CS 188
Fall 2023

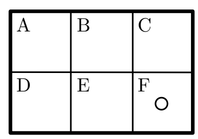
Midterm Review MDPs

Q1. MDP

Pacman is using MDPs to maximize his expected utility. In each environment:

- Pacman has the standard actions {North, East, South, West} unless blocked by an outer wall
- There is a reward of 1 point when eating the dot (for example, in the grid below, $R(C, South, F) = 1$)
- The game ends when the dot is eaten

(a) Consider a the following grid where there is a single food pellet in the bottom right corner (F). The **discount** factor is 0.5. There is no living reward. The states are simply the grid locations.



(i) What is the optimal policy for each state?

State	$\pi(state)$
A	
B	
C	
D	
E	

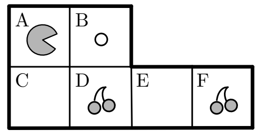
(ii) What is the optimal value for the state of being in the upper left corner (A)? Reminder: the discount factor is 0.5.

$V^*(A) =$

(iii) Using value iteration with the value of all states equal to zero at k=0, for which iteration k will $V_k(A) = V^*(A)$?

$k =$

(b) Consider a new Pacman level that begins with cherries in locations D and F. Landing on a grid position with cherries is worth 5 points and then the cherries at that position disappear. There is still one dot, worth 1 point. The game still only ends when the dot is eaten.



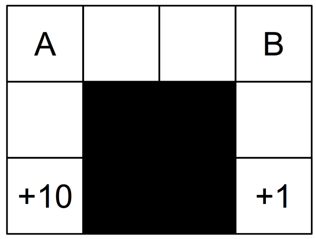
(i) With no discount ($\gamma = 1$) and a living reward of -1, what is the optimal policy for the states in this level's state space?

(ii) With no discount ($\gamma = 1$), what is the range of living reward values such that Pacman eats exactly one cherry when starting at position A?

Q2. MDPs: Value Iteration

An agent lives in gridworld G consisting of grid cells $s \in S$, and is not allowed to move into the cells colored black. In this gridworld, the agent can take actions to move to neighboring squares, when it is not on a numbered square. When the agent is on a numbered square, it is forced to exit to a terminal state (where it remains), collecting a reward equal to the number written on the square in the process.

Gridworld G



You decide to run value iteration for gridworld G. The value function at iteration k is $V_k(s)$. The initial value for all grid cells is 0 (that is, $V_0(s) = 0$ for all $s \in S$). When answering questions about iteration k for $V_k(s)$, either answer with a finite integer or ∞. For all questions, the discount factor is $\gamma = 1$.

(a) Consider running value iteration in gridworld G. Assume all legal movement actions **will always succeed** (and so the state transition function is deterministic).

(i) What is the smallest iteration k for which $V_k(A) > 0$? For this smallest iteration k, what is the value $V_k(A)$?

$k =$ ______________________ $V_k(A) =$ ______________________

(ii) What is the smallest iteration k for which $V_k(B) > 0$? For this smallest iteration k, what is the value $V_k(B)$?

$k =$ ______________________ $V_k(B) =$ ______________________

(iii) What is the smallest iteration k for which $V_k(A) = V^*(A)$? What is the value of $V^*(A)$?

$k =$ ______________________ $V^*(A) =$ ______________________

(iv) What is the smallest iteration k for which $V_k(B) = V^*(B)$? What is the value of $V^*(B)$?

$k =$ ______________________ $V^*(B) =$ ______________________

(b) Now assume all legal movement actions **succeed with probability** 0.8; with probability 0.2, the action fails and the agent remains in the same state.
Consider running value iteration in gridworld G. What is the smallest iteration k for which $V_k(A) = V^*(A)$? What is the value of $V^*(A)$?

$k =$ ______________________

$V^*(A) =$ ______________________