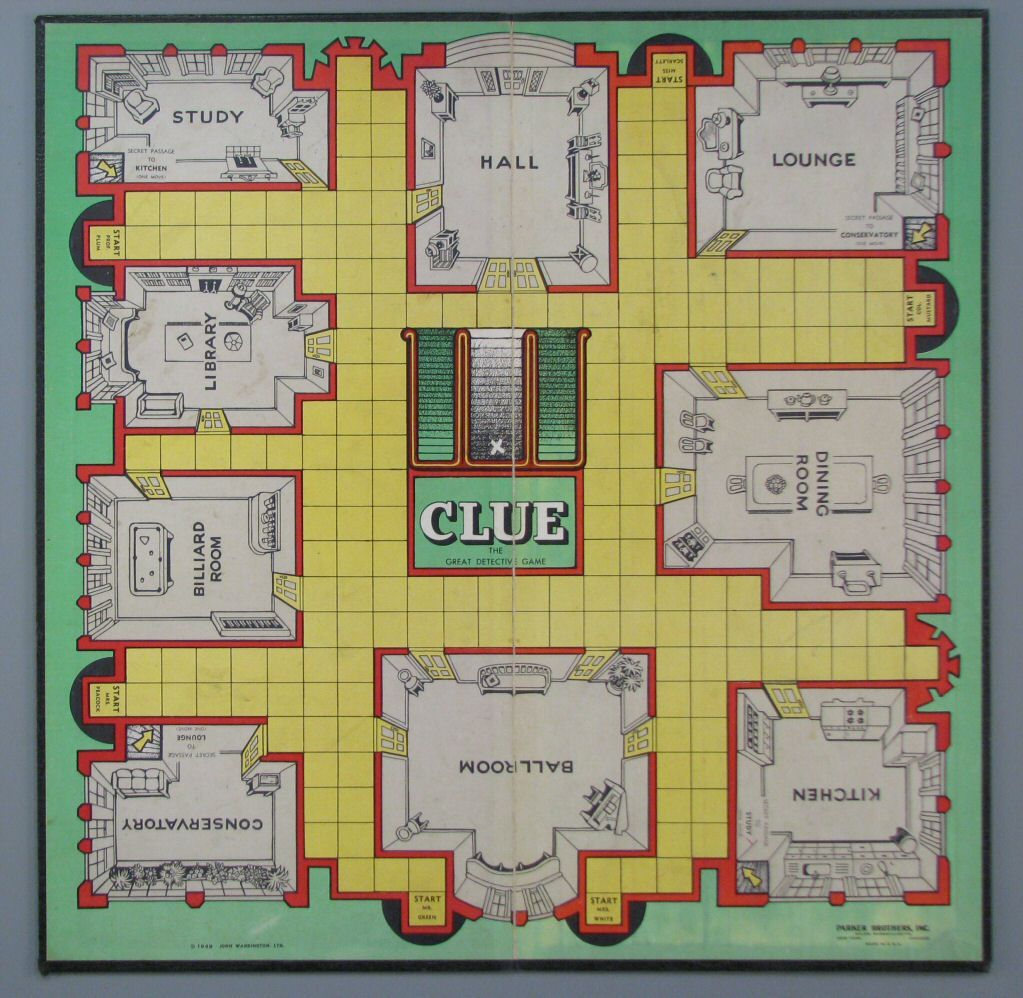
Exam 2

CS 4731/7632

Due May 1, 2024, 11:59pm

Submit to Gradescope

**Question 1:** Suppose you are developing a game with procedurally generated murder mysteries. In the game, a murder involves a murderer, a victim, a place of the crime, a weapon, and a motive (a reason why the murder was committed). You will design a grammar to generate the clues to the murderer, weapon, and motives. Use the image below as a possible map of locations where the body and murder weapon would be found.



The grammar should be able to produce instructions to a hypothetical game engine. Assume that game engine has functions called:

* spawn\_body(location)
* spawn\_weapon(type, location)
* spawn\_innocent\_suspect(name, location, occupation)
* spawn\_murderer(name, location, occupation)

Locations can be one of the rooms in the image above. The type of weapon can be one of { candlestick, dagger, revolver, lead pipe, wrench, and rope}. Names can be whatever you want, but no two people can have the same name.

A game should have one murderer and four innocent people who are suspects because they happen to be in the house. The game should have exactly one murder weapon, which can be found in any room. The occupation of the murderer must be related to the weapon in some way (for example, the weapon is wrench and the murderer’s occupation is plumber).

**a.** (4 points) Use a standard context free grammar that will produce a valid game.

**b.** (1 point) Show one possible string that can be generated.

**c.** (1 point) The output of the grammar should be instructions that can be used to configure the game engine. Describe how the output of the grammar will be used to call the functions that spawn the body, weapon, murderer, and innocent suspects.

**Question 2.** (1 point) Suppose we are using a genetic algorithm to procedurally generate levels for a game in which a character must do a variety of activities, such as kill monsters, jump over holes, and collect coins. The genetic algorithm must lay out the monsters, holes, and coins on a grid. We use a fitness function that gives specific target numbers for different types of content to be included in a level (e.g., the target number of coins). Is a genetic algorithm guaranteed to create a level that includes the precise number of each type of content requested? Answer “yes” or “no” and explain why. A sentence or two should suffice.

**Question 3.** (4 points) A game company wants to develop a game with a million levels. Under what circumstances would the game company choose design-time PCG instead of run-time PCG (Circle or underline “design-time” or “run-time”)? Provide a brief explanation of a sentence or two why design-time or run-time is the best choice. A sentence or two for each part should suffice.

1. The game has very little computation time for AI,   
   a lot of storage space is available, and the PCG   
   process has NP-hard complexity design-time run-time

Why:

1. The game has a lot of computation time for AI,   
   very little storage is available, and the PCG process   
   has polynomial complexity design-time run-time

Why:

1. The game has a little computation time for AI,   
   very little storage is available, and the PCG process   
   has polynomial complexity design-time run-time

Why:

1. The game has a lot of computation time for AI,   
   a lot of storage space is available, and the PCG   
   process has NP-hard complexity design-time run-time

Why:

**Question 4.** (3 points) For each statement below, explain why minimax, Monte Carlo Tree Search (MCTS), and deep Q-learning (DQL) may or may not be an appropriate choice of algorithm to play the game. One sentence per algorithm should suffice.

a. Actions can have stochastic effects.

Minimax:

MCTS:

DQL:

b. Can handle games with too many states to store in memory.

Minimax:

MCTS:

DQL:

c. You require a guarantee that the optimal action is performed in every state and it is possible to store all states in memory if necessary.

Minimax:

MCTS:

DQL:

**Question 5.** (1 point) Explain why Deep Q-Learning uses a replay buffer instead of directly learning from the agent’s state-action-state transitions at the very moment that the transactions are made.

**Question 6.** Genetic algorithms have two mechanisms for producing individuals: mutation and cross-over. Answer the following questions about genetic algorithms.

**a.** (1 point) What would happen if you only implemented mutation? What is a parameter you could change to increase the likelihood of finding the global maximum?

**b.** (1 point). What would happen if you only implemented cross-over? Is there any guarantee that a GA that only uses cross-over finds a global max or a local max?

**Question 7.** (1 point) Consider the following game boards:

Game A:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| st. |  |  |  | -2 |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  | 5 |  |  |
| -2 |  |  |  |  |  |
|  |  |  |  |  | 10 |

Game B:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| st. | 1 | 2 | 3 | 4 | 5 |
| 1 | 2 | 3 | 4 | 5 | 6 |
| 2 | 3 | 4 | 5 | 6 | 7 |
| 3 | 4 | 5 | 6 | 7 | 8 |
| 4 | 5 | 6 | 7 | 8 | 9 |
| 5 | 6 | 7 | 8 | 9 | 10 |

The agent starts in the position labeled “st.”. It can move up, down, left, or right. The numbers indicate the points the agent will receive when it transitions into the cell. If the cell is empty, the agent receives zero points for transitioning into that cell.

You implement the same Deep Q Learning algorithm on each game. Between Game A and Game B, which agent will converge faster. Explain why. Fewer than five sentences should be sufficient.

**Question 8.** (2 point). Suppose you have a 10-by-10 grid world. The agent starts in the corner at position (1, 1), and there is a single state in the opposite corner (10, 10) that provides a reward of ten. All other states provide zero reward. What would happen to the utility values of all the states if you implemented tabular Q learning and gamma 𝝲 to 0.0? What would happen to the utility values of all the states if you implemented tabular Q learning and gamma 𝝲 to 1.0? One to two sentences per gamma value should be sufficient.

𝝲 = 0.0

𝝲 = 1.0