

## **Lab Report for Lab-2 COMS 472**

**RECEIVED AN EXTENSION TILL Tue, Apr 2, 2024 23:59 to submit this assignment**

### **Introduction**

In this lab assignment, an AI agent capable of playing the game Gomoku (also known as Five-in-a-Row) was developed.

### **Answer to Question 6**

What happens to increasing search depth? (found in the main method of gomoku.py)

To analyze the effect of increasing search depth on the agent's performance, the agent was tested with different depth values (1, 2, 3, and 4). The time taken for the agent to make a decision at each depth level was noted.

The results showed that **as the search depth increased**, the agent's **decision-making time grew exponentially**. However, the **quality of the agent's moves also improved** with increased depth, as it could anticipate and plan further ahead. A balance must be struck between search depth and computational efficiency to ensure optimal performance.

It grew more challenging to win against the AI as the depth increased.

### **Answer to Question 7**

What happens when we improve the evaluation function? (compute\_utility / ( \_2))

The two implemented evaluation functions, compute\_utility, and compute\_utility\_2, were compared by playing games against each other.

**compute\_utility** demonstrated a more *balanced approach*, considering both offensive and defensive aspects, while **compute\_utility\_2** prioritized *defensive play*.

The results showed that compute\_utility generally outperformed compute\_utility\_2, as it could capitalize on offensive opportunities while maintaining a solid defense. However, compute\_utility\_2 was effective in situations where preventing the opponent's winning moves was crucial.

This was tested by making methods that involved the agents playing against one another. (Check the commented code in the gomoku.py to do this)