Implementation of a BoardGame

LUDO

**Group 12 Details**

1. **Rashika Jain AU2020178**
2. **Kanvi Patel AU2040235**
3. **Jatin Parmar AU2040118**
4. **Kishan Akbari AU2040155**

## Introduction

Ludo is a strategy board game that two to four players can play. The players need to compete with their four tokens in the game, consistent with the die rolls from start to end.

## Background

Ludo is derived from the Indian game Pachisi. The fun and its varieties are documented in numerous nations and under different names.

## Motivation

These days the world is moving towards Information Technology, there was a requirement for an enhancement for our manual ludo game.

## Literature Survey

By doing some research on the topic, we found out that Ludo is a board game that can be modelled as a first-order Markov chain as it is memoryless. Furthermore, the algorithm

can implement the Markov process by using a transition matrix.

## Contribution

Primarily, our work had four parts.

**1. Research 2. Coding**

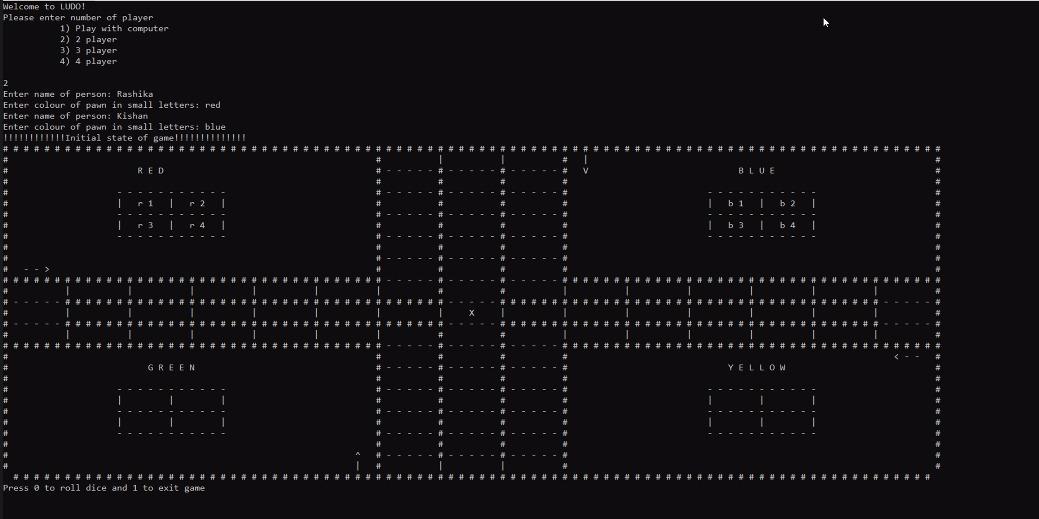
**3. Documentation 4. Presentation**

Each of us has contributed equally to the project but in different sectors of it. We kept in mind the strong points of each teammate and worked cooperatively. Jatin and Kanvi came up with the idea of making a game, and the four of us together chose LUDO. Kishan and Rashika did most of the research. All four of us coded specific segments of the project. Jatin did the final compilation of the project, and Kishan was responsible for debugging the compiled code. Coming to the documentation, Kanvi headed this department. Rashika handled the presentation and demonstration of the entire project.

## Mathematical model

In the algorithm, we used the first-order Markov chain. A transition matrix can be used to implement the Markov process. We used graphs, a disjoint union set, and the Dijkstra algorithm in our code. We used a graph as the data structure to connect the coordinates in our matrix. The pawn moves through the graph, traversing the matrix. The Dijkstra Algorithm was then applied. To find the shortest path, the Dijkstra algorithm is used. In our project, we used the Dijkstra algorithm to guide the pawn home.

## Numerical results



This is our game's user interface.

We roll the dice with zero and can end the game with one. When we get 6, the pawn will leave the house. We will have to keep pressing zero until we get a 6. After we get a 6, we must enter which pawn we want to move.

If we get three 6s in a row, our turn is over.

In the end, whichever player reaches the position in the middle, indicated by the sign X, wins the game.

## References

"Ludo Game Project In PYTHON." *Projectworlds*, <https://projectworlds.in/python-projects-with-source-code/ludo-game-project-in-python/>.

"Geometric Algorithms." *GeeksforGeeks*, 28 July 2021, <https://www.geeksforgeeks.org/geometric-algorithms/.>

Petersen, Vegard Krog. *Pachisi & Ludo - PC Games, Rules and History - Ludo*, <http://pachisi.vegard2.net/ludo.html>

*Ludo-Official BSE Rules*. <http://www.banglagym.com/wp-content/uploads/2016/06/Ludo_Rules.pdf>.

.