N(n))}{N(n,a)}}\right)$$

$$\mathrm{A}: \quad \frac{2}{4} + \sqrt{\frac{\log 16}{4}} = 1.5$$

$$\mathrm{B}: \quad \boxed{\frac{8}{12}} + \sqrt{\frac{\log \boxed{16}}{\boxed{12}}} = 1.244$$

State:





d) Consider a search tree where the reward is zero everywhere except at the leaves. When a MCTS trial goes through a node, we say that an action at the node wins if the trial ends in a leaf with reward 1. Consider an MCTS simulation where a node has been visited 16 times and has two actions, A and B. Action A has a won 2 out 4 times whereas action B has won 8 out of 12 times. Which action will the MCTS algorithm chose given the exploration parameter c is set to 1? Give the values of π_{UCT} for the node (consider log base 2 in UCT bound).

$$\pi_{UCT}(n) = \underset{a}{\operatorname{argmax}}\left(\hat{Q}(n,a) + c\sqrt{\tfrac{\log(N(n))}{N(n,a)}}\right)$$

$$\mathrm{A}: \quad \frac{2}{4} + \sqrt{\frac{\log 16}{4}} = 1.5$$

$$\mathrm{B}: \quad \frac{8}{12} + \sqrt{\frac{\log 16}{12}} = 1.244$$

Question?

<EOF>