pacman.md

Pacman + Map Editor

Requirements: Windows or Linux Environment

Quick Install:

1. Download the latest **pacman.zip** release for Windows or Linux
2. Extract the files
3. Run pacman.exe
    - If permissions are denied, run `chmod 700 <path to pacman.exe>` and try running again

Build Yourself:

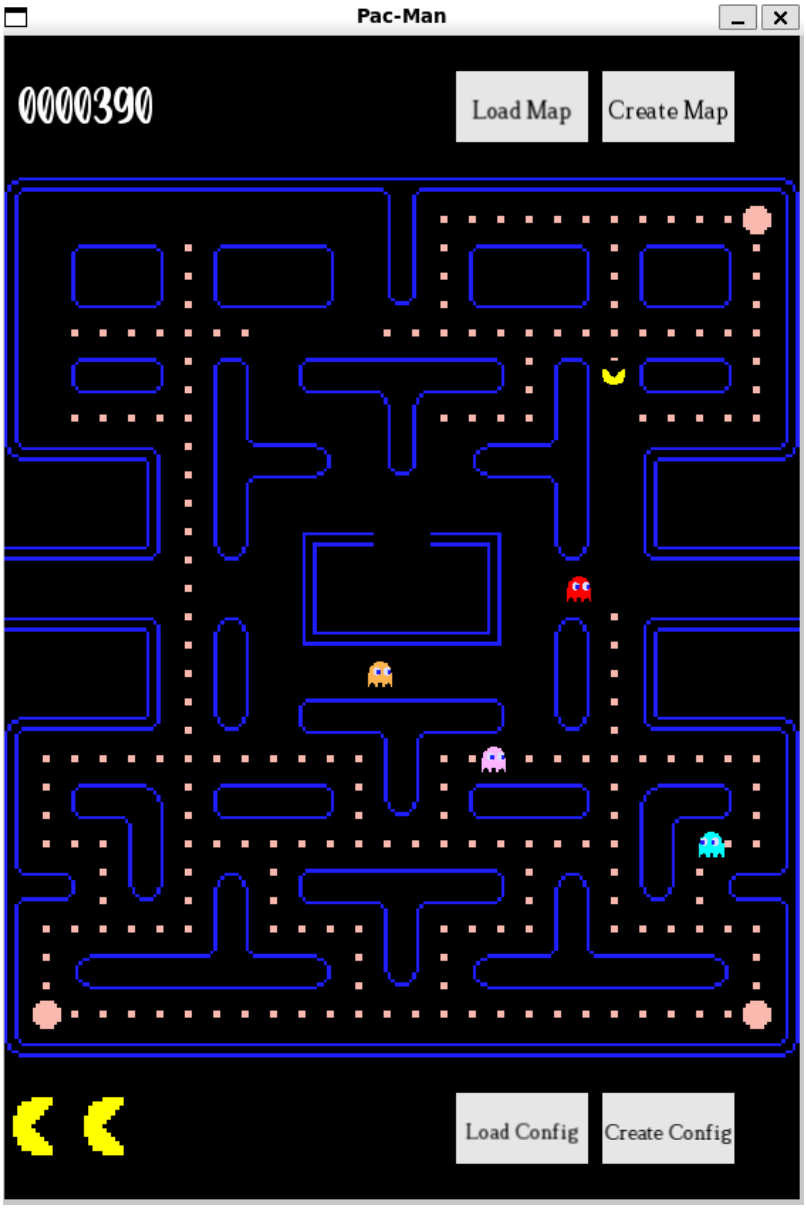
1. Clone the repo or download the source code in a Linux environment
2. Install the requirements: `sudo apt-get install libsfml-dev` and `sudo apt-get install build-essential`
3. Run the makefile in command shell or IDE terminal
    - Note: Neccesary changes to install location of SFML might be required in the makefile
    - Note: Reinstallation of **RapidJSON** in lib/ might be neccessary

Features

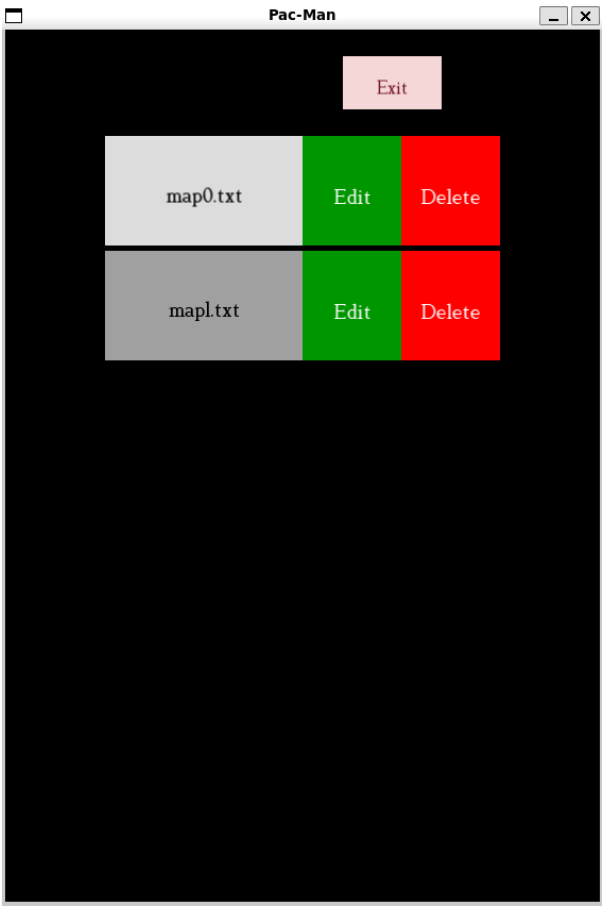
- This project features a playable Pacman game, as well as a fleshed out map editor + config editor to make the experience more customizable.
    - Each ghost has two modes **(scatter and chase)**. While scattering, the ghosts will target a designated target tile. While chasing, each ghost will implement its own unique **AI algorithm** to chase Pacman.
        - **Blinky:** Targets Pacman directly
        - **Pinky:** Targets 4 tiles ahead of Pacman's current direction

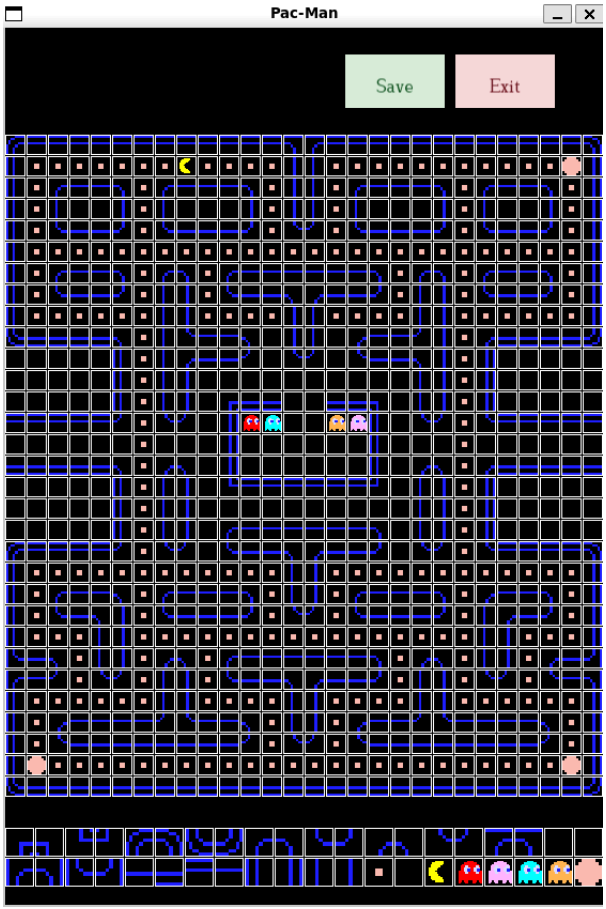
    - **Inky:** Targets Pacman directly until he is within an 8-tile radius, then switch back to scatter mode
    - **Clyde:** Targets the vector that is twice the distance between Blinky's position and Pacman's position

- Using WASD, the player can move Pacman throughout the map

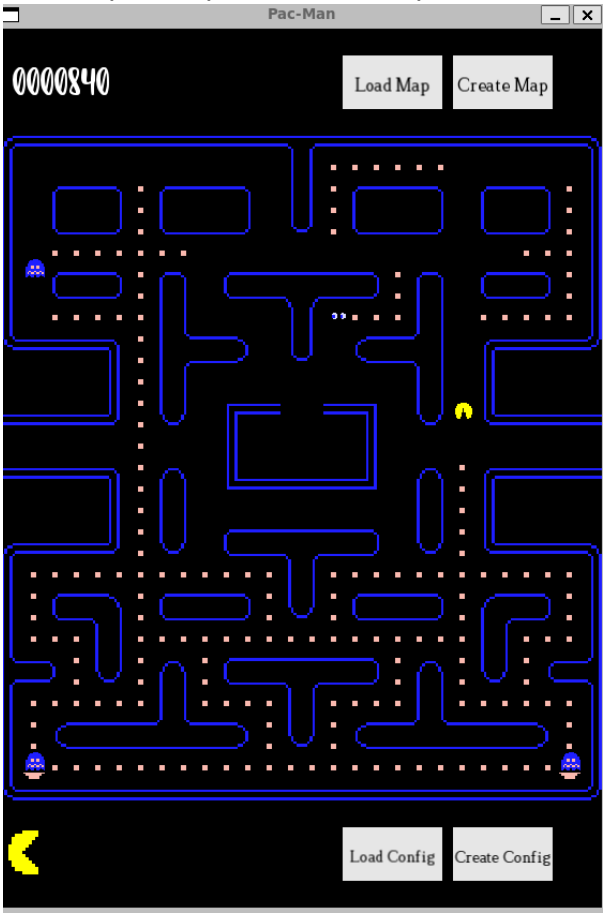


- One can create, edit, customize, and delete maps using the tile editor
    - Note: The max size of a map is 99x99.
    - Note: There are three tiles without a texture.
        a. Non-passable wall
        b. Ghost gate (non-passable for Pacman)
        c. Empty space (passable for all)





- Collect power pellets to earn points and eat ghosts!



- The config editor can be used to customize the difficulty and game settings
    - Note: The ghost's escape tiles are determined by the placement of the gates. When creating custom maps, it is necessary to assign each ghost to a gate. This can be a one-to-one or many-to-one relationship. The ghost's AI algorithm uses these gates to determine when they have entered or exited the den.