**List only the BITS (Name) of active contributors in this assignment:**

**1. 2023aa05152 - Shruti S Kumar**

**2. 2023aa05072 - Peyala Samarasimha Reddy**

**3. 2023aa05195 - Viswanath Hemanth Chadalawada**

**4. 2023aa05229 - POLAVARAPU SATYA DURGA LALITHA RAO SARMA**

**5. 2023aa05930 - Sridhar K**

**Problem Statement 1**

**PEAS:**

1. **Performance measure:**

- The performance measure in this game is determined by how well a player maximizes their sum of chosen numbers while minimizing their opponent's sum.

- A player aims to either have a higher sum, equal sum, or lose by the smallest possible margin if winning or equalizing is not feasible.

1. **Environment:**

- The environment consists of a set of natural numbers {1, 2, 3, ..., n}, where n is agreed upon before the game.

- Players take turns choosing numbers from this set, with the chosen numbers being removed and unavailable for subsequent choices.

- The environment dynamically updates as numbers are chosen and sums are calculated, eventually leading to the end of the game when all numbers are chosen.

1. **Actuators:**

- Actuators in this game are the actions taken by each player to choose numbers from the set.

- Players alternate turns, with each player choosing one or more numbers in a way that maximizes their sum while ensuring it does not exceed or equal their opponent's previous sum.

1. **Sensors:**

- Sensors gather information about the state of the game, including the remaining numbers in the set, the sums chosen by each player, and the decisions made by the opponent.

- Players use this information to make strategic decisions about which numbers to choose to maximize their own sum while preventing their opponent from doing the same.

**Min-Max Algorithm Design of Logic:**

1. Game Representation: In this game, the states represent different configurations of chosen and remaining numbers. Each node in the game tree represents a state of the game, and the edges represent the possible moves (i.e., choosing a number) that players can make from one state to another.

2. Objective:

- P1: This player aims to maximize their sum of chosen numbers.

- P2: This player aims to minimize P1's sum while maximizing their own sum.

3. Recursion:

- The Min-Max algorithm recursively explores the game tree, starting from the initial state.

- At each level of the tree, it alternates between maximizing and minimizing players.

- It evaluates the possible outcomes of each move by considering the opponent's best response.

4. **Static Evaluation Function:**

- At terminal states (i.e., when all numbers have been chosen), the algorithm evaluates the utility or score of the game state.

- The evaluation function calculates the difference between the sums of P1 and P2.

- A positive value indicates P1's advantage, a negative value indicates P2's advantage, and zero indicates a tie.

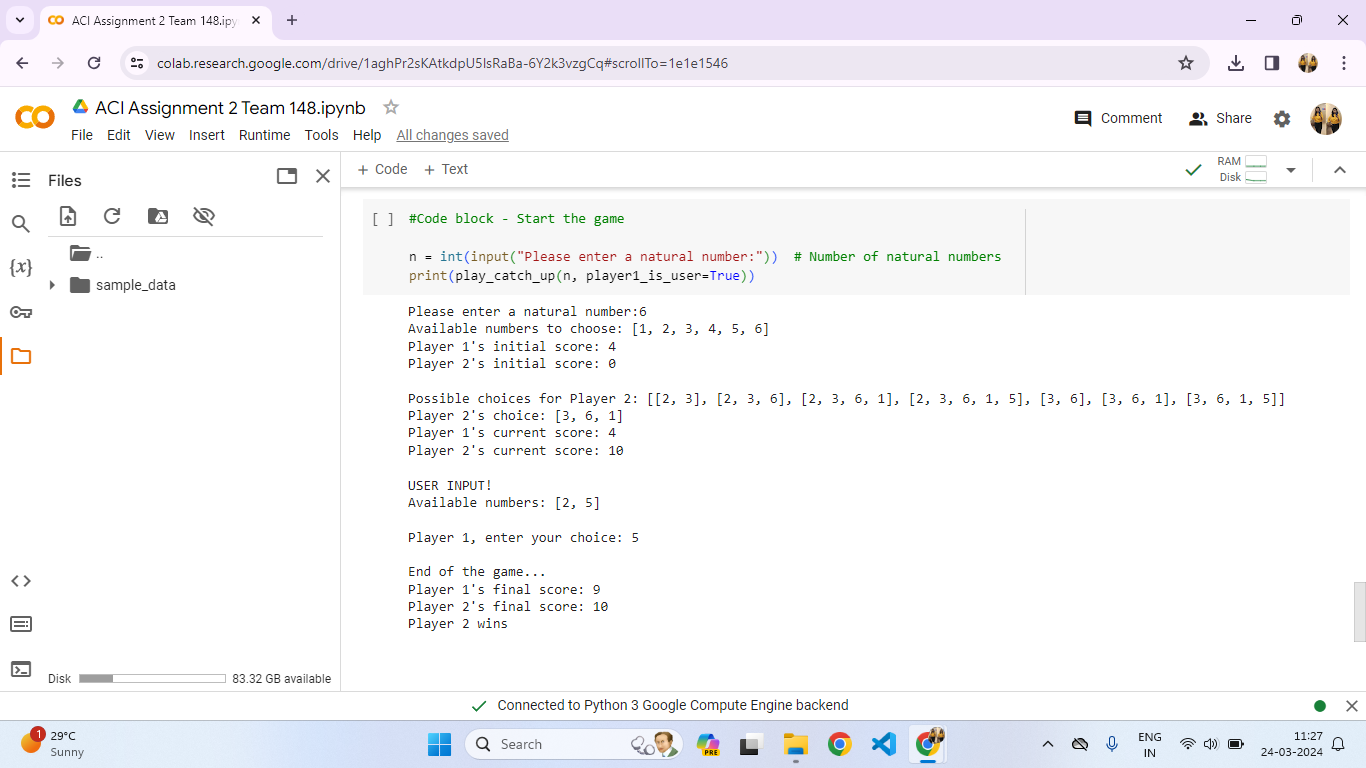
5. Backtracking:

- As the algorithm traverses the game tree, it backtracks to previous states, updating the best move found so far based on the evaluation of child nodes.

- This ensures that the algorithm chooses the optimal move at each decision point.

- This reduces the number of nodes evaluated, significantly speeding up the search process.

**OUTPUT Screenshot:**

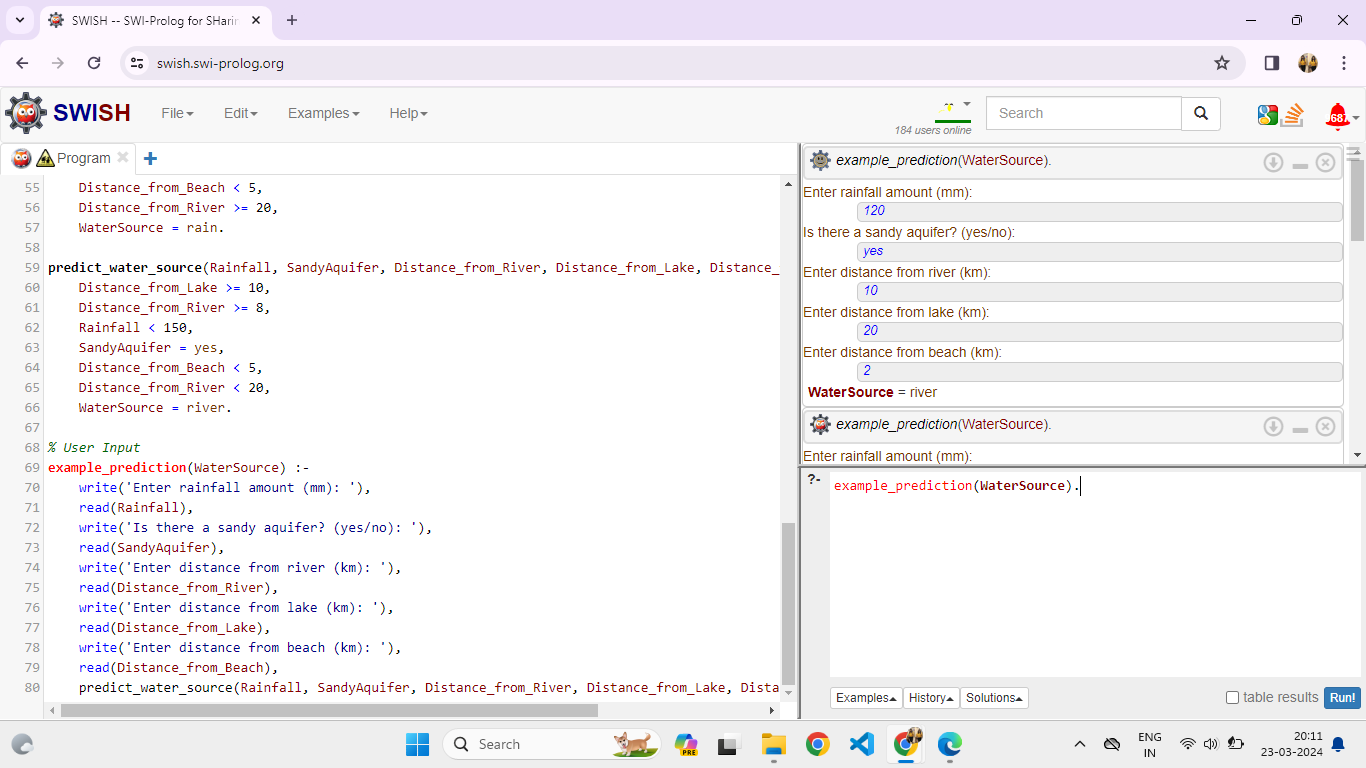
****

**Problem Statement 2**

**Example 1:**

%predict\_water\_source(120, yes, 10, 20, 2, WaterSource).

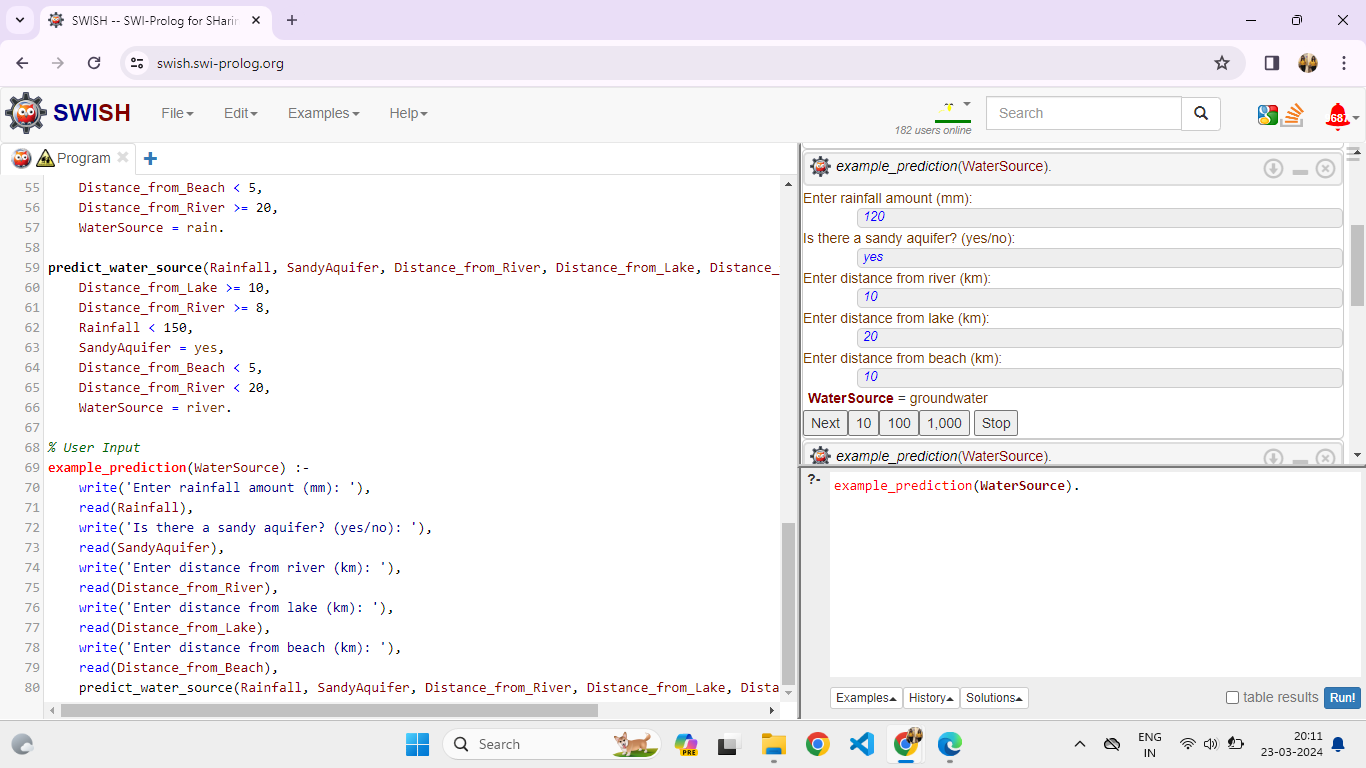
example\_prediction(WaterSource).



**Example 2:**

%predict\_water\_source(120, yes, 10, 20, 10, WaterSource).

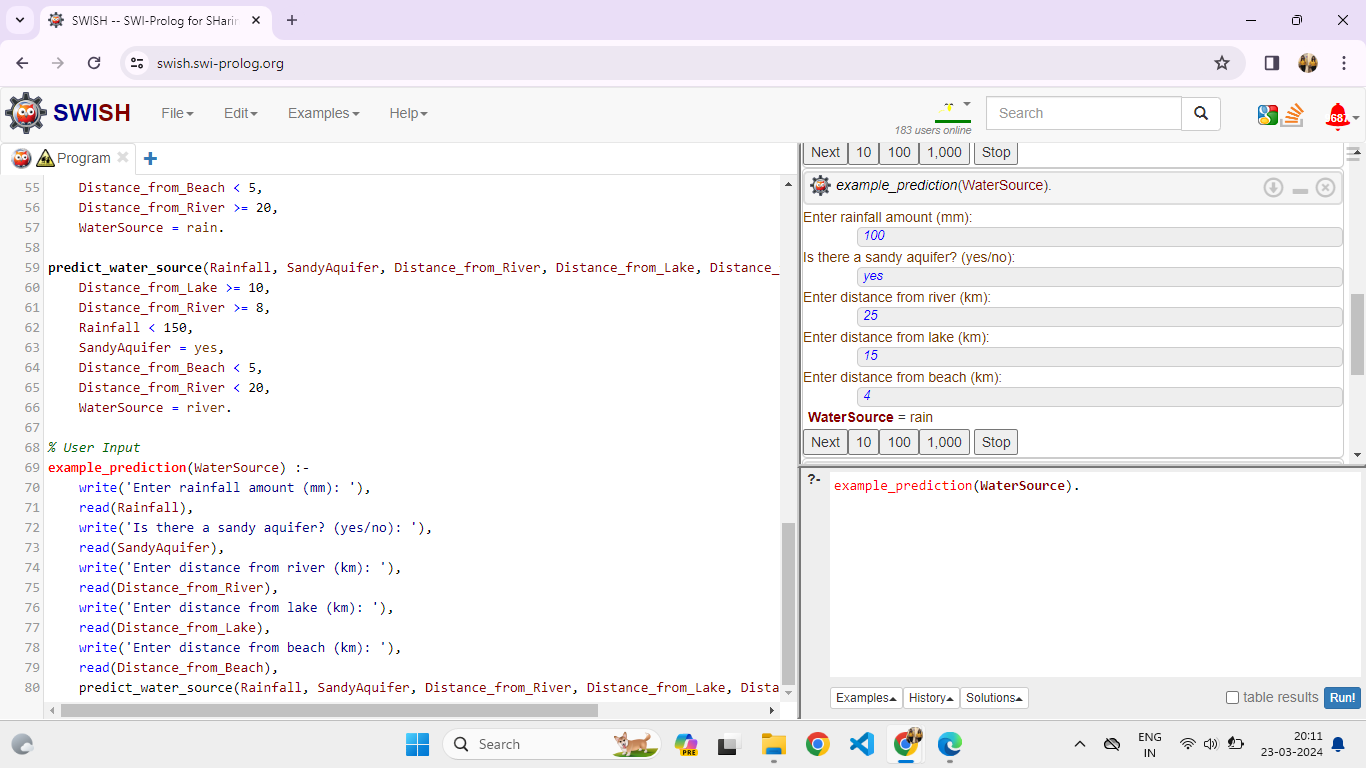
example\_prediction(WaterSource).



**Example 3:**

%predict\_water\_source(100, yes, 25, 15, 4, WaterSource).

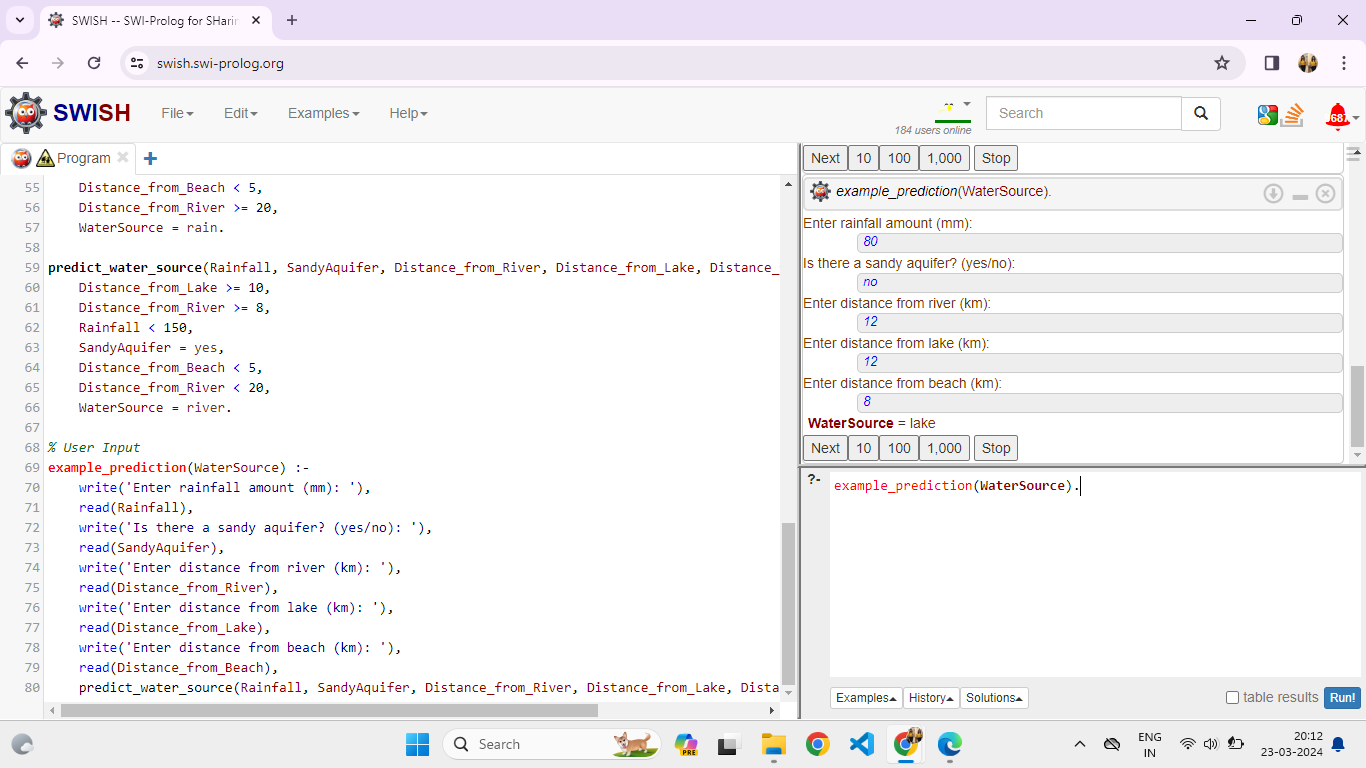
example\_prediction(WaterSource).



**Example 4:**

%predict\_water\_source(80, no, 12, 12, 8, WaterSource).

example\_prediction(WaterSource).



**Decision Tree interpretation:**

1. Rainfall Intensity:

◦ If the location receives less than 150 mm/month of rainfall, it is not suitable for relying on rainfall as a water source.

◦ Otherwise, if the rainfall intensity is 150 mm/month or more, consider using rainfall.

2. Distance to River:

◦ If the location is more than 10 km away from a perennial river, river water is not the best choice.

◦ Otherwise, if the distance to the river is 8 km or less, consider utilizing river water.

3. Aquifer Type:

◦ If there is a sandy aquifer, groundwater can be a viable option.

◦ Otherwise, if the aquifer is not sandy, groundwater may not be the best choice.

4. Distance to Lake and Beach:

◦ If the location is 20 km away from the lake, consider using lake water.

◦ If the location is 2 km away from the beach, sandy aquifers near the beach could be a good source of groundwater.

Based on the given conditions for **Example 1**:

• Rainfall: 120 mm/month (less than 150 mm/month)

• River Distance: 10 km (within the 8 km limit)

• Aquifer Type: Sandy

• Lake Distance: 20 km

• Beach Distance: 2 km

The most suitable water resource for this community would be **river water**.