SID: _____

Q1. [14 pts] Searching for a Password

You are trying to recover a password to an encrypted file, by using search. You know that the password is up to 10 letters long and contains only the letters A, B, C.

You formulate a search problem:

- The initial state is the empty string.
- The successor function is to append one letter (A, B, or C) to the string.
- The goal test is to verify a candidate password using the decryption software. There are 6 correct passwords: AAACCC, ABBCC, BABAB, BCABACB, CBAC, and CBACB.
- Assume that **all ties are broken alphabetically**. For example, if there is a tie between states "A", "B", and "C", expand "A" first, then "B", then "C".

(a) [3 pts] From the six correct passwords below, select the one that will be returned by **depth-first search**:

- ● AAACCC
- ○ ABBCC
- ○ BABAB
- ○ BCABACB
- ○ CBAC
- ○ CBACB

(b) [3 pts] From the six correct passwords below, select the one that will be returned by **breadth-first search**:

- ○ AAACCC
- ○ ABBCC
- ○ BABAB
- ○ BCABACB
- ● CBAC
- ○ CBACB

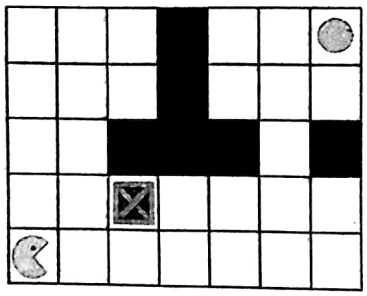
(c) [4 pts] You suspect that some letters are more likely to occur in the password than others. You model this by setting $cost(A) = 1$, $cost(B) = 2$, $cost(C) = 3$. From the six correct passwords below, select the one that will be returned by **uniform cost search** using these costs:

- ○ AAACCC
- ○ ABBCC
- ● BABAB
- ○ BCABACB
- ○ CBAC
- ○ CBACB

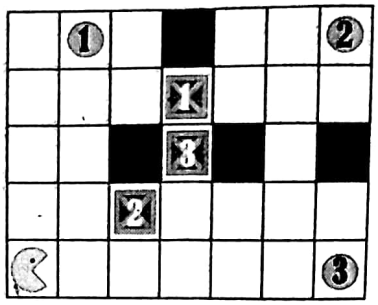
(d) [4 pts] Now suppose that all letters have cost 1, and that there is **a single correct password**, chosen uniformly at random from the state space. Candidate passwords can be checked using the decryption software, but the correct password is unknown. Which of the following statements is correct? *(The phrase "on average" reflects the fact that any password up to 10 letters long could be the correct one, all with equal probability.)*

- ● Given any heuristic, A^* search will, on average, expand fewer states than depth-first search.
- ○ There exists a heuristic, using which A^* search will, on average, expand fewer states than depth-first search.
- ○ Given any heuristic, A^* search will, on average, expand the same number of states as depth-first search.
- ○ Given any heuristic, A^* search will, on average, expand more states than depth-first search.
- ○ There exists a heuristic, using which A^* search will, on average, expand more states than depth-first search.

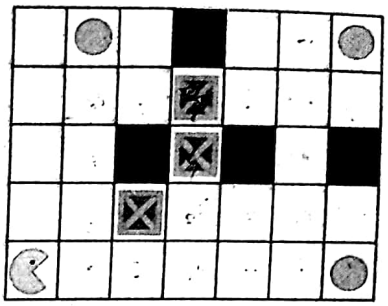
Q2. [12 pts] Pushing Boxes



(a) One box



(b) Numbered boxes and buttons



(c) Any box to any button

Pacman has to solve several levels of mazes by pushing boxes to circular buttons in the maze. Obviously, Pacman can only push a box (he does not have hands to pull it!). Pacman pushes a box by standing behind it and moving into its position. Pacman is not strong enough to push more than one box at a time. You can assume that the maze is $M \times N$ and that initially no box is upon any button. At each timestep, Pacman can just move either up, down, left, or right if he does not collide with any wall or the box that Pacman is pushing does not collide. Each action has a cost of 1. Actions that do not result in Pacman or a box being moved still have cost of 1. The figures display a possible configuration for each maze.

Note that for all parts of this question, d_{Man} is the Manhattan distance.

(a) In the first level, Pacman has to push a single box to a specific button (Figure 1a).

(i) [2 pts] What is the size of the minimal state space? Express your answer using the symbols M and N.

$(MN)^2$ (~~a tuple with xy coords~~)

(ii) [2 pts] What is the branching factor? The answer should be a whole, positive number.

4

(b) In the next level things get trickier for Pacman. Now, he has to push 3 boxes to 3 different buttons. Each box and button are numbered, and Pacman has to push the box to the button with the same number (Figure 1b).

(i) [2 pts] What is the size of the minimal state space? Express your answer using the symbols M and N.

$(MN)^4$

(ii) [2 pts] Which of the following heuristics are admissible?

- [x] d_{Man}(Pacman, button 1) + d_{Man}(Pacman, button 2) + d_{Man}(Pacman, button 3) - 3
- [x] d_{Man}(box 1, button 1) + d_{Man}(box 2, button 2) + d_{Man}(box 3, button 3)
- [ ] d_{Man}(box 1, box 2) + d_{Man}(box 1, box 3)
- [x] min(d_{Man}(box 1, button 1), d_{Man}(box 2, button 2), d_{Man}(box 3, button 3))
- [ ] None of the above

(c) In the third maze, the 3 boxes can go to any of the 3 buttons (Figure 1c).

(i) [2 pts] What is the size of the minimal state space? Express your answer using the symbols M and N.

~~$(MN)^4/3!$~~ $(MN)^4/3!$

(ii) [2 pts] Which of the following heuristics are consistent?

- [ ] $\max_{ij}$ d_{Man}(box i, button j)
- [x] $\min_{ij}$ d_{Man}(box i, button j)
- [ ] $\max_{j}$ d_{Man}(Pacman, button j)
- [ ] $\min_{i}$ d_{Man}(Pacman, box i) - 1
- [ ] None of the above

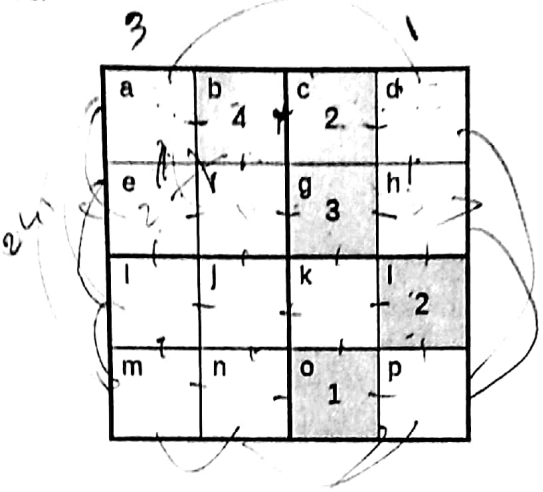
opt cost=

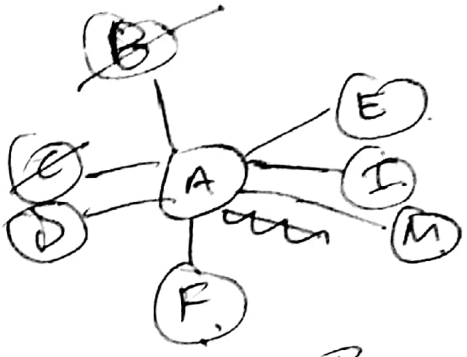
SID: _____

Q3. [12 pts] Simple Sudoku

Pacman is playing a simplified version of a Sudoku puzzle. The board is a 4-by-4 square, and each box can have a number from 1 through 4. In each row and column, a number can only appear once. Furthermore, in each group of 2-by-2 boxes outlined with a solid border, each of the 4 numbers may only appear once as well. For example, in the boxes a, b, e, and f, each of the numbers 1 through 4 may only appear once. Note that the diagonals do not necessarily need to have each of the numbers 1 through 4.

In front of Pacman, he sees the board below. Notice that the board already has some boxes filled out! Box $b = 4$, $c = 2$, $g = 3$, $l = 2$, and $o = 1$.





Explicitly, we represent this simple Sudoku puzzle as a CSP which has the constraints:

1. Each box can only take on values 1, 2, 3, or 4.
2. 1, 2, 3, and 4 may only appear once in each row.
3. 1, 2, 3, and 4 may only appear once in each column.
4. 1, 2, 3, and 4 may only appear once in each set of 2-by-2 boxes with solid borders.
5. $b = 4$, $c = 2$, $g = 3$, $l = 2$, and $o = 1$.

(a) [4 pts] Pacman is very excited and decides to naively do backtracking using only forward checking. Assume that he solves the board from left to right, top to bottom (so he starts with box a and proceeds to box b, then c, etc), and assume that he has already enforced all unary constraints. Pacman assigns 3 to box a. If he runs forward checking, which boxes' domains should he attempt to prune?

■ d ■ e ■ f ☐ h ■ i ☐ j ☐ k ■ m ☐ n ☐ p

(b) Pacman decides to start over and play a bit smarter. He now wishes to use arc-consistency to solve this Sudoku problem. Assume for all parts of the problem Pacman has already enforced all unary constraints, and Pacman has erased what was previously on the board.

(i) [4 pts] How many arcs are there in the queue prior to assigning any variables or enforcing arc consistency? Write your final, whole-number answer in the box below.

Answer: 35

(ii) [2 pts] Enforce the arc $d \to c$, from box d to box c. What are values remaining in the domain of box d?

- [x] 1
- [ ] 2
- [x] 3
- [x] 4

(iii) [2 pts] After enforcing arc consistency, what is the domain of each box in the first row?

Block a: 1, 3 Block b: 4

Block c: 2 Block d: 1, 3

SID: _____

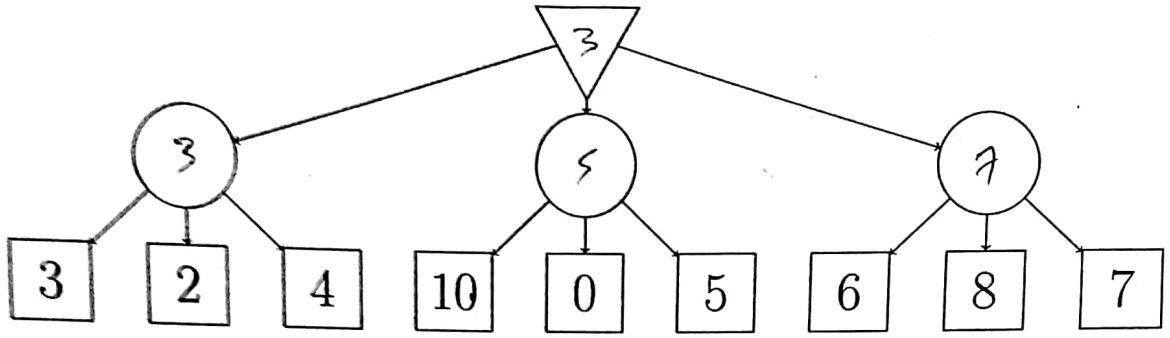
Q4. [14 pts] Expectimin

14/14

In this problem we model a game with a minimizing player and a random player. We call this combination "expectimin" to contrast it with expectimax with a maximizing and random player. Assume all children of expectation nodes have equal probability and sibling nodes are visited left to right for all parts of this question.

(a) [2 pts] Fill out the "expectimin" tree below.

x2



(b) [3 pts] Suppose that before solving the game we are given additional information that all values are non-negative and all nodes have exactly 3 children. Which leaf nodes in the tree above can be pruned with this information?

x3

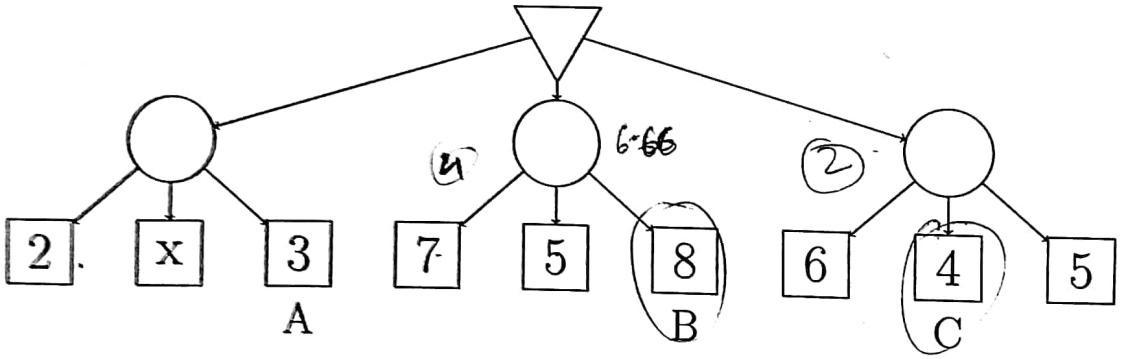
☐ 3 ☐ 2 ☐ 4 ☐ 10 ■ 0 ■ 5 ☐ 6 ☐ 8 ■ 7

(c) [3 pts] In which of the following other games can we also use some form of pruning?

x3

- ☐ Expectimax
- ☐ Expectimin
- ■ Expectimax with all non-negative values and known number of children
- ☐ Expectimax with all non-positive values and known number of children
- ☐ Expectimin with all non-positive values and known number of children

(d) For each of the leaves labeled A, B, and C in the tree below, determine which values of x will cause the leaf to be pruned, given the information that all values are non-negative and all nodes have 3 children. Assume we do not prune on equality.



Below, select your answers as one of (1) an inequality of x with a constant, (2) "none" if no value of x will cause the pruning, or (3) "any" if the node will be pruned for all values of x. Fill in the bubble, then if you select one of the inequalities, fill in the blank with a number.

+2 (i) [2 pts] A: ○ $x <$ _____ ○ $x >$ _____ ● None ○ Any

+2 (ii) [2 pts] B: ● $x <$ 8 ○ $x >$ _____ ○ None ○ Any

+2 (iii) [2 pts] C: ● $x <$ 1 ○ $x >$ _____ ○ None ○ Any

Q5. [14 pts] Utility of Sugar

14/14

(a) [4 pts] Every day after school Robert stops by the candy store to buy candy. He really likes Skittles, which cost \$4 a pack. His utility for a pack of Skittles is 30. KitKat bars cost \$1 each. His utility from KitKats is 6 for the first KitKat he buys, 4 for the second, 2 for the third, and 0 for any additional. He receives no utility if he doesn't buy anything at the store. The utility of m Skittle packs and n KitKats is equal to the following sum: utility of m Skittle packs PLUS utility of n KitKats.

In the table below, write the maximum total utility he can achieve by spending exactly each amount of money.

Robert:

\$0	\$1	\$2	\$3	\$4
0	6	10	12	30

For the remaining parts of this question, assume Sherry can achieve the following utilities with each amount of money when she goes to the candy store.

Sherry:

\$0	\$1	\$2	\$3	\$4
0	5	8	9	20

(b) Before Sherry goes to the store one afternoon, Juan offers her a game: they flip a coin and if it comes up heads he gives her a dollar; if it comes up tails she gives him a dollar. She has \$2, so she would end up with \$3 for heads or \$1 for tails.

(i) [2 pts] What is Sherry's expected utility at the candy store if she accepts his offer?

Answer: 7

(ii) [1 pt] Should she accept the game?

○ Yes ● No

(iii) [1 pt] What do we call this type of behavior?

○ Risk averse
● Risk prone
○ Risk neutral

(c) The next day, Sherry again starts with \$2 and Juan offers her another coin flip game: if heads he gives her \$2 and if tails she gives him \$2.

(i) [2 pts] What is Sherry's expected utility at the candy store if she accepts his offer?

Answer: 10

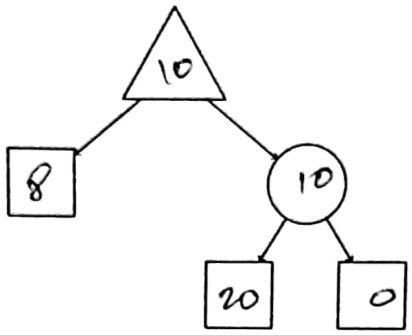
(ii) [1 pt] Should she accept the game?

● Yes ○ No

(iii) [1 pt] What do we call this type of behavior?

● Risk averse
○ Risk prone
○ Risk neutral

(iv) [1 pt] For this scenario (from part c), fill in the expectimax tree below with the utility of each node (including value, chance, and maximization nodes).



(d) [1 pt] If someone is risk averse in one lottery but risk prone in another, does that mean they must be behaving irrationally?

○ Yes ● No

Q6. [14 pts] Card Decision Processes

We have a card game, and there are three different cards: one has a value of 1, one a value of 2, and one a value of 3.

You have two actions in this game. You can either Draw or Stop. Draw will draw a card with face value 1, 2, or 3, each with probability $\frac{1}{3}$ (we assume we draw from a deck with replacement). You will bust if your hand's value goes above 5. This means that you immediately enter the terminal state "Done" and get 0 reward upon that transition.

Stop will immediately transition to the "Done" state, and receive a reward, which is the value of the cards in your hand. That is, $R(s, \text{Stop}, \text{"}Done\text{"})$ will be equal to s, or your hand value.

The state in this MDP will be the value of the cards in your hand, and therefore all possible states are "0", "1", "2", "3", "4", "5", and "Done", which is all possible hand values and also the terminal state. The starting state will always be "0", because you never have any cards in your hand initially. "Done" is a terminal state that you will transition to upon doing the action Stop, which was elaborated above.

Discount factor $\gamma = 1$.

(a) [6 pts] Fill out the following table with the optimal value functions for each state.

States	0	1	2	3	4	5
$V^*(s)$	0	1	2	3	4	5

(b) [2 pts] How many iterations did it take to converge to the optimal values? We will initialize all value functions at iteration 0 to have the values 0. Take the iteration where all values first have the optimal value function as the iteration where we converged to the optimal values.

○ 0 ● 1 ○ 2 ○ 3 ○ 4 ○ 5 ○ ∞

(c) [3 pts] What was the optimal policy that you found above? "D" stands for the action "Draw" and "S" stands for the action "Stop".

States	0	1	2	3	4	5
$\pi^*(s)$	■ D □ S	■ D □ S	■ D □ S	■ D ■ S	□ D ■ S	□ D ■ S

Q(s, stop) = R(s, stop, done) + γ V(Done) = s.

init = 0. so first update all are try to stop.

iter 1: 0 1 2 3 4 5. Draw for state 3:

iter 2: = $\frac{1}{3} \times 5 + \frac{1}{3} \times 4 + 0$, 3 no equal score.

= 2 → 2, 1 → 1, 0 → 0.

iter 3: $\frac{10}{3}$, $\frac{4}{3}$, 4, 3, 4, 5

iter 4: $\frac{32}{9}$, $\frac{4}{3}$, 4 no 3, 4, 5

Q7. [14 pts] Square World

10/14

SID: _____

In this question we will consider the following gridworld:

0	0	0	0	0	0
+1	A	B	C	D	+100
0	0	0	0	0	0

Every grid square corresponds to a state. If a state is annotated with a number, it means that after entering this state only one action is available, the Exit action, and this will result in the reward indicated by the number and the episode will end. There is no reward otherwise. For the other 4 states, named A, B, C, D, two actions are available: Left and Right. Assume $\gamma = 1$ in this problem.

(a) [4 pts] Assume that the failure probability for the actions Left and Right is 0.5, and in case of failure the agents moves up or down, and the episode is terminated after the Exit action. What are the optimal values?

+4

States	A	B	C	D
$V^*(s)$	16.25	12.5	25	50

(b) [4 pts] Still assume the failure probability in the previous part. Now assume further that there is an *integer* living reward r when the episode is not terminated. Is there a value of r that would make the optimal policy **only** decide Left at state D? If so, what's the minimum value of r?

$V^*(D) = \max(4 + 0.5(V(100)) + 0.5(V(0)),$
$4 + 0.5(V(C)) + 0.5(0))$

Answer: ≥ -51 $V(C) > V(100)$. So $V^*(x) = 24$.

(c) [4 pts] Assume we collected the following episodes of experiences in the form (state, action, next state, reward): (we use $X1$ and $X100$ to denote the leftmost and rightmost states in the middle row and Done to indicate the terminal state after an Exit action).

$$(B, Left, A, 0), (A, Left, X1, 0), (X1, Exit, Done, +1)$$
$$(B, Right, C, 0), (C, Right, D, 0), (D, Right, X100, 0), (X100, Exit, Done, +100)$$

If we run Q-learning initializing all Q-values equal to 0, and with appropriate stepsizing, replaying each of the above episodes infinitely often till convergence, what will be the resulting values for:

(State, Action)	(B, Left)	(B, Right)	(C, Left)	(C, Right)
$Q^*(s, a)$	1	100	0	100

(d) [2 pts] Now we are trying to do feature-based Q-learning. Answer the below True or False question.
There exists a set of features that are functions of state only such that approximate Q-learning will converge to the optimal Q-values.

● True ○ False

Q8. [6 pts] Exploring the World

In this question, our CS188 agent is stuck in a maze. We use Q learning with an epsilon greedy strategy to solve the task. There are 4 actions available: north (N), east (E), south (S), and west (W).

(a) [2 pts] What is the probability of each action if the agent is following an epsilon greedy strategy and the best action in state s under the current policy is N? Given that we are following an epsilon greedy algorithm, we have a value ϵ. Use this value ϵ in your answer. $p(a_i|s)$ is the probability of taking action a_i in state s.

$p(N|s)$ _____ $p(E|s)$ _____

$p(S|s)$ _____ $p(W|s)$ _____

(b) [2 pts] We also modify the reward original reward function $R(s, a, s')$ to visit more states and choose new actions. Which of the following rewards would encourage the agent to visit unseen states and actions? $N(s, a)$ refers to the number of times that you have visited state s and taken action a in your samples.

- ☐ $R(s, a, s') + \sqrt{N(s, a)}$
- ☐ $R(s, a, s') + \sqrt{\frac{1}{N(s,a)+1}}$
- ■ $\sqrt{\frac{1}{N(s,a)+1}}$
- ■ $R(s, a, s') - \sqrt{N(s, a)}$
- ☐ $-\sqrt{\frac{1}{N(s,a)+1}}$
- ☐ $\exp(R(s, a, s') - N(s, a))$

(c) [2 pts] Which of the following modified rewards will eventually converge to the optimal policy with respect to the original reward function $R(s, a, s')$? $N(s, a)$ is the same as defined in part (b).

- ☐ $R(s, a, s') + \sqrt{N(s, a)}$
- ■ $R(s, a, s') + \sqrt{\frac{1}{N(s,a)+1}}$
- ☐ $\sqrt{\frac{1}{N(s,a)+1}}$
- ☐ $R(s, a, s') - \sqrt{N(s, a)}$
- ☐ $-\sqrt{\frac{1}{N(s,a)+1}}$
- ☒ $\exp(R(s, a, s') - N(s, a))$