|  |
| --- |
| Artificial and Computational Intelligence (S2-23\_AIMLCZG557) |
| **Assignment 2 – Statement 1** |
| By Group 1, 17th September 2024 |

**Submitted by Group 1 with 100% contribution from all participants:**

*Name:* ***Reddy Balaji .C*** *(BITS ID:* ***2023AC05862****); Email:* [*2023ac05862@wilp.bits-pilani.ac.in*](mailto:2023ac05862@wilp.bits-pilani.ac.in)

*Name:* ***Saurabh Jalendra*** *(BITS ID:* ***2023AC05912****); Email:* [*2023ac05912@wilp.bits-pilani.ac.in*](mailto:2023ac05912@wilp.bits-pilani.ac.in)

*Name:* ***Tushar Shandilya*** *(BITS ID:* ***2023AC05573****); Email:* [*2023ac05573@wilp.bits-pilani.ac.in*](mailto:2023ac05573@wilp.bits-pilani.ac.in)

*Name:* ***Bandana Kumari*** *(BITS ID:* ***2023AC05879****); Email:* [*2023ac05879@wilp.bits-pilani.ac.in*](mailto:2023ac05879@wilp.bits-pilani.ac.in)

*Name:* ***Monica Malik*** *(BITS ID:* ***2023AC05875****); Email:* [*2023ac05875@wilp.bits-pilani.ac.in*](mailto:2023ac05875@wilp.bits-pilani.ac.in)

**Problem Statement-1 Gaming**

**Catch-Up with numbers**

Problem statement is for a game called "Catch-Up," where two players are alternatively choosing numbers from a set of natural numbers and aim to accumulate a sum higher than or equal to their opponent's. The game involves strategy as well as randomness as players must select subsets of numbers whose sum equals or exceeds their opponent’s previous total.

**Key points of the game**

Here are the key points of the "Catch-Up with numbers" game with the PEAS (Performance, Environment, Actuators, Sensors) description:

1. Performance (P)

* Maximizing Sum: The primary goal for both players is to maximize the sum of their chosen numbers relative to their opponent.
* Minimizing Loss: If a win is unlikely, players aim to minimize the point difference between their total sum and their opponent’s total.
* Forcing a Tie: If winning or minimizing the loss is not achievable, players strive to equalize their sum with the opponent to force a tie.
* Optimal Decisions: Players must make efficient decisions by choosing the best possible combination of numbers during each turn to improve their chances of winning or minimizing the loss.

2. Environment (E)

* Finite Set of Numbers: The game is played with a finite set of natural numbers {1, 2, 3, ..., n}, where "n" is decided before the start of the game.
* Two Players: Players P1 and P2 alternate turns, with P1 going first.
* Game Conditions: Once a number is selected by a player, it is removed from the set and cannot be chosen again.
* Turn-based Constraints: During each turn, players must continue choosing numbers until the sum of their selected numbers equals or exceeds the opponent’s current sum.
* Game End: The game ends when all numbers are chosen, or there are no valid moves left for one or both players.

3. Actuators (A)

* Choosing Numbers: Players actively select one or more numbers from the remaining available set during their turn.
* Turn Management: A player's turn ends when their sum equals or exceeds their opponent's previous sum.
* Strategic Decision-Making: Players can choose to maximize their sum or minimize their loss by selecting the optimal combination of numbers available at each stage of the game.

4. Sensors (S)

* Player Sums: Players are aware of both their own and their opponent's current sum of chosen numbers.
* Remaining Numbers: Players know which numbers are still available and which ones have been selected.
* Opponent’s Moves: Players can observe and anticipate their opponent’s strategy based on the previously selected numbers.
* Game State: Players can sense when the game is nearing its conclusion and may adjust their strategy to either maximize their sum, minimize their loss, or force a tie depending on the situation.

This structure ensures that players operate under full observability and compete to maximize their advantage through each decision-making step of the game.

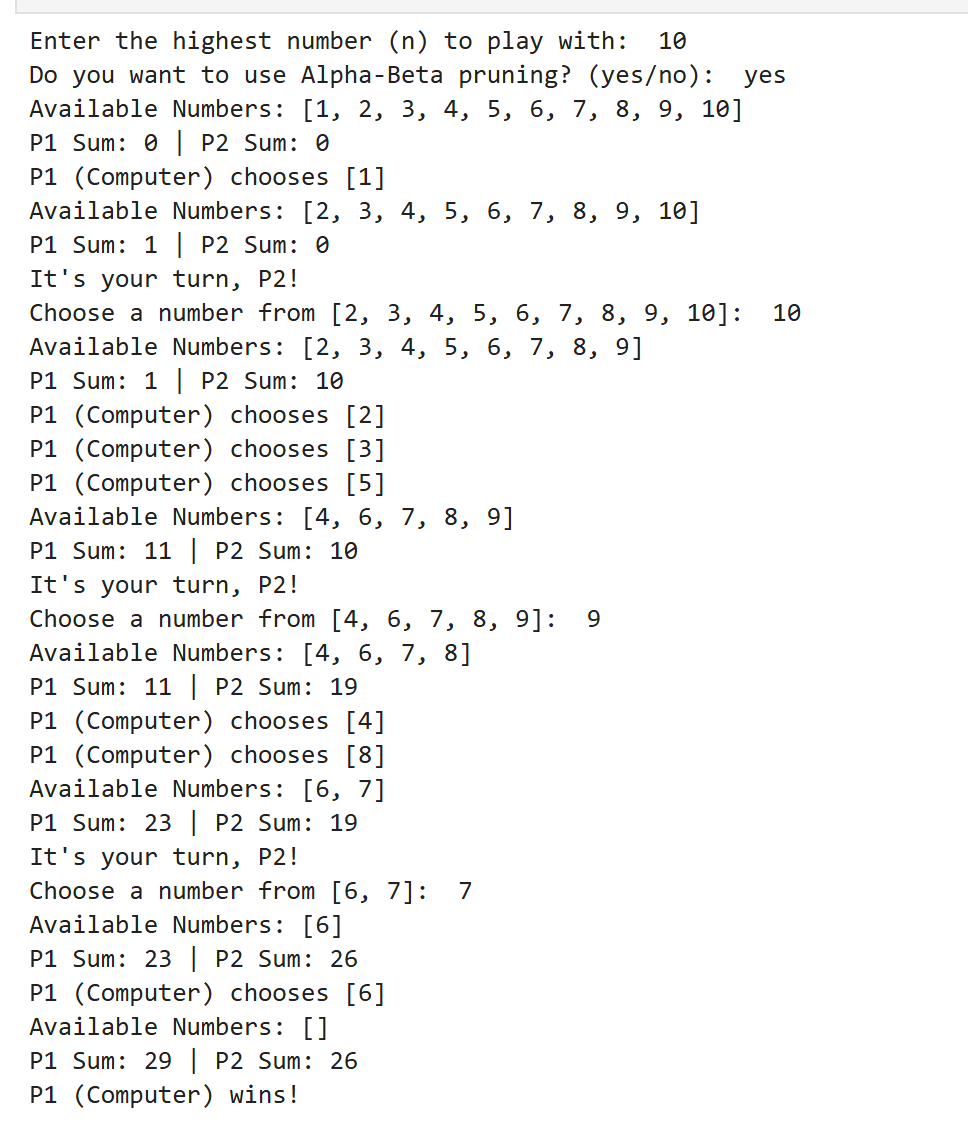
**Solution 1**

Jupyter Notebook with the Name “ PS1-Part1\_Group1\_ACI\_Assignment2.ipynb ” has been included as part of submission.

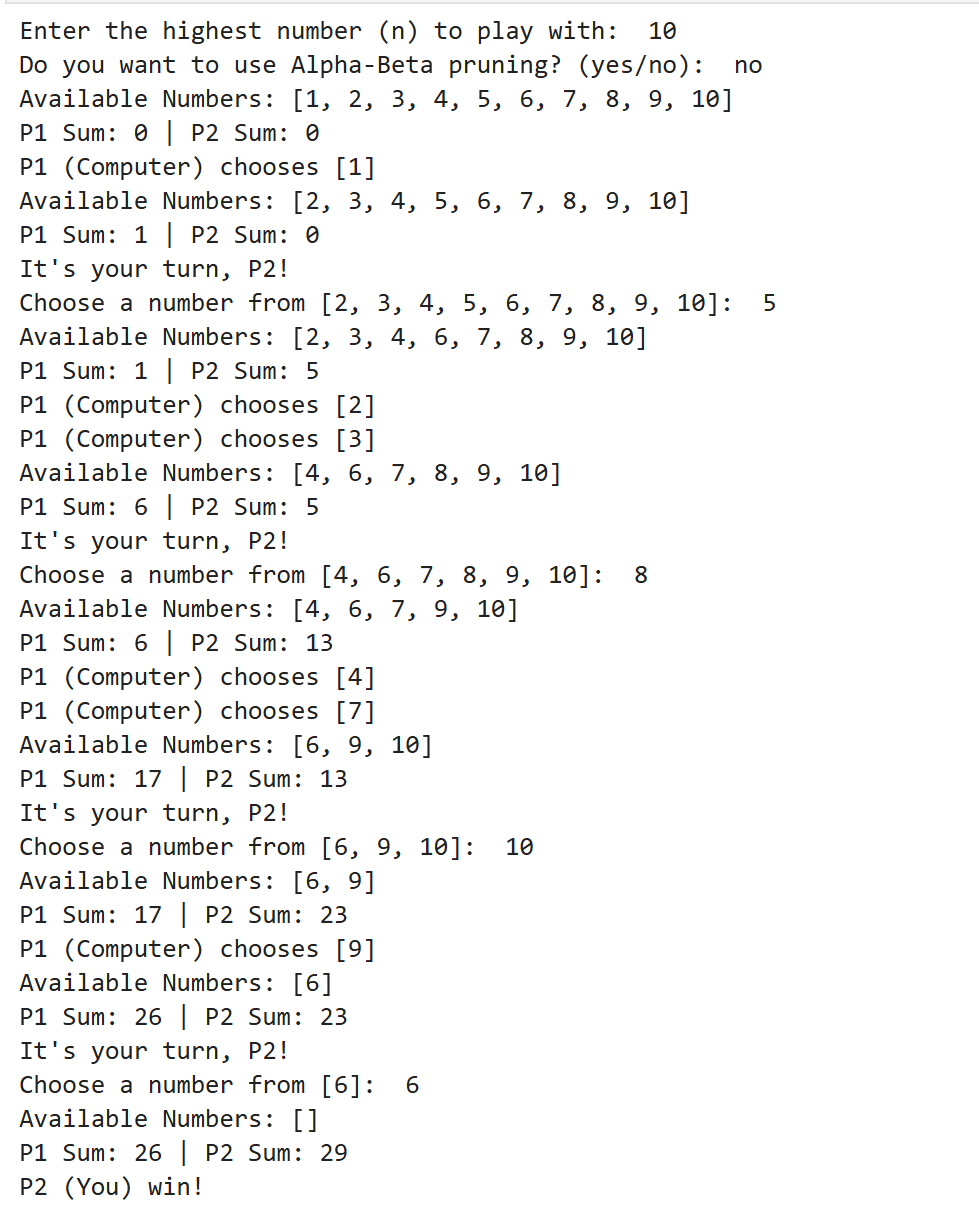
These snippets shows the interactive implementation, dynamic inputs-based run of the game with step wise board display and error free game ending.

Below are the snippets of some of the outputs of the game our code is performing :

* **When Alpha/Beta Pruning is “yes”**

****

* **When Alpha/Beta Pruning is “No”**

****

**A screenshot of a computer program

Description automatically generated**

**Problem Statement 2 : Logic**

Below decision tree is given in the problem statement based on which we have created Prolog rules to predict which water source is best for the community under the given conditions**.**

A diagram of a river

Description automatically generated

**Solution fpr Problem Statement 2**

Below is the Model’s Matrix we have derived post evaluation of the given Decision tree :

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Lake distance (km)** | **River distance (km)** | **Rainfall intensity (mm)** | **Sandy aquifer** | **Beach distance (km)** | **Water Source** |
| < 10 | - | - | - | - | Lake |
| ≥ 10 | ≥ 8 | < 150 | Yes | < 5 | River |
| ≥ 10 | ≥ 8 | < 150 | Yes | ≥ 5 | Groundwater |
| ≥ 10 | ≥ 8 | < 150 | No | - | Rain |
| ≥ 10 | ≥ 8 | ≥ 150 | - | - | Rain |
| ≥ 10 | < 8 | ≥ 200 | - | - | Rain |
| ≥ 10 | < 8 | < 200 | - | - | River |

We have defined and created the Prolog rules in the file named “ **PS1-Part2\_Group1\_ACI\_Assignment2.pl ”** which we have included as the submission for this Assignment.

Below are the sample prompts and the output of the prolog code which can be checked by running the code on any Prolog interpreter :

A screenshot of a computer program

Description automatically generated

A screenshot of a computer program

Description automatically generated

**Thank You!**

***Attached References:***

* *Problem Statement 1 Solution : Python File* ***(PS1-Part1\_Group1\_ACI\_Assignment2.ipynb)***
* *Problem Statement 2 Solution : Prolog File (****PS1-Part2\_Group1\_ACI\_Assignment2.pl)***