CMPT 276: Project

Phase 4: Report and Presentation
Group 24

Armin Hatami
Jacob Yee
Reid Lockhart
Ryan Marwaha

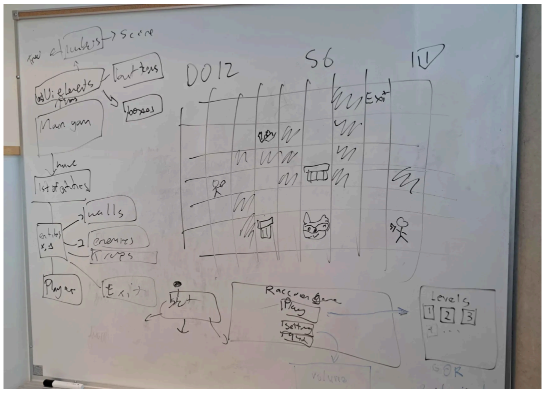
April 11th, 2025

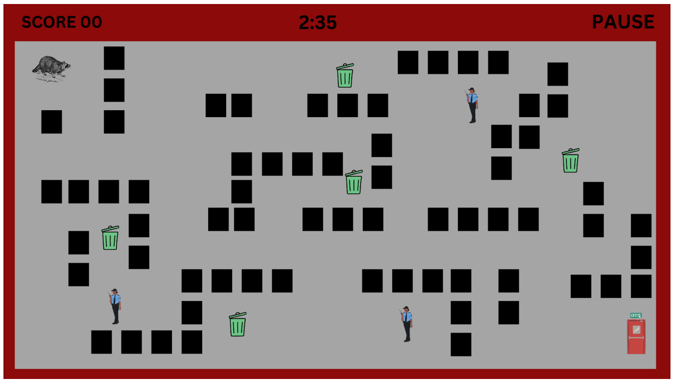
Game

2.1-The Game

The idea for 'Raccoon Escape' was inspired by the raccoons often seen near trash bins at SFU. The original concept was that the player (raccoon) needs to collect trash and avoid punishments/security guards, then reach an exit to beat a level (shown below). However, adjustments were made to this idea throughout the 3 phases of the project, which made the game more enjoyable to play.

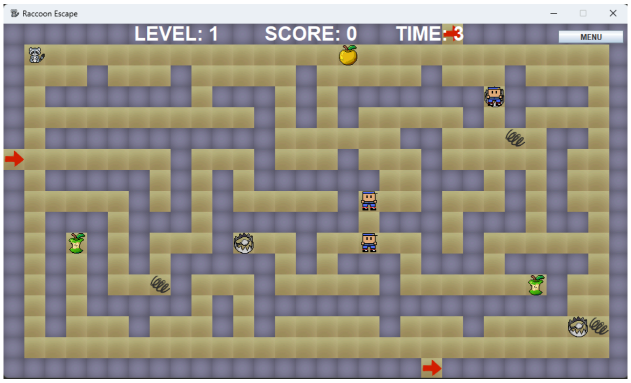
Early Drafts/Sketches



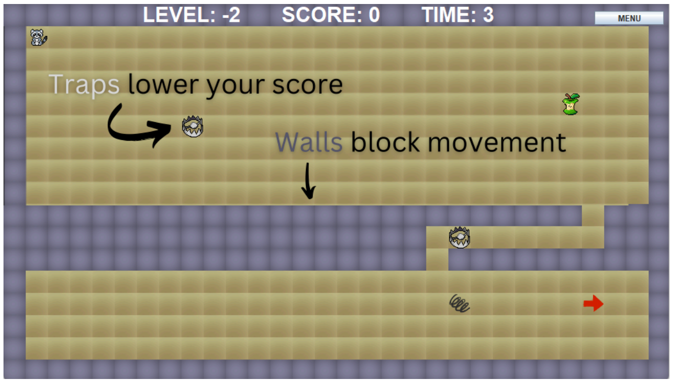


Final Design

Main Level:



Tutorial mode:



Changes to Original Plan and Design

Level Design - From Hard-Coded to Procedural Generation

Initially, we planned to design a set of hard-coded levels to be able to control the player experience and progression. However, we realized that creating engaging maps with various difficulties manually would be time-consuming and that none of us were particularly interested in focusing on level design. As a result, we decided early in development to implement a maze generator to procedurally create new levels each time the game is played. This change added a degree of replayability by ensuring the player was not limited to a finite number of static levels. However, this also introduced new challenges, such as implementing logic to ensure that all generated mazes were fully connected (without unreachable areas) and that entities were placed appropriately. Since this change was made near the beginning of the project, it was relatively easy to integrate. Still, it significantly increased the amount of testing required to ensure the generated levels functioned correctly.

Player Experience - Adding a Tutorial

To introduce the player to the game mechanics, we constructed a second game mode that consists of a series of fixed levels. This tutorial mode shows off the mechanics and gameplay rules sequentially so that the player can understand them all before starting the actual game and not be overwhelmed by receiving the information all at once.

Player Experience - Addition of Background Music

We added background music to enhance the overall user experience. This feature had a relatively small scope as it involved integrating a single looping track. Because it didn't interact with core gameplay systems, it required minimal testing and was straightforward to implement.

More entities added:

The following entities were introduced after the initial planning,

- Traps decrease the player's score

- springs set the player back to the corner

- rewards increase score

- bonus rewards have a time limit to collect

Classes and Structural Changes

There were a number of ways that our project differed from the original UML.

- No more wall, tile, and exit classes. This functionality was implemented through an integer array in the TileMap class.

- Reward and Punishment classes were merged into one. After all, both do the same thing except one adds and one subtracts points.
- Our original UML included an abstract factory design pattern, but this was scrapped because there was no need to implement it.
- All of our classes related to UI were handled by the Java Swing library instead of needing to create them ourselves.

The structural changes were done to make the game simpler to implement and update, as well as saving us from designing our own libraries when there are already great ones available.

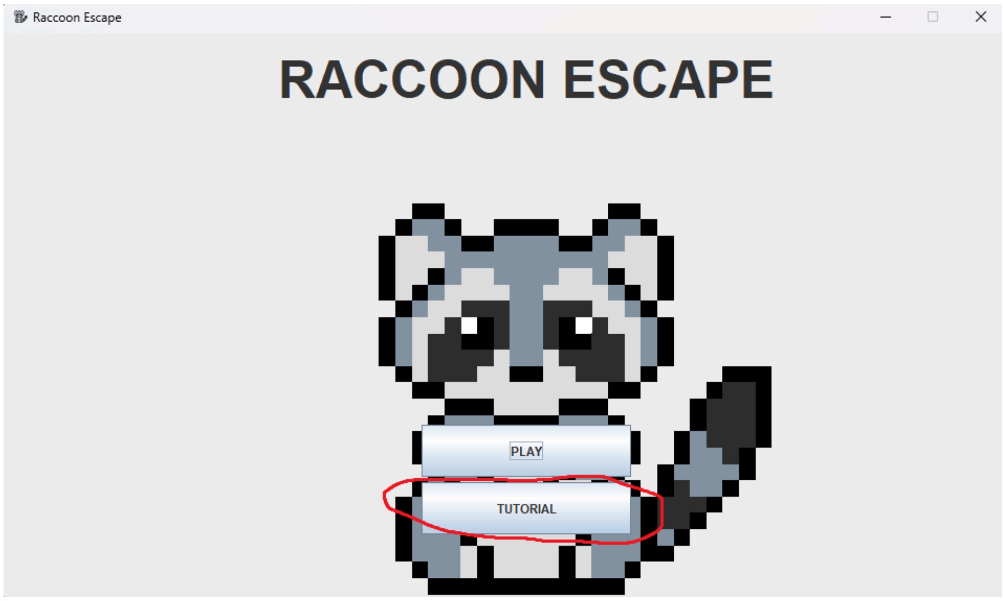
Lessons Learned:

Several lessons were learnt during the coding process, and the testing phase. For example, a problem was the file paths did not work across all operating systems, this made images disappear on some devices while appearing properly on others. Solving this issue highlighted how important it is to check that the software works across all intended devices. Furthermore, as more features and game mechanics were added, the code began becoming more disorganized: some classes were too long, and there was a lot of unnecessary code, and room for refactoring. This showed that it's always necessary to continuously refactor code so it's easier to modify when programming new features. Finally, the testing phase demonstrated how some code bugs can only be caught through extensive testing, which often reveals problems at edge cases.

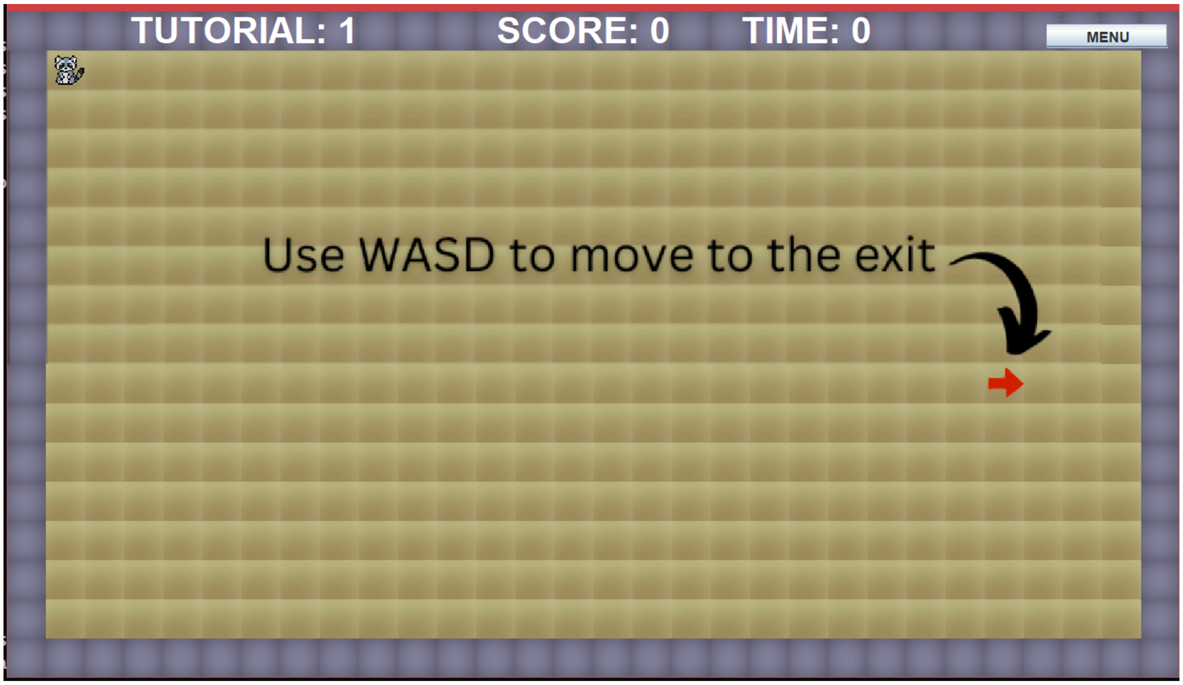
Tutorial

Since our game's tutorial already does the job of walking players through how to play the game, here is a walkthrough of what the tutorial covers.

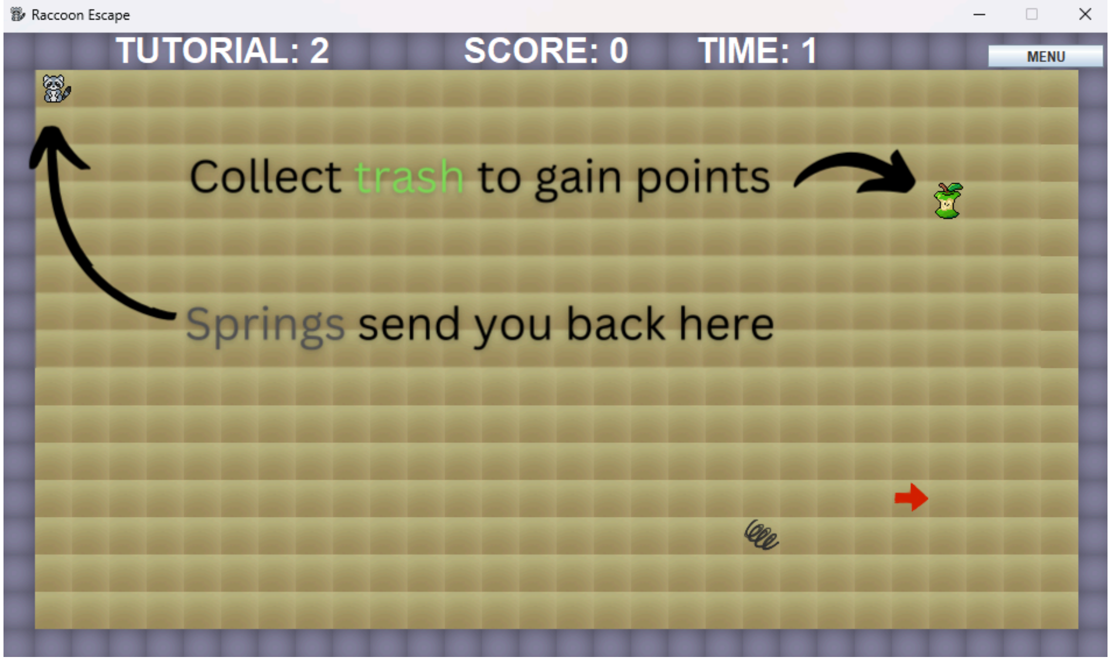
Once the game has been built and run according to the instructions in our README file, click on the tutorial button to begin it.



The first level of the tutorial is very simple. To progress in our game, the player needs to make it to the exit. To move, use the WASD keys to move the raccoon onto the exit tile.



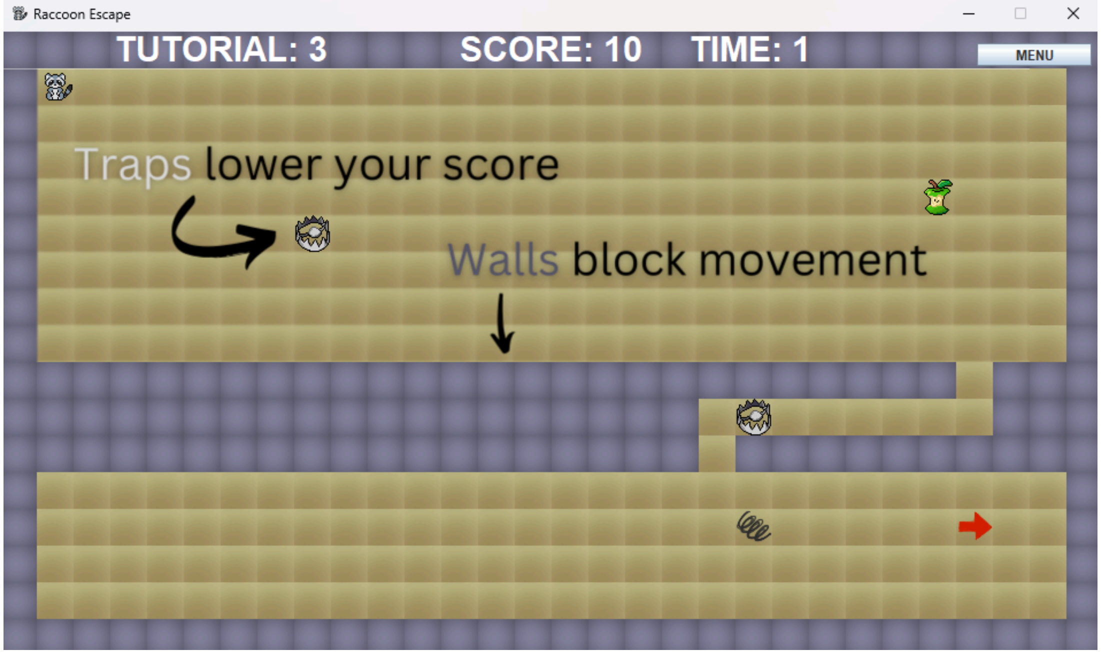
To score points, the raccoon needs to collect the garbage, which are half-eaten apples. Once the raccoon moves onto one of these, it disappears and rewards 10 points. In addition, the game features springs. These send the player back to the beginning tile, which is located in the top left of the map. Springs can be good or bad depending on the situation. For example, they could help to escape the guards if used at the right time.



Collect trash to gain points

Springs send you back here

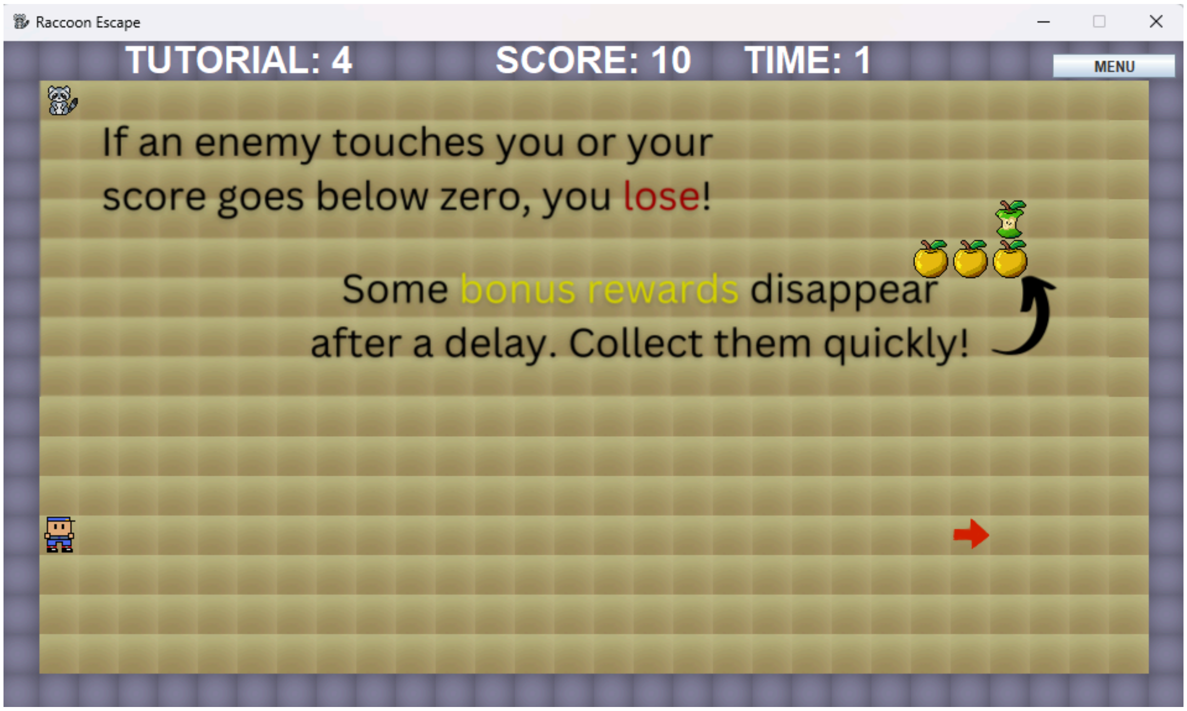
The purple tiles on the game board represent walls, which are impassable. In addition, the spiked traps remove 10 points if stepped on. If the player's score goes below zero from this, they lose!



Traps lower your score

Walls block movement

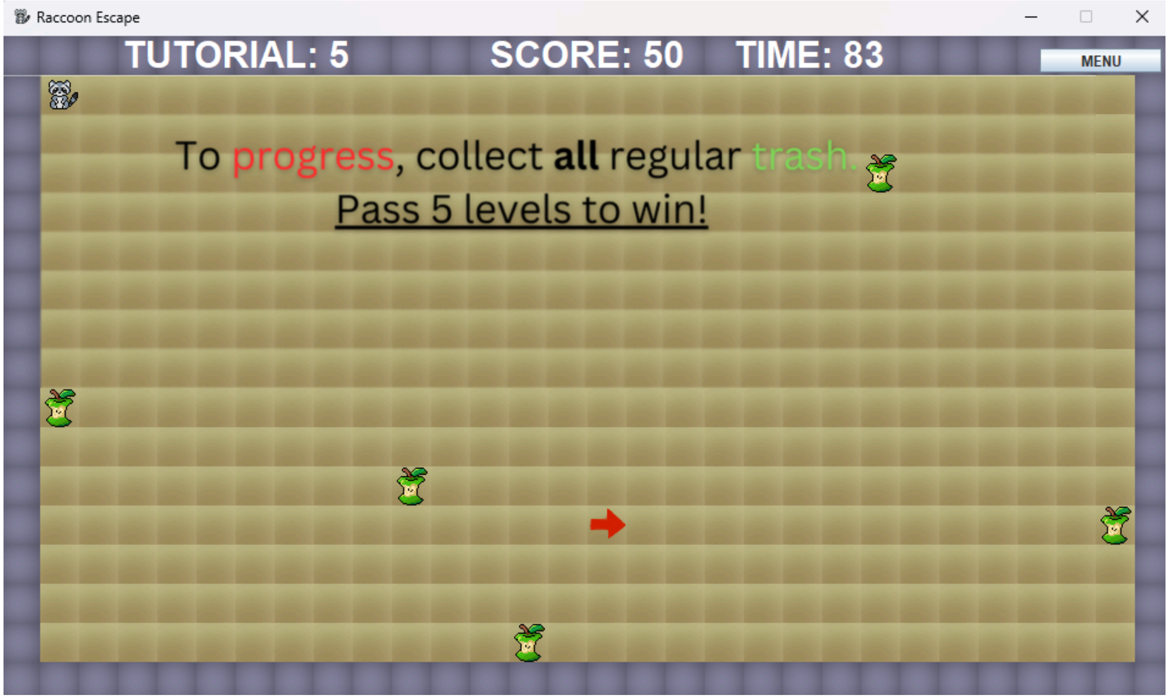
The moving enemies in the game are very dangerous. If they contact the player, it is an instant game-over! Avoid them using the arrow keys. Golden apples that sometimes appear in levels have a delay of 5 seconds before disappearing for good. They must be collected quickly to increase your score.



If an enemy touches you or your score goes below zero, you lose!

Some bonus rewards disappear after a delay. Collect them quickly!

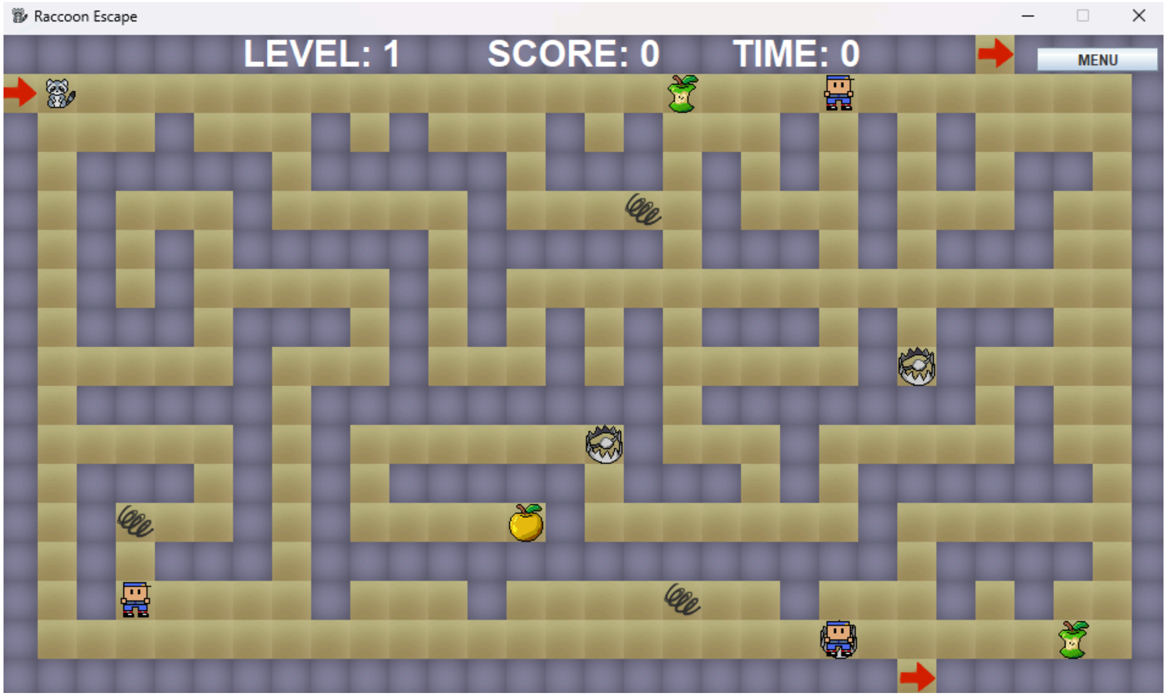
Once the actual game begins, the player will need to collect all of the regular trash from around the level in order to enter the exit and progress.



To progress, collect **all** regular trash.

Pass 5 levels to win!

Now that the tutorial is finished, the player enters the first level of the real game. This can also be done by pressing Play from the start menu. At any time, the player may click the menu button located in the top right of the map to return to the main menu, resetting the current game. To win, the player must navigate through 5 different levels, collecting all the trash to reach the exit while avoiding the traps and guards. Good luck!


Raccoon Escape
LEVEL: 1
SCORE: 0
TIME: 0
MENU