

**DSE 3260 ARTIFICIAL INTELLIGENCE LABORATORY**  
**6th Semester BTech Data Science & Engineering**  
**SECTION B**

**WEEK 1: Tutorial on OpenAI Gymnasium**

1. Take the tutorial: Getting Started With OpenAI Gym: The Basic Building Blocks

<https://www.gymnasium.dev/content/tutorials/>

<https://blog.paperspace.com/getting-started-with-openai-gym/>

2. Use the CartPole-v0 environment and write a program to :
  - a. Implement the CartPole environment for a certain number of steps
  - b. Implement the CartPole environment for a certain number of episodes
  - c. Compare and comment on the rewards earned for both approaches.
  - d. Plot the cumulative reward of the games and note down the results.

**WEEK 2: Problem Solving Agents & SEARCH, POLICY**

**QUESTION 1: The Game :**

According to the “Six Degrees of Kevin Bacon” game, anyone in the Hollywood film industry can be connected to Kevin Bacon within six steps, where each step consists of finding a film that two actors both starred in. To solve the problem, find the shortest path between any two actors by choosing a sequence of movies that connects them. For example, the shortest path between Jennifer Lawrence and Tom Hanks is **2**: Jennifer Lawrence is connected to Kevin Bacon by both starring in “X-Men: First Class,” and Kevin Bacon is connected to Tom Hanks by both starring in “Apollo 13.”

**Problem Solving Agent:**

Given two actors nodes in the graph we need to find the distance (shortest path) between the nodes. Write a python program to determine how many “degrees of separation” apart two actors are. Find the distance or the degree of separation., using

- a. Breadth first search
- b. Depth first search

**Distribution Code:**

Data & Download the distribution code from:

<https://cdn.cs50.net/ai/2020/x/projects/0/degrees.zip>

The distribution code contains two sets of CSV data files: one set in the large directory and one set in the small directory. Use the small dataset for ease of testing and experimentation. Each dataset consists of three CSV files.

1. **small/people.csv**: each person has a unique id, corresponding with their id in IMDb’s database. They also have a name, and a birth year.
2. **small/movies.csv**: You’ll see here that each movie also has a unique id, in addition to a title and the year in which the movie was released.
3. **small/stars.csv**: This file establishes a relationship between the people in people.csv and the movies in movies.csv. Each row is a pair of a person\_id value and movie\_id value.

For example: The first row (ignoring the header), for example, states that the person with id 102 starred in the movie with id 104257. Checking that against people.csv and movies.csv, you'll find that this line is saying that Kevin Bacon starred in the movie "A Few Good Men."

#### 4. degrees.py:

At the top, several data structures are defined to store information from the CSV files. The names dictionary is a way to look up a person by their name: it maps names to a set of corresponding ids (because it's possible that multiple actors have the same name). The people dictionary maps each person's id to another dictionary with values for the person's name, birth year, and the set of all the movies they have starred in. And the movies dictionary maps each movie's id to another dictionary with values for that movie's title, release year, and the set of all the movie's stars. The load\_data function loads data from the CSV files into these data structures.

The main function in this program first loads data into memory (the directory from which the data is loaded can be specified by a command-line argument). Then, the function prompts the user to type in two names. The person\_id\_for\_name function retrieves the id for any person (and handles prompting the user to clarify, in the event that multiple people have the same name). The function then calls the shortest\_path function to compute the shortest path between the two people, and prints out the path.

#### Code the shortest\_path function using

- a. **Breadth First Search**
- b. **Depth First Search**

**QUESTION 2:** Use the **CartPole-v0** environment, the status of the system is specified by an "observation" of four parameters ( $x$ ,  $v$ ,  $\theta$ ,  $\omega$ ), where

- $x$ : the horizontal position of the cart (positive means to the right)
- $v$ : the horizontal velocity of the cart (positive means moving to the right)
- $\theta$ : the angle between the pole and the vertical position (positive means clockwise)
- $\omega$ : angular velocity of the pole (positive means rotating clockwise)

**Use gym and write the following policies for 100 episodes :**

- a. **Theta policy:** if the pole is tilted to the left ( $\theta < 0$ ), then push the cart to the left, and vice versa.
- b. **Omega policy :** when the pole is moving away from the vertical position ( $\omega < 0$ ), push the cart to the left (action = 0). And vice versa.
- c. Plot the cumulative rewards (based on steps) for each policy , write down the average rewards with standard deviation.
- d. Comment on performance of policies.