# PRACTICE QUESTIONS

## Multiple Choice Questions

### Introduction to CI and Responsible CI

1. (1 point) What is the main point of the Chinese room thought experiment, as proposed by the philosopher John Searle in 1980?
   A. A machine's "intelligence" is proportional to the number of operations that it can do autonomously.
   **B. Machines might be able to demonstrate "intelligent behavior" while not being conscious or having understanding in a human sense.**
   C. Machines that demonstrate "intelligent behavior" must posses some sort of consciousness, even if that consciousness is different from what humans have.
   D. In order for a machine to be considered "intelligent", it needs to be able to translate messages between two native speakers of different languages without being identified as a technical artifact.

2. (1 point) What statement best captures the relation of transfer and fitness function?
   A. After specifying the fitness function, the transfer function is no longer needed.
   **B. The fitness function specifies the objective for the behavior modeled by the transfer function.**
   C. The transfer function specifies how to behave by investigating the fitness function.
   D. They are two words for the same thing.

3. (1 point) The main difference between regression and classification is:
   **A. Targets of a classification problem are discrete, while regression targets are continuous.**
   B. Regression cannot be done with neural networks.
   C. Regression is unsupervised while classification is supervised learning.
   D. Regression is supervised while classification is unsupervised learning.

4. (1 point) A conundrum known as the frame problem within artificial intelligence concerns the application of knowledge about the past to draw inferences about the future. More specifically, it deals with defining what is relevant for a given system. Why is the frame problem so difficult to solve?
   **A. We don't know everything contextually required to solve a particular problem, so we cannot convey all the relevant data to an AI agent.**
   B. We need tremendous amounts of computational capacity to compute it.
   C. AI agent always make generalizations that may end up harming people even when we do not expect it.
   D. Because there is always the possibility the AI agents hacks his way into optimizing its utility function rather than solving the problem.

**Answer:** The option *"We don't know everything contextually required to solve a particular problem, so we cannot convey all the relevant data to an AI agent."* is correct. While contextually it may be true that AI agents make generalizations and it may also be true that we can perhaps (though unlikely) solve the problem with tremendous amounts of computational prowess. The real problem is also that we don't know what we require to solve a particular problem. Option *"Because there is always the possibility the AI agents hacks his way into optimizing its utility function rather than solving the problem."* is technically true, but is incomplete (AI agents can get it right by accident) and it is a result from the correct choice.

## Artificial Neural Networks

5. (1 point) Which of the following statements about convolutional neural networks (CNNs) is **FALSE**:
   A. Filters at the first layers detect more low-level features than filters at last layers.
   B. Multiple convolution filters are applied in parallel in every CNN layer.
   **C. Convolution filter can only be applied on 2D feature maps.**
   D. CNNs are more sparsely connected than MLPs.

6. (1 point) What type of activation function is best to use for the output layer of a multilayer perceptron if you want to train the model on predicting a person's weight given her height?
   **A. None.**
   B. ReLU.
   C. Tanh.
   D. Softmax.

7. (1 point) Which of the following statements about the standard backpropagation algorithm is **FALSE**:
   A. Backpropagation computes the gradient of the loss (objective) function with respect to the weights of the neural network.
   B. Backpropagation reduces the number of operations by reusing the results of previous gradient computations.
   C. The gradient of the loss (objective) function indicates the direction of steepest descent.
   **D. Backpropagation always leads to the global optimum.**

## Genetic Algorithms

8. (1 point) Consider that you are designing a tablet computer for children. You want to optimize the dimensions ($width$, $height$, and $length$) of the tablet. The evaluation criterion is the $comfort$ with which the children can hold the tablet in their hands. If you want to solve this problem via the Genetic Algorithm, how would you encode the problem?
   A. The chromosome represents the sum of the three dimension of the tablet.
   **B. The chromosome is a concatenation of 3 elements, representing each dimension of the tablet.**
   C. The chromosome has one element, representing the evaluation criterion.
   D. The chromosome is a concatenation of 4 elements, representing each dimension of the tablet and the evaluation criterion.

9. (1 point) The *performance graph* of a Genetic Algorithm plots ...
   A. the computational time and memory used by the GA over generations.
   B. the convergence time of the GA over different sizes of initial population.
   C. the average fitness of the last generation (of a fixed number of generations) over different chromosome sizes.
   **D. the average fitness of the population and the fitness of the fittest individual per population over generations.**

10. (1 point) Consider that a genetic algorithm uses chromosomes of the form $X = abcd$ with a fixed length of four genes. Each gene can be any integer value between 0 (inclusive) and 9 (inclusive).

    Let the fitness of an individual be calculated as: $F(X) = 18 + (a - b + c - d)$.

    Which of the following four chromosomes has the highest likelihood of getting selected for modification according to roulette wheel selection strategy?
    A. $X_1 = 5412$
    B. $X_2 = 1987$
    **C.** $X_3 = 3693$
    D. $X_4 = 0450$

## Swarm Intelligence

11. (1 point) A key difference between Particle Swarm Optimization (PSO) and Genetic Algorithms (GA) is:
    A. A GA uses stochastic search, PSO does not.
    B. A GA uses a population of individuals, PSO does not.
    C. A GA uses a fitness function, PSO does not.
    **D. A GA has generations of individuals, PSO does not.**

12. (1 point) In Particle Swarm Optimization (PSO), each particle i moves through the solution space depending on its velocity v, which for each solution space dimension j, is expressed as:

$$v_{ij}(t+1) = v_{ij}(t) + c_1 r_{1j}(t)[y_{ij}(t) - x_{ij}(t)] + c_2 r_{2j}(t)[\hat{y}_{ij}(t) - x_{ij}(t)]$$

    If you want the whole swarm to be strongly biased towards the collectively found best solution, then you need to set:
    **A. $c_2 \gg c_1$.**
    B. $c_2 \ll c_1$.
    C. $\hat{y}_{ij}(t) \gg y_{ij}(t)$.
    D. $\hat{y}_{ij}(t) \ll y_{ij}(t)$.

13. (1 point) What is a valid reason for the evaporation of pheromone in an Ant Colony Optimization algorithm?
    A. The evaporation of pheromone indicates a clear route so that, ant after ant, shorter and shorter routes can be found.
    B. The evaporation of pheromone avoids that a single ant can reach the shortest route before the other ants.
    C. The evaporation of pheromone avoids that ants visit the same node more than once.
    **D. The evaporation of pheromone can be seen as an exploration mechanism that avoids quick convergence of all the ants toward a suboptimal path.**

## Reinforcement Learning

14. (1 point) Reinforcement Learning (RL) is *NOT* needed to (pick one):
    A. solve a repeated action selection problem with delayed rewards.
    B. optimize expected cumulative return over a series of state-action-state transitions.
    **C. solve *supervised classification* problems.**
    D. solve temporal credit assignment problems.

15. (1 point) In Reinforcement Learning (RL), which reward formulation might be used:
    A. $R(s')$
    B. $R(s, a, s')$
    C. $\Pr(r|s, a, s')$
    **D. All of the above.**

# Open-ended Questions

## Artificial Neural Networks

16. A multilayer perceptron (MLP) is designed to perform binary classification (i.e. two classes) given a 3-dimensional input. The network has one hidden layer with 4 hidden neurons.
    (a) (1 point) Write down the equations in matrix form for forward-passing one sample. Please, define any new notation you introduce that is different from the one we used in class.

**Answer:**

$$z^{[1]} = W^{[1]}x + b^{[1]}$$
$$a^{[1]} = \sigma(z^{[1]})$$
$$z^{[2]} = W^{[2]}a^{[1]} + b^{[2]}$$
$$\hat{y} = \sigma(z^{[2]})$$

Note: The equations should be written in matrix form.
Grading: 0.75 pts. if equations are correct but there is no bias term + 0.25 pts. for including the bias term.

(b) (1 point) What type of activation function did you use? Why?

**Answer:** The first activation can be any except the identity. The second one should be sigmoid.
*Note: If Sigmoid was chosen there should be only one output neuron and the dimensions of $\hat{y}$ in the next question should be $1 \times 1$. If there are two output neurons ($\hat{y} \to 2 \times 1$) then softmax is needed so that the output is normalized.*
*Grading: 0.5 pt for choosing sigmoid/softmax + 0.5 for writing "(binary) classification" as the reason.*

(c) (1 point) Specify the size of all vectors and matrices you indicated in part **a**.

**Answer:**
Option A: 1 output neuron

$$\begin{aligned} x &\to \quad 3 \times 1 \quad W^{[1]} \to \quad 4 \times 3 \\ b^{[1]} &\to \quad 4 \times 1 \quad z^{[1]} \to \quad 4 \times 1 \\ a^{[1]} &\to \quad 4 \times 1 \quad W^{[2]} \to \quad 1 \times 4 \\ b^{[2]} &\to \quad 1 \times 1 \quad z^{[2]} \to \quad 1 \times 1 \\ & \qquad\qquad \hat{y} \to \quad 1 \times 1 \end{aligned}$$

Option B: 2 output neurons

$$\begin{aligned} x &\to \quad 3 \times 1 \quad W^{[1]} \to \quad 4 \times 3 \\ b^{[1]} &\to \quad 4 \times 1 \quad z^{[1]} \to \quad 4 \times 1 \\ a^{[1]} &\to \quad 4 \times 1 \quad W^{[2]} \to \quad 2 \times 4 \\ b^{[2]} &\to \quad 2 \times 1 \quad z^{[2]} \to \quad 2 \times 1 \\ & \qquad\qquad \hat{y} \to \quad 2 \times 1 \end{aligned}$$

Grading: 1 pt. if either option A or option B is given.

(d) (1 point) How would you forward-pass $10$ samples at the same time? What would be the new dimensions of all vectors and matrices?

**Answer:**
Stack the $10$ samples as columns to form a matrix.

Option A: 1 output neuron. All of them are the same dimensions except:

$$\begin{aligned} X \to &\quad 3 \times 10 \quad Z^{[1]} \to \quad 4 \times 10 \\ A^{[1]} \to &\quad 4 \times 10 \quad Z^{[2]} \to \quad 1 \times 10 \\ &\quad Y \to \quad 1 \times 10 \end{aligned}$$

Option B: 2 output neurons. All of them are the same dimensions except:

$$\begin{aligned} X \to &\quad 3 \times 10 \quad Z^{[1]} \to \quad 4 \times 10 \\ A^{[1]} \to &\quad 4 \times 10 \quad Z^{[2]} \to \quad 2 \times 10 \\ &\quad Y \to \quad 2 \times 10 \end{aligned}$$

*Note 1: Answers that expand the bias terms and write rows $\times 10$ are also correct.*
*Note 2: The answer here should be consistent with part C. That is, if option A was chosen in part C option A should be chosen here.*
*Grading: 1 pt. if either option A or option B is given.*

## Genetic Algorithms

17. (4 points) Assume that we are employing the Non-dominated Sorting Genetic Algorithm, NSGA-II, for finding vendors of COVID-19 test kits. We are interested in finding vendors who supply test kits having low *cost* and high *reliability*.

    At an intermediate stage in the search process, we have eight candidate vendors ($A$–$H$) whose test kits have the cost and reliability as shown in Table 1. All the values have been normalized from 0 to 10.

    Now, if we want the NSGA-II algorithm to select only **four** candidates to pass onto the next step, which four candidates would the algorithm choose? For each candidate you identify, provide a justification as to why NSGA-II would choose that candidate.

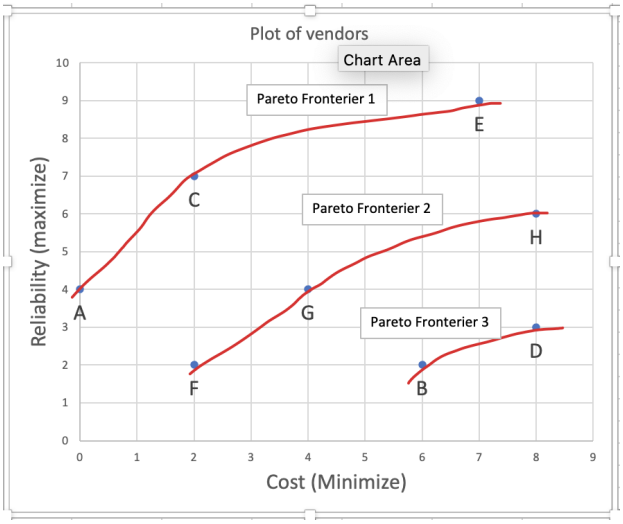
Table 1: Candidate COVID-19 test kit vendors

| Vendor | Cost | Reliability |
|---|---|---|
| $A$ | 0 | 4 |
| $B$ | 6 | 2 |
| $C$ | 2 | 7 |
| $D$ | 8 | 3 |
| $E$ | 7 | 9 |
| $F$ | 2 | 2 |
| $G$ | 4 | 4 |
| $H$ | 8 | 6 |

**Answer:**

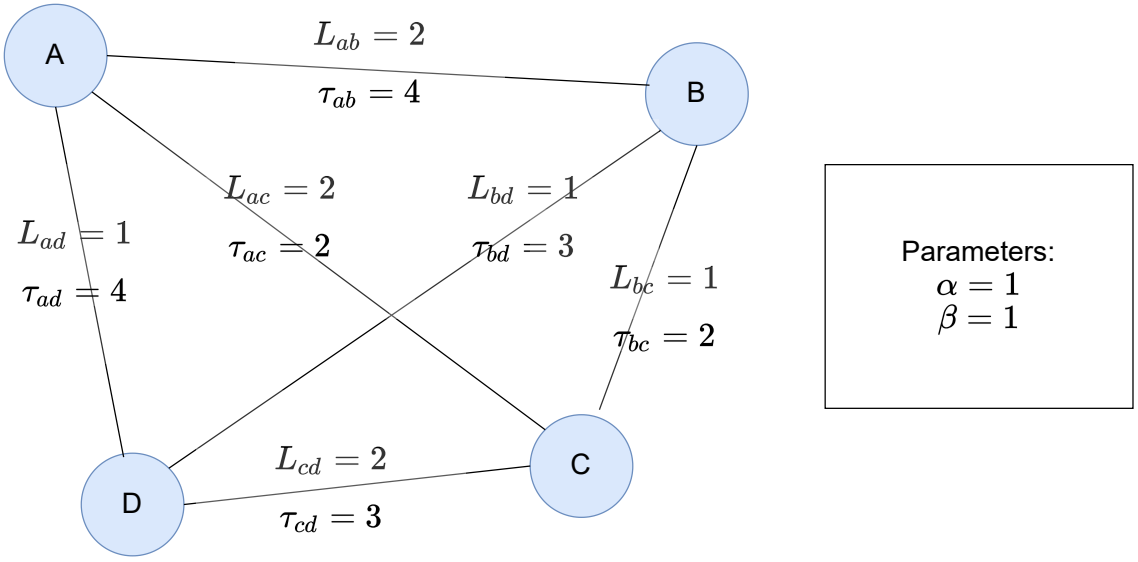
- $A$, $C$, $E$: Each of these points are on the best Pareto front.
- $F$ or $H$ (mentioning one of these is sufficient): These points are on the next best Pareto front but have the highest crowding distance in their font.

**Rubric**: 0.5 for identifying each correct candidate, and 0.5 for providing the correct justification for each identified point. Not necessary to include a picture like the one below.




## Swarm Intelligence

18. Consider the Traveling Salesman Problem presented in the figure below. The length of the links ($L$) and the amount of pheromone ($\tau$) deposited in each link after a given number of iterations are indicated in the figure. Answer the following questions using Ant Colony Optimization (ACO, Ant-Cycle algorithm), considering the parameters indicated in the box.




(a) (2 points) Suppose a given ant starts at node A. What is the probability that this ant will move from node A to node B?

**Answer:**
Grading criteria:

- 1 point: Calculate the term for each link (ab, ac, ad). Each link missing or wrong discount 0.33.
- 1 point: Final answer

$p_{ab} = \frac{\tau_{ab}^{\alpha} \cdot (1/L_{ab})^{\beta}}{\sum_{m \in C_i^k} \tau_{am}^{\alpha} \cdot (1/L_{am})^{\beta}}$
For each of the unvisited cities:
B: $\tau_{ab}^{\alpha} \cdot (1/L_{ab})^{\beta} \rightarrow 4^1 \cdot (1/2)^1 \rightarrow 2$
C: $\tau_{ac}^{\alpha} \cdot (1/L_{ac})^{\beta} \rightarrow 2^1 \cdot (1/2)^1 \rightarrow 1$
D: $\tau_{ad}^{\alpha} \cdot (1/L_{ad})^{\beta} \rightarrow 4^1 \cdot (1/1)^1 \rightarrow 4$
Then we can calculate that:
$p_{ab} = \frac{2}{2+1+4} = 0.2857$
The probability that the ant will move from node A to node B is 28.57%.

(b) (2 points) Consider that the same ant that started on node A has now moved to node B. What is the probability that the next move of this ant will be from node B to node C?

**Answer:**
Grading criteria:

- 1 point: Calculate the term for each link (bc, bd). Each link missing or wrong discount 0.5. If calculated the visited link as well (ba), discount 0.5 (to a minimum of 0 point).
- 1 point: Final answer.

$p_{bc} = \frac{\tau_{bc}^{\alpha} \cdot (1/L_{bc})^{\beta}}{\sum_{m \in C_i^k} \tau_{bm}^{\alpha} \cdot (1/L_{bm})^{\beta}}$
For each of the unvisited cities:
C: $\tau_{bc}^{\alpha} \cdot (1/L_{bc})^{\beta} \rightarrow 2^1 \cdot (1/1)^1 \rightarrow 2$
D: $\tau_{bd}^{\alpha} \cdot (1/L_{bd})^{\beta} \rightarrow 3^1 \cdot (1/1)^1 \rightarrow 3$
Then we can calculate that:
$p_{ab} = \frac{2}{2+3} = 0.4$
The probability that the ant will move from node B to node C, assuming that it came from node A (previous question), is 40%.

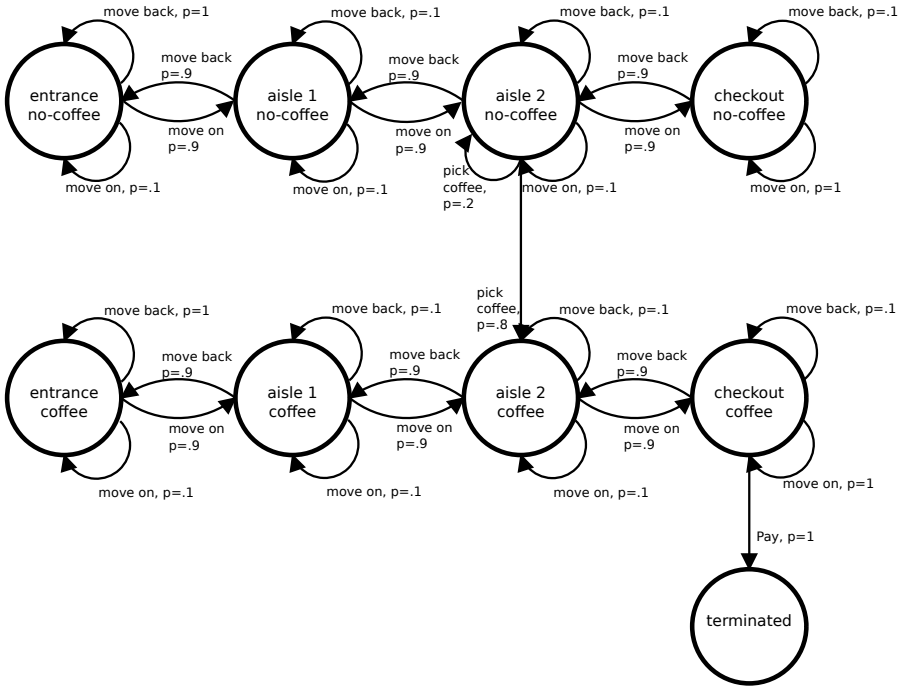
## Reinforcement Learning

19. You ran out of coffee, and are busy, so you send your robot to the store on your behalf. The store consists of 4 locations: entrance, aisle 1, aisle 2, and checkout that can be navigated in the stated order by the action "move on" and in the reverse order by action "move back". There is a small probability (10%) of not moving. The coffee is located in aisle 2, and can be grabbed with the "pick coffee action" (80% success rate, if not successful the state does not change). When the robot has the coffee, it should of course "pay" at the checkout (100% success rate), after which the problem terminates and the robot is rewarded with $R(s = *, a = *, s' = terminated) = +10$. Otherwise, the reward is $-1$. There are no other action effects: e.g., performing the "pick coffee" in aisle 1 means that the state will not change. The problem begins after robot enters the store at the entrance, without coffee.

(a) (2 points) Draw a description of the transitions of the MDP: nodes represent states, while arrows annotated with actions and probabilities should represent the transitions. Actions that lead deterministically to no change in state can be omitted to avoid clutter. Put names in the states that you can use below.

**Answer:**
Grading criteria:

- 2 points can be earned
- there need to be 8 (+ 1 terminal state) states
- the effect of all actions needs to be indicated clearly
- the terminated state is optional




(b) (3 points) Iterative policy evaluation, applies the Bellman equation in iterations indexed by $k$. For a deterministic policy $\pi$, that takes action $\pi(s) = a$ in state $s$, the update rule is given by:

$$v_{k+1}(s) := \sum_{s'} P(s'|s, \pi(s)) \left[R(s, \pi(s), s') + \gamma v_k(s')\right]$$

In this question, you need to use this formula to determine the $k = 4$-steps-to-go value function, $v_4(s)$, for a policy that we will now describe:

- When coffee is picked up: move towards the checkout and then pay. When not picked up: perform the "pickup" action.
- First write down the explicit (tabular) representation of the policy
- Use a discount factor of 0.9.

**Answer:**

Grading criteria:

- 1.0 Evaluation of the correct policy.
- 1.0 Student discusses or shows how the Bellman update equation is used. (show intermediate steps, or motivate answer by referring to equation)
- 1.0 Correct final answer.

We abbreviate 'c' for 'coffee' and 'n' for 'no coffee'. We start with writing down the policy:

$\pi(\langle entrance, n\rangle) = pickup$
$\pi(\langle aisle1, n\rangle) = pickup$
$\pi(\langle aisle2, n\rangle) = pickup$
$\pi(\langle checkout, n\rangle) = pickup$
$\pi(\langle entrance, c\rangle) = move\,on$
$\pi(\langle aisle1, c\rangle) = move\,on$
$\pi(\langle aisle2, c\rangle) = move\,on$
$\pi(\langle checkout, c\rangle) = pay$
$\pi(terminated) = *$s

Now we start IPE:

Iteration 0 (initialization):

$v_0(\langle entrance, n\rangle) = 0$
$v_0(\langle aisle1, n\rangle) = 0$
$v_0(\langle aisle2, n\rangle) = 0$
$v_0(\langle checkout, n\rangle) = 0$
$v_0(\langle entrance, c\rangle) = 0$
$v_0(\langle aisle1, c\rangle) = 0$
$v_0(\langle aisle2, c\rangle) = 0$
$v_0(\langle checkout, c\rangle) = 0$
$v_o(terminated) = 0$

Iteration 1:

$$\begin{aligned} v_1(\langle entrance, n\rangle) =& \Pr(\langle entrance, n\rangle \mid \langle entrance, n\rangle, pickup)\,[R(\langle entrance, n\rangle, pickup, \langle entrance, n\rangle) + \gamma v_0(\langle entrance, n\rangle)] \\ =& 1\,[-1 + 0.9 \cdot 0] = -1 \end{aligned}$$

$$\begin{aligned} v_1(\langle aisle1, n\rangle) =& \Pr(\langle aisle1, n\rangle \mid \langle aisle1, n\rangle, pickup)\,[R(\langle aisle1, n\rangle, pickup, \langle aisle1, n\rangle) + \gamma v_0(\langle aisle1, n\rangle)] \\ =& 1\,[-1 + 0.9 \cdot 0] = -1 \end{aligned}$$

$$\begin{aligned} v_1(\langle aisle2, n\rangle) =& \Pr(\langle aisle2, c\rangle \mid \langle aisle2, n\rangle, pickup)\,[R(\langle aisle2, n\rangle, pickup, \langle aisle2, c\rangle) + \gamma v_0(\langle aisle2, c\rangle)] \\ &+ \Pr(\langle aisle2, n\rangle \mid \langle aisle2, n\rangle, pickup)\,[R(\langle aisle2, n\rangle, pickup, \langle aisle2, n\rangle) + \gamma v_0(\langle aisle2, n\rangle)] \\ =& .8\,[-1 + 0.9 \cdot 0] + .2\,[-1 + 0.9 \cdot 0] = -1 \end{aligned}$$

$$\begin{aligned} v_1(\langle checkout, n\rangle) =& \Pr(\langle checkout, n\rangle \mid \langle checkout, n\rangle, pickup)\,[R(\langle checkout, n\rangle, pickup, \langle checkout, n\rangle) + \gamma v_0(\langle checkout, n\rangle)] \\ =& 1\,[-1 + 0.9 \cdot 0] = -1 \end{aligned}$$

$$\begin{aligned} v_1(\langle entrance, c\rangle) =& \Pr(\langle aisle1, c\rangle \mid \langle entrance, c\rangle, move\,on)\,[R(\langle entrance, c\rangle, move\,on, \langle aisle1, c\rangle) + \gamma v_0(\langle aisle1, c\rangle)] \\ &+ \Pr(\langle entrance1, c\rangle \mid \langle entrance, c\rangle, move\,on)\,[R(\langle entrance, c\rangle, move\,on, \langle entrance1, c\rangle) + \gamma v_0(\langle entrance1, c\rangle)] \\ =& .9\,[-1 + 0.9 \cdot 0] + .1\,[-1 + 0.9 \cdot 0] = -1 \end{aligned}$$

$$\begin{aligned} v_1(\langle aisle1, c\rangle) =& \Pr(\langle aisle2, c\rangle \mid \langle aisle1, c\rangle, move\,on)\,[R(\langle aisle1, c\rangle, move\,on, \langle aisle2, c\rangle) + \gamma v_0(\langle aisle2, c\rangle)] \\ &+ \Pr(\langle aisle1, c\rangle \mid \langle aisle1, c\rangle, move\,on)\,[R(\langle aisle1, c\rangle, move\,on, \langle aisle1, c\rangle) + \gamma v_0(\langle aisle1, c\rangle)] \\ =& .9\,[-1 + 0.9 \cdot 0] + .1\,[-1 + 0.9 \cdot 0] = -1 \end{aligned}$$

$$\begin{aligned} v_1(\langle aisle2, c\rangle) =& \Pr(\langle checkout, c\rangle \mid \langle aisle2, c\rangle, move\,on)\,[R(\langle aisle2, c\rangle, move\,on, \langle checkout, c\rangle) + \gamma v_0(\langle checkout, c\rangle)] \\ &+ \Pr(\langle aisle2, c\rangle \mid \langle aisle2, c\rangle, move\,on)\,[R(\langle aisle2, c\rangle, move\,on, \langle aisle2, c\rangle) + \gamma v_0(\langle aisle2, c\rangle)] \\ =& .9\,[-1 + 0.9 \cdot 0] + .1\,[-1 + 0.9 \cdot 0] = -1 \end{aligned}$$

$$v_1(\langle checkout, c\rangle) = \Pr(terminated|\ \langle checkout, n\rangle, pay)\ [R(\langle checkout, c\rangle, pickup, terminated) + \gamma v_0(terminated)]$$
$$= 1\ [10 + 0.9 \cdot 0] = 10$$

$v_1(terminated) = 0$

Iteration $k = 2$:

$$v_k(\langle entrance, n\rangle) = \Pr(\langle entrance, n\rangle \mid \langle entrance, n\rangle, pickup)\ [R(\langle entrance, n\rangle, pickup, \langle entrance, n\rangle) + \gamma v_{k-1}(\langle entrance, n\rangle)]$$
$$= 1\ [-1 + 0.9 \cdot -1] = -1.9$$

$$v_k(\langle aisle1, n\rangle) = \Pr(\langle aisle1, n\rangle \mid \langle aisle1, n\rangle, pickup)\ [R(\langle aisle1, n\rangle, pickup, \langle aisle1, n\rangle) + \gamma v_{k-1}(\langle aisle1, n\rangle)]$$
$$= 1\ [-1 + 0.9 \cdot -1] = -1.9$$

$$v_k(\langle aisle2, n\rangle) = \Pr(\langle aisle2, c\rangle \mid \langle aisle2, n\rangle, pickup)\ [R(\langle aisle2, n\rangle, pickup, \langle aisle2, c\rangle) + \gamma v_{k-1}(\langle aisle2, c\rangle)]$$
$$+ \Pr(\langle aisle2, n\rangle \mid \langle aisle2, n\rangle, pickup)\ [R(\langle aisle2, n\rangle, pickup, \langle aisle2, n\rangle) + \gamma v_{k-1}(\langle aisle2, n\rangle)]$$
$$= .8\ [-1 + 0.9 \cdot -1] + .2\ [-1 + 0.9 \cdot -1] = -1.9$$

$$v_k(\langle checkout, n\rangle) = \Pr(\langle checkout, n\rangle \mid \langle checkout, n\rangle, pickup)\ [R(\langle checkout, n\rangle, pickup, \langle checkout, n\rangle) + \gamma v_{k-1}(\langle checkout, n\rangle)]$$
$$= 1\ [-1 + 0.9 \cdot -1] = -1.9$$

$$v_k(\langle entrance, c\rangle) = \Pr(\langle aisle1, c\rangle \mid \langle entrance, c\rangle, move\ on)\ [R(\langle entrance, c\rangle, move\ on, \langle aisle1, c\rangle) + \gamma v_{k-1}(\langle aisle1, c\rangle)]$$
$$+ \Pr(\langle entrance1, c\rangle \mid \langle entrance, c\rangle, move\ on)\ [R(\langle entrance, c\rangle, move\ on, \langle entrance1, c\rangle) + \gamma v_{k-1}(\langle entrance1, c\rangle)]$$
$$= .9\ [-1 + 0.9 \cdot -1] + .1\ [-1 + 0.9 \cdot -1] = -1.9$$

$$v_k(\langle aisle1, c\rangle) = \Pr(\langle aisle2, c\rangle \mid \langle aisle1, c\rangle, move\ on)\ [R(\langle aisle1, c\rangle, move\ on, \langle aisle2, c\rangle) + \gamma v_{k-1}(\langle aisle2, c\rangle)]$$
$$+ \Pr(\langle aisle1, c\rangle \mid \langle aisle1, c\rangle, move\ on)\ [R(\langle aisle1, c\rangle, move\ on, \langle aisle1, c\rangle) + \gamma v_{k-1}(\langle aisle1, c\rangle)]$$
$$= .9\ [-1 + 0.9 \cdot -1] + .1\ [-1 + 0.9 \cdot -1] = -1.9$$

$$v_k(\langle aisle2, c\rangle) = \Pr(\langle checkout, c\rangle \mid \langle aisle2, c\rangle, move\ on)\ [R(\langle aisle2, c\rangle, move\ on, \langle checkout, c\rangle) + \gamma v_{k-1}(\langle checkout, c\rangle)]$$
$$+ \Pr(\langle aisle2, c\rangle \mid \langle aisle2, c\rangle, move\ on)\ [R(\langle aisle2, c\rangle, move\ on, \langle aisle2, c\rangle) + \gamma v_{k-1}(\langle aisle2, c\rangle)]$$
$$= .9\ [-1 + 0.9 \cdot 10] + .1\ [-1 + 0.9 \cdot -1] = -.9 \cdot 8 + .1 \cdot -1.9 = 7.2 - 0.19 = 7.01$$

$$v_k(\langle checkout, c\rangle) = \Pr(terminated|\ \langle checkout, n\rangle, pay)\ [R(\langle checkout, c\rangle, pickup, terminated) + \gamma v_{k-1}(terminated)]$$
$$= 1\ [10 + 0.9 \cdot 0] = 10$$

$v_k(terminated) = 0$

Iteration $k = 3$:

$$v_k(\langle entrance, n\rangle) = \Pr(\langle entrance, n\rangle \mid \langle entrance, n\rangle, pickup)\ [R(\langle entrance, n\rangle, pickup, \langle entrance, n\rangle) + \gamma v_{k-1}(\langle entrance, n\rangle)]$$
$$= 1\ [-1 + 0.9 \cdot -1.9] = -2.71$$

$$v_k(\langle aisle1, n\rangle) = \Pr(\langle aisle1, n\rangle \mid \langle aisle1, n\rangle, pickup)\ [R(\langle aisle1, n\rangle, pickup, \langle aisle1, n\rangle) + \gamma v_{k-1}(\langle aisle1, n\rangle)]$$
$$= 1\ [-1 + 0.9 \cdot -1.9] = -2.71$$

$$v_k(\langle aisle2, n\rangle) = \Pr(\langle aisle2, c\rangle \mid \langle aisle2, n\rangle, pickup)\ [R(\langle aisle2, n\rangle, pickup, \langle aisle2, c\rangle) + \gamma v_{k-1}(\langle aisle2, c\rangle)]$$
$$+ \Pr(\langle aisle2, n\rangle \mid \langle aisle2, n\rangle, pickup)\ [R(\langle aisle2, n\rangle, pickup, \langle aisle2, n\rangle) + \gamma v_{k-1}(\langle aisle2, n\rangle)]$$
$$= .8\ [-1 + 0.9 \cdot 7.01] + .2\ [-1 + 0.9 \cdot -1.9] = 3.7052$$

$$v_k(\langle checkout, n\rangle) = \Pr(\langle checkout, n\rangle \mid \langle checkout, n\rangle, pickup)\ [R(\langle checkout, n\rangle, pickup, \langle checkout, n\rangle) + \gamma v_{k-1}(\langle checkout, n\rangle)]$$
$$= 1\ [-1 + 0.9 \cdot -1.9] = -2.71$$

$$v_k(\langle entrance, c\rangle) = \Pr(\langle aisle1, c\rangle \mid \langle entrance, c\rangle, move\ on)\ [R(\langle entrance, c\rangle, move\ on, \langle aisle1, c\rangle) + \gamma v_{k-1}(\langle aisle1, c\rangle)]$$
$$+ \Pr(\langle entrance1, c\rangle \mid \langle entrance, c\rangle, move\ on)\ [R(\langle entrance, c\rangle, move\ on, \langle entrance1, c\rangle) + \gamma v_{k-1}(\langle entrance1, c\rangle)]$$
$$= .9\ [-1 + 0.9 \cdot -1.9] + .1\ [-1 + 0.9 \cdot -1.9] = -2.71$$

$$v_k(\langle aisle1, c\rangle) = \Pr(\langle aisle2, c\rangle \mid \langle aisle1, c\rangle, move\ on)\ [R(\langle aisle1, c\rangle, move\ on, \langle aisle2, c\rangle) + \gamma v_{k-1}(\langle aisle2, c\rangle)]$$
$$+ \Pr(\langle aisle1, c\rangle \mid \langle aisle1, c\rangle, move\ on)\ [R(\langle aisle1, c\rangle, move\ on, \langle aisle1, c\rangle) + \gamma v_{k-1}(\langle aisle1, c\rangle)]$$
$$= .9\ [-1 + 0.9 \cdot 7.01] + .1\ [-1 + 0.9 \cdot -1.9] = 4.5071$$

$$
\begin{aligned}
v_k(\langle aisle2, c\rangle) =& \Pr(\langle checkout, c\rangle \mid \langle aisle2, c\rangle, move\ on)\,[R(\langle aisle2, c\rangle, move\ on, \langle checkout, c\rangle) + \gamma v_{k-1}(\langle checkout, c\rangle)] \\
&+ \Pr(\langle aisle2, c\rangle \mid \langle aisle2, c\rangle, move\ on)\,[R(\langle aisle2, c\rangle, move\ on, \langle aisle2, c\rangle) + \gamma v_{k-1}(\langle aisle2, c\rangle)] \\
=& .9\,[-1 + 0.9 \cdot 10] + .1\,[-1 + 0.9 \cdot 7.01] = 7.7309
\end{aligned}
$$

$$
\begin{aligned}
v_k(\langle checkout, c\rangle) =& \Pr(terminated \mid \langle checkout, n\rangle, pay)\,[R(\langle checkout, c\rangle, pickup, terminated) + \gamma v_{k-1}(terminated)] \\
=& 1\,[10 + 0.9 \cdot 0] = 10
\end{aligned}
$$

$v_k(terminated) = 0$

Iteration $k = 4$:

$$
\begin{aligned}
v_k(\langle entrance, n\rangle) =& \Pr(\langle entrance, n\rangle \mid \langle entrance, n\rangle, pickup)\,[R(\langle entrance, n\rangle, pickup, \langle entrance, n\rangle) + \gamma v_{k-1}(\langle entrance, n\rangle)] \\
=& 1\,[-1 + 0.9 \cdot -2.71] = -3.439
\end{aligned}
$$

$$
\begin{aligned}
v_k(\langle aisle1, n\rangle) =& \Pr(\langle aisle1, n\rangle \mid \langle aisle1, n\rangle, pickup)\,[R(\langle aisle1, n\rangle, pickup, \langle aisle1, n\rangle) + \gamma v_{k-1}(\langle aisle1, n\rangle)] \\
=& 1\,[-1 + 0.9 \cdot -2.71] = -3.439
\end{aligned}
$$

$$
\begin{aligned}
v_k(\langle aisle2, n\rangle) =& \Pr(\langle aisle2, c\rangle \mid \langle aisle2, n\rangle, pickup)\,[R(\langle aisle2, n\rangle, pickup, \langle aisle2, c\rangle) + \gamma v_{k-1}(\langle aisle2, c\rangle)] \\
&+ \Pr(\langle aisle2, n\rangle \mid \langle aisle2, n\rangle, pickup)\,[R(\langle aisle2, n\rangle, pickup, \langle aisle2, n\rangle) + \gamma v_{k-1}(\langle aisle2, n\rangle)] \\
=& .8\,[-1 + 0.9 \cdot 7.7309] + .2\,[-1 + 0.9 \cdot 3.7052] = 5.2332
\end{aligned}
$$

$$
\begin{aligned}
v_k(\langle checkout, n\rangle) =& \Pr(\langle checkout, n\rangle \mid \langle checkout, n\rangle, pickup)\,[R(\langle checkout, n\rangle, pickup, \langle checkout, n\rangle) + \gamma v_{k-1}(\langle checkout, n\rangle)] \\
=& 1\,[-1 + 0.9 \cdot -2.71] = -3.439
\end{aligned}
$$

$$
\begin{aligned}
v_k(\langle entrance, c\rangle) =& \Pr(\langle aisle1, c\rangle \mid \langle entrance, c\rangle, move\ on)\,[R(\langle entrance, c\rangle, move\ on, \langle aisle1, c\rangle) + \gamma v_{k-1}(\langle aisle1, c\rangle)] \\
&+ \Pr(\langle entrance1, c\rangle \mid \langle entrance, c\rangle, move\ on)\,[R(\langle entrance, c\rangle, move\ on, \langle entrance1, c\rangle) + \gamma v_{k-1}(\langle entrance1, c\rangle)] \\
=& .9\,[-1 + 0.9 \cdot 4.5071] + .1\,[-1 + 0.9 \cdot -2.71] = 2.4069
\end{aligned}
$$

$$
\begin{aligned}
v_k(\langle aisle1, c\rangle) =& \Pr(\langle aisle2, c\rangle \mid \langle aisle1, c\rangle, move\ on)\,[R(\langle aisle1, c\rangle, move\ on, \langle aisle2, c\rangle) + \gamma v_{k-1}(\langle aisle2, c\rangle)] \\
&+ \Pr(\langle aisle1, c\rangle \mid \langle aisle1, c\rangle, move\ on)\,[R(\langle aisle1, c\rangle, move\ on, \langle aisle1, c\rangle) + \gamma v_{k-1}(\langle aisle1, c\rangle)] \\
=& .9\,[-1 + 0.9 \cdot 7.7309] + .1\,[-1 + 0.9 \cdot 4.5071] = 4.8564
\end{aligned}
$$

$$
\begin{aligned}
v_k(\langle aisle2, c\rangle) =& \Pr(\langle checkout, c\rangle \mid \langle aisle2, c\rangle, move\ on)\,[R(\langle aisle2, c\rangle, move\ on, \langle checkout, c\rangle) + \gamma v_{k-1}(\langle checkout, c\rangle)] \\
&+ \Pr(\langle aisle2, c\rangle \mid \langle aisle2, c\rangle, move\ on)\,[R(\langle aisle2, c\rangle, move\ on, \langle aisle2, c\rangle) + \gamma v_{k-1}(\langle aisle2, c\rangle)] \\
=& .9\,[-1 + 0.9 \cdot 10] + .1\,[-1 + 0.9 \cdot 7.7309] = 7.7958
\end{aligned}
$$

$$
\begin{aligned}
v_k(\langle checkout, c\rangle) =& \Pr(terminated \mid \langle checkout, n\rangle, pay)\,[R(\langle checkout, c\rangle, pickup, terminated) + \gamma v_{k-1}(terminated)] \\
=& 1\,[10 + 0.9 \cdot 0] = 10
\end{aligned}
$$

$v_k(terminated) = 0$

(c) (1 point) What is $v_\pi(s_0)$ the value for the policy $\pi$ starting in the initial state $s_0$. Clearly explain your calculations.

**Answer:**

Grading criteria:

- 1.0 for correct computation.

The initial state is $s_0 = \langle entrance, n\rangle$. The policy will get -1 reward on each time step, leading to

$$v_\pi(s_0 = \langle entrance, n\rangle) = \sum_{t=0}^{\infty} \gamma^t \cdot (-1) = \frac{-1}{1-\gamma} = -10.$$

.

End of exam.
Please make sure you answerred all 19 questions.