### Artificial Intelligence

### AL2001

### Project Report



### Two Player Dynamic Pacman

### 

### Group Members:

### Laiba Bint-e-Zia[23K-0006]

### Amna[23K-0066]

### Alisha Zaidi[23K-0025]

### Layyana Junaid[23K-0056]

AI Project Report

**Project Title:** AI-Powered Two Player Dynamic Pacman

**Submitted By:** Laiba Bint-e-Zia

**Course:** Artificial Intelligence

**Instructor:** Sir Bilal Ahsan and Miss Ramsha Jatt

**Submission Date:** 12th May 2025

# 1. Executive Summary

## Project Overview:

This project involved the development of a two-player version of Pacman, enhanced with AI and dynamic map mechanics. The map regenerates every 30 seconds, repositioning the Pacman and altering the game layout. Ghosts, acting as intelligent adversaries, utilize multiple AI search algorithms to locate and chase Pacmans. The goal for the ghosts is to eliminate both players, while the Pacman players aim to outscore each other or survive.

# 2. Introduction

## Background:

Pacman is a classic arcade maze game where a single player navigates through a fixed maze, consuming pellets and avoiding ghosts. Traditionally, the game involves four ghosts with predefined chasing logic, and a single player attempting to clear the map while utilizing power pellets to turn the ghosts vulnerable. Our version introduces competitive multiplayer mechanics and dynamic map elements to enhance strategic depth and replayability.

## Objectives of the Project:

- Implement a dynamic, regenerating map that updates every 30 seconds.  
- Develop multiple AI algorithms for ghosts to intelligently track and chase Pacmans.  
- Introduce new game mechanics such as power-ups (e.g., ghost freezing).  
- Enable competitive two-player gameplay with win/loss conditions based on scoring and survival.

# 3. Game Description

## Original Game Rules:

In traditional Pacman, the player navigates a maze to consume all pellets while avoiding capture by four ghosts. Power pellets allow Pacman to eat ghosts temporarily. The goal is to clear the map without losing all lives.

## Innovations and Modifications:

- Two-player mode with individual scores.  
- Dynamic map regeneration every 30 seconds.  
- Ghosts use AI algorithms to pursue players intelligently.  
- New power-ups such as freezing ghosts temporarily.  
- If both players eat all pellets, the higher scorer wins.  
- A Pacman respawns after 5 seconds upon death.  
- Ghosts win if both Pacmans are eliminated; if one survives, the surviving player wins.

# 4. AI Approach and Methodology

## AI Techniques Used:

The project utilized search algorithms such as Genetic Algorithm, Min Max , Breadth-First Search (BFS), and A\* Search to control ghost behavior. Ghosts treat Pacman as the goal state and adapt their strategy based on current map structure.

## Algorithm and Heuristic Design:

Ghosts use evaluation functions based on heuristic distances (e.g., Manhattan distance) to prioritize paths. These functions help in optimizing their pursuit of the Pacmans even as the map dynamically changes.

## AI Performance Evaluation:

Ghosts were evaluated based on their responsiveness and efficiency. On average, the AI located and reached Pacman within 1.5(depends on the original speed set of the ghost)seconds after a map regeneration, demonstrating strong adaptability and real-time decision-making.

# 5. Game Mechanics and Rules

## Modified Game Rules:

- Dynamic map regeneration every 30 seconds.  
- Pacman repositioning after each regeneration.  
- Power-ups include ghost freezing.  
- Ghosts win if both Pac-men are eliminated.  
- If one Pac-men survives, they win.  
- The player with the highest score wins if all pellets are consumed.

## Turn-based Mechanics:

The game operates in real-time. Both Pac-men and ghosts move simultaneously, with ghosts recalculating paths frequently using AI logic.

## Winning Conditions:

- Both Pac-men eliminated → Ghosts win.  
- One Pac-men survives → Surviving player wins.  
- All pellets eaten → Higher scoring Pacman wins.

# 6. Implementation and Development

## Development Process:

The game was implemented using Python and Pygame. Custom classes handled map regeneration, AI logic, and player interaction. AI modules were designed to be modular and responsive to map changes.

## Programming Languages and Tools:

- Programming Language: Python  
- Libraries: Pygame, NumPy  
- Tools: GitHub (version control), Visual Studio Code

## Challenges Encountered:

Developing the ghost cage and release logic proved difficult, particularly ensuring consistent behavior when the gate opened. Implementing dynamic chasing logic using AI across changing maps and handling real-time maze generation were major technical hurdles.

# 7. Team Contributions

- [Laiba Bint-e-Zia]: Genetic Algorithm, front page  
- [Amna]: Min Max, dynamic map development.  
- [Alisha Zaidi]: A\*, integration and player mechanics. Score board  
- [Layyana Junaid]: BFS, game over page.

Testing, analysis, game rules, movements and the rest of the features were developed together

# 8. Results and Discussion

Ghost AI demonstrated the ability to adapt and pursue players efficiently, even as maps regenerated. On average, ghosts re-engaged with Pacmans in under 2 seconds, showing effective implementation of search algorithms.

# 9. References

**Sounds:**

<https://pixabay.com/sound-effects/search/pac-man/>

<https://classicgaming.cc/classics/pac-man/sounds>

**Sprites:**

https://www.spriters-resource.com/arcade/pacman/sheet/52631/