

Academia Manual

Academia Team

2023-05-11

Table of Contents

- 1. General Game Overview 2
- 2. Game Play Details for Core 1-Player Version 3
 - 2.1. Objective 3
 - 2.2. Rules 3
 - 2.3. Start/Title Screen 4
 - 2.4. 1-Player Level Start State 7
 - 2.5. Player Scoring 7
 - 2.6. Player Death 7
 - 2.7. Infinite Scroller 8
 - 2.8. Player/Hazard Movement and Collisions 8
 - 2.9. Game Over Screen 9
 - 2.10. Objects 11
 - 2.11. Asynchronous (Input) Events 25
 - 2.12. Synchronous (Timed) Events 27
 - 2.13. Condition-Based (Cascaded) Events 28
 - 2.14. 1-Player Hypothetical Gaming Session 31
- 3. Game Play Details for Core 2-Player Version 34
 - 3.1. 2-Player Hypothetical Gaming Session 35
- 4. Sound Effects 37

Copyright 2023 Academia Team

All portions of code in this spec file are hereby licensed under the GPL-2.0-only license.

The rest of the text is licensed under the CC-BY-SA-4.0 license.

All images, if unspecified, have been created by the Academia Team and are under the CC0 License.
The origins of all other images are specified in [CREDIT.txt](#).

Chapter 1. General Game Overview

Academia: Dreams of Madness is a top down 2D Frogger like game where the player can move infinitely forwards through the world. This world constitutes a 11 by 19 grid full of rows of the following types: grass, road, or track. The goal is to achieve the highest score that you can, which can be monitored by staring at the top-left corner of the screen. Points can be accumulated by advancing onto previously unvisited rows and collecting the letter grades A, B, and C (via standing on them). But, be warned, as you are a mere student with only three lives and enemies will spawn above you to oppose any progress made. Enemies include cars, trains, and Mr. Feathers (an avian foe). They can only move horizontally left or right, guarding the rows on which they lie. Luckily, the player can move not only horizontally left and right, but vertically up and down as well, which helps you outmaneuver them. Unfortunately, nothing can save you—you will fail to make it to class. There exists spike cells to stab you and hedge cells to block your path which occur on grass rows—look ahead and plan wisely. Gaze in horror at the top-right corner of the screen as your number of lives slowly dwindles. Survive as long as you can, for as soon as you lose all of your lives, it will be the end of the road for you.

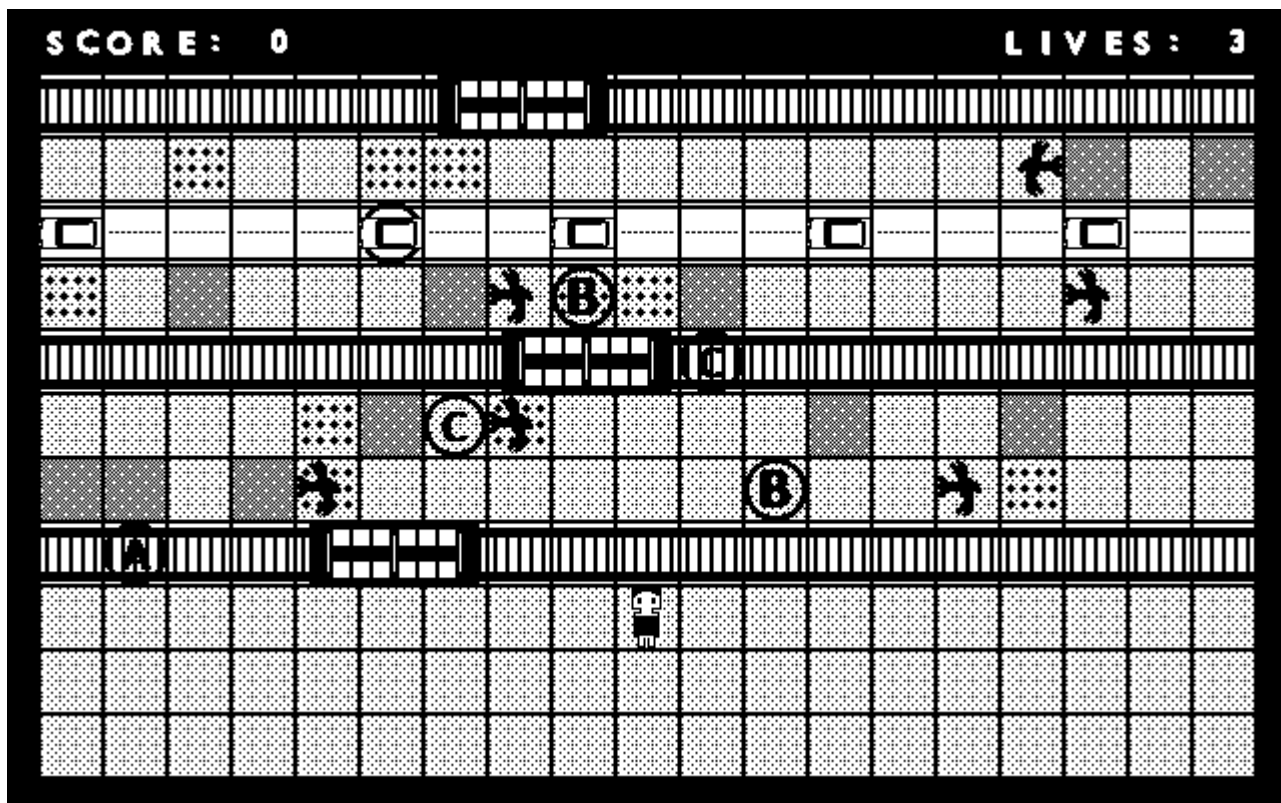


Figure 1. An example of what the player might see—complete with all of enemies and cells.

Chapter 2. Game Play Details for Core 1-Player Version

2.1. Objective

Your task is simple: Stay Alive! Move forward towards your academic dreams while collecting bonuses along the way.

2.2. Rules

1. World consists of 11 rows in 1-Player mode and 10 rows in 2-Player mode.
 - a. Rows consist of 19 cells.
 - i. Row randomly determines the type of cells generated; all have equal chance.
 - ii. Rows determines the hazard(s) type based on cell type and limits the number of hazards generated. Cars will spawn on roads, trains will spawn on tracks, and Mr. Feathers will spawn on grass.
 - A. Limit of 5 cars on road row, 2 trains on track rows, and 2 Mr. Feathers on grass rows.
 - B. Cars have a $1/5$ chance of spawning, trains have a $1/10$ chance of spawning, and Mr. Feathers have a $1/10$ chance of spawning.
 - iii. Both cars and Mr. Feathers can only span one cell at any given time while trains span 3 cells (except when moving onto and off of the screen, when it would occupy fewer cells).
 - iv. Row determines generation of static (non-moving) spikes and hedges if and only if the row's cell type is grass. Limits the number of spikes and hedges.
 - A. Limit of 3 Hedges and 3 Spikes.
 - B. Both Hedges and Spikes have a $1/6$ chance of spawning.
 - v. Rows determine the direction all moving hazards will travel: either horizontally left or right.
 - vi. Row may generate collectables if the limits on the number of collectables haven't been reached.
 - vii. A collectable has a $1/40$ chance of spawning, B collectable has a $1/30$ chance of spawning, and C collectable has a $1/20$ chance of spawning.
 - A. Collectables are not able to be spawned on hedge cells.
 - viii. New hazards are able to spawn in pre-existing rows every second.
 2. The player can travel horizontally left and right, as well as vertically up and down.
 - a. The hedges and the boundary around the screen will block player movement.
 - b. The player can move at a rate of one cell every half second.

3. Enemies travel left or right horizontally and always move before the player.
 - a. They travel onto, across, and off the screen, in the row's set direction one cell at a time.
 - b. Cars move one cell every $\frac{4}{5}$ of a second, trains move one cell every $\frac{1}{5}$ of a second, and Mr. Feathers move one cell every $\frac{3}{5}$ of a second.
 - c. When an enemy reaches the end of a row they are removed from the row and are no longer counted towards their row's limit on hazards.
4. Avoid Hazards on the way to school.
 - a. The player can move onto an enemy and vice versa, which will result in a collision.
 - i. Collision with a hazard will result in the player losing a life (Car, Train, Mr. Feathers and Spikes).
 - ii. Player remains in place and is granted two seconds of immunity immediately to hazards after a collision.
 - A. During this period of immunity, the player can not lose any more lives.
 - B. If a collision with a hazard occurs during this period, the immunity timer will not be reset.
 - iii. A Player having no lives remaining will result in death and the displaying of the Game Over screen.
5. Each new row visited nets the player 100 points.
6. Attempt to collect letter grades. They may be worth the risk.
 - a. Collectable Grade 'A' - Grants the player a score of 500. Limit 1 per gameplay area.
 - b. Collectable Grade 'B' - Grants the player a score of 200. Limit 2 per gameplay area.
 - c. Collectable Grade 'C' - Grants the player a score of 100. Limit 3 per gameplay area.

2.3. Start/Title Screen

The game's start screen consists of the game's logo and the following three buttons: *1-Player*, *2-Player*, and a *Flee* (quit) button. The buttons can be clicked on using a mouse. Keyboard input is also accepted on that screen. Pressing the tab key will cycle between selecting the *1-Player*, *2-Player*, and *Flee* buttons. Also, moving the mouse cursor to a button will select that button. Hitting the enter, space, or return key, as well as clicking the mouse while over a button, will activate the currently selected button. By default, no button is selected. When all buttons have been cycled through, no button will be selected. Hitting tab once more will select the first button.

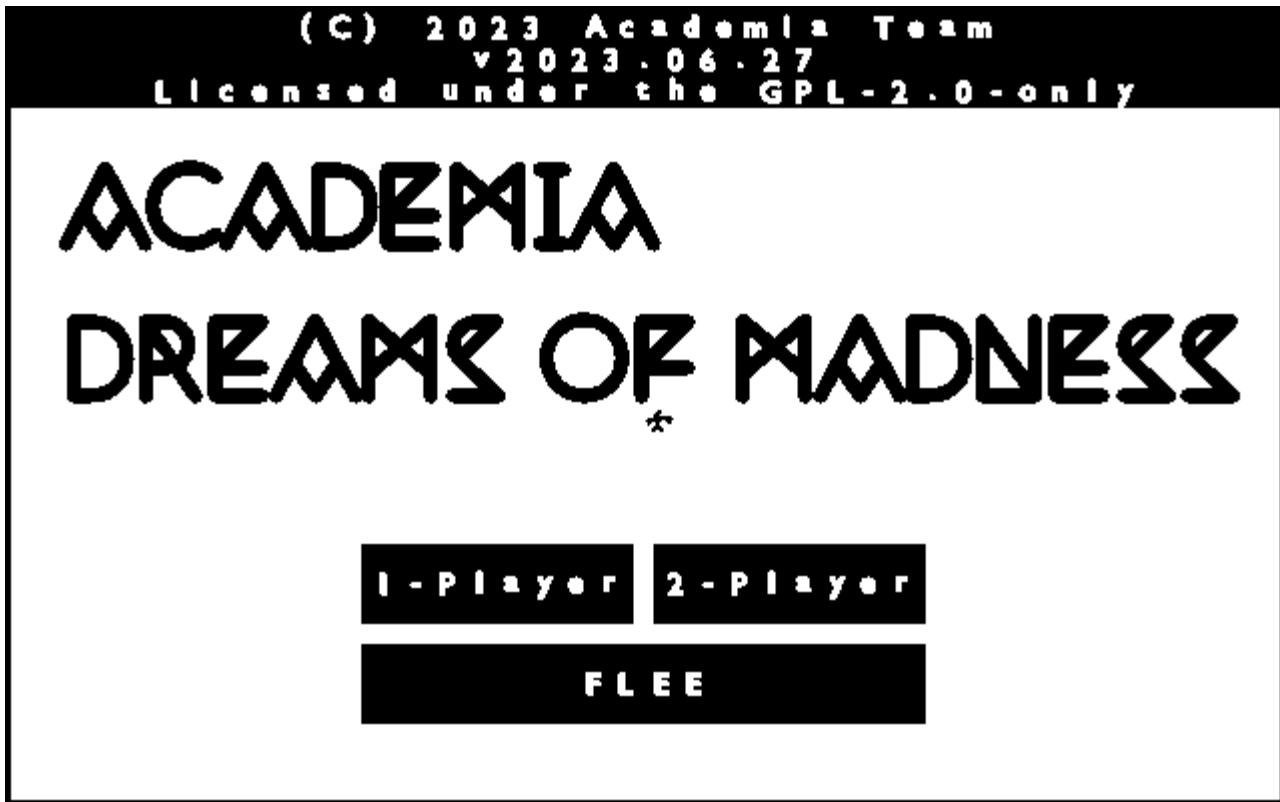


Figure 2. The Title Screen with no buttons selected.

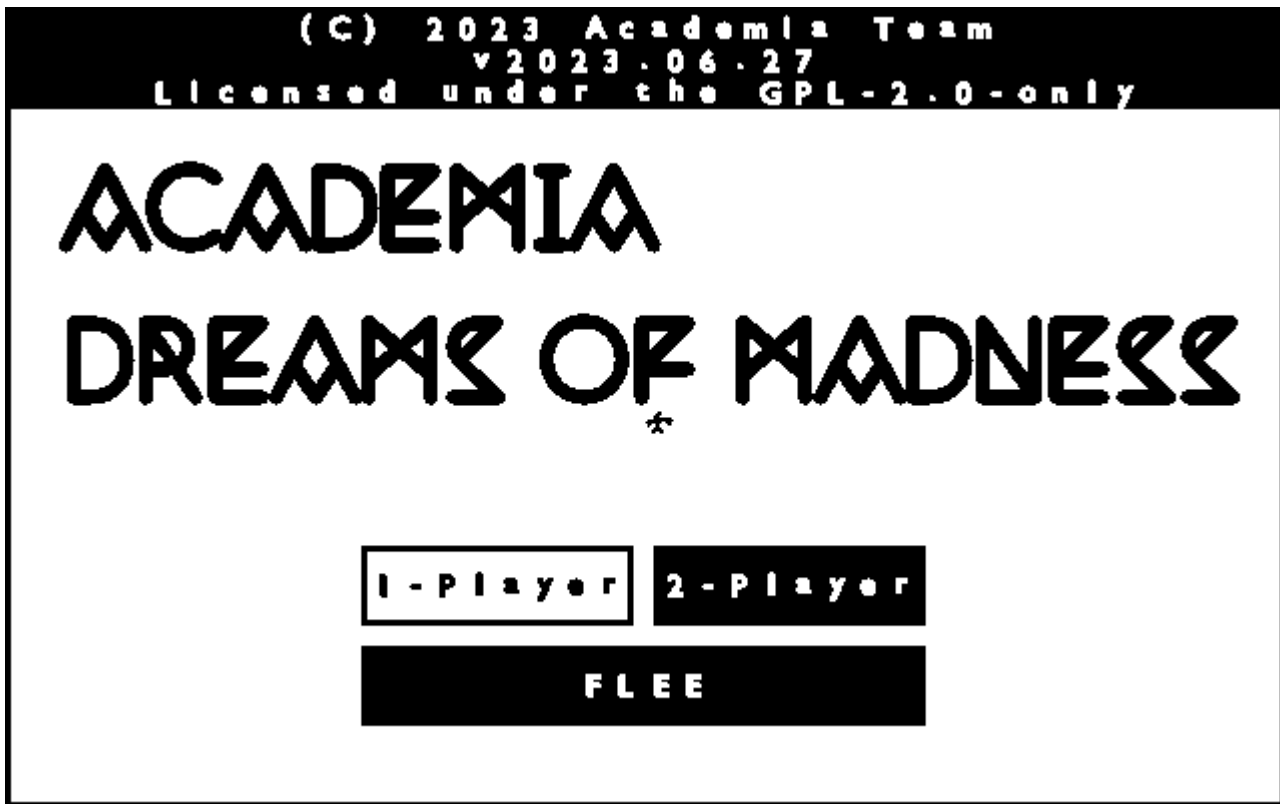


Figure 3. Title Screen with 1-Player button selected.

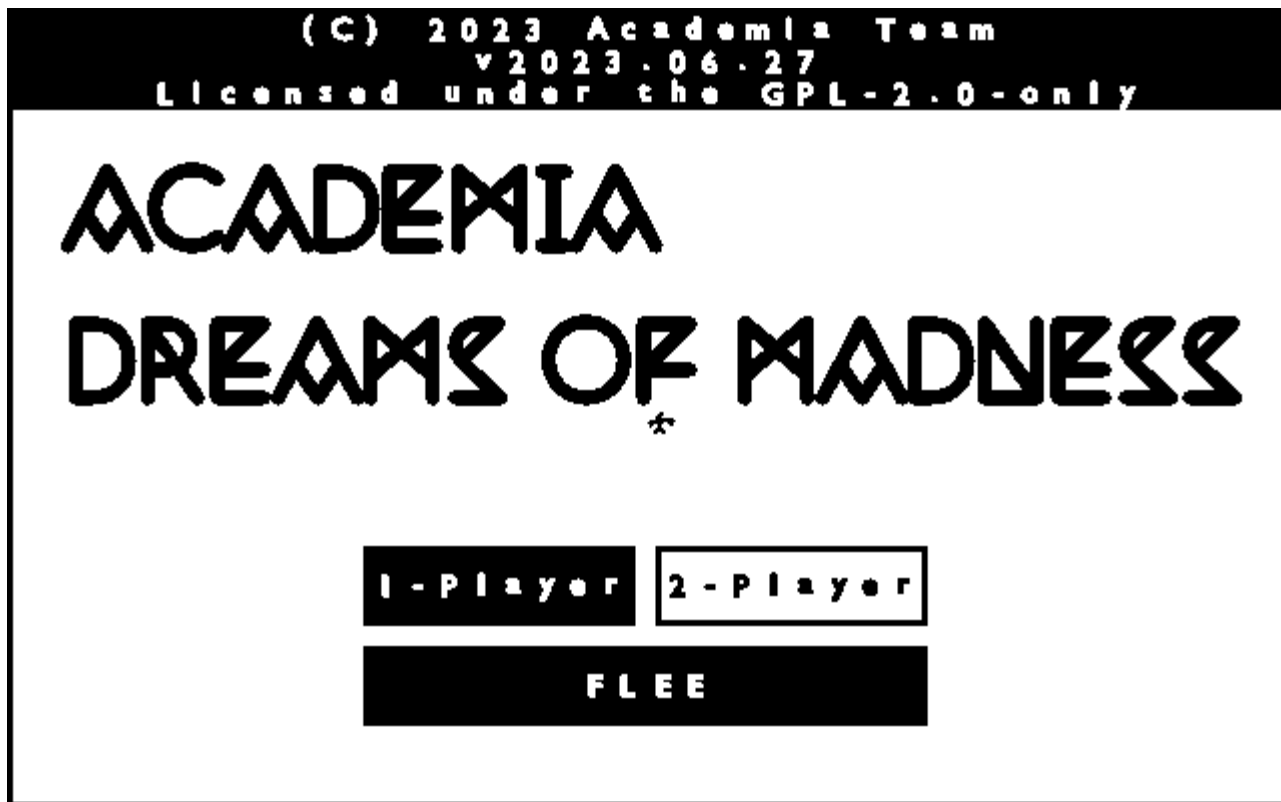


Figure 4. Title Screen with 2-Player button selected.

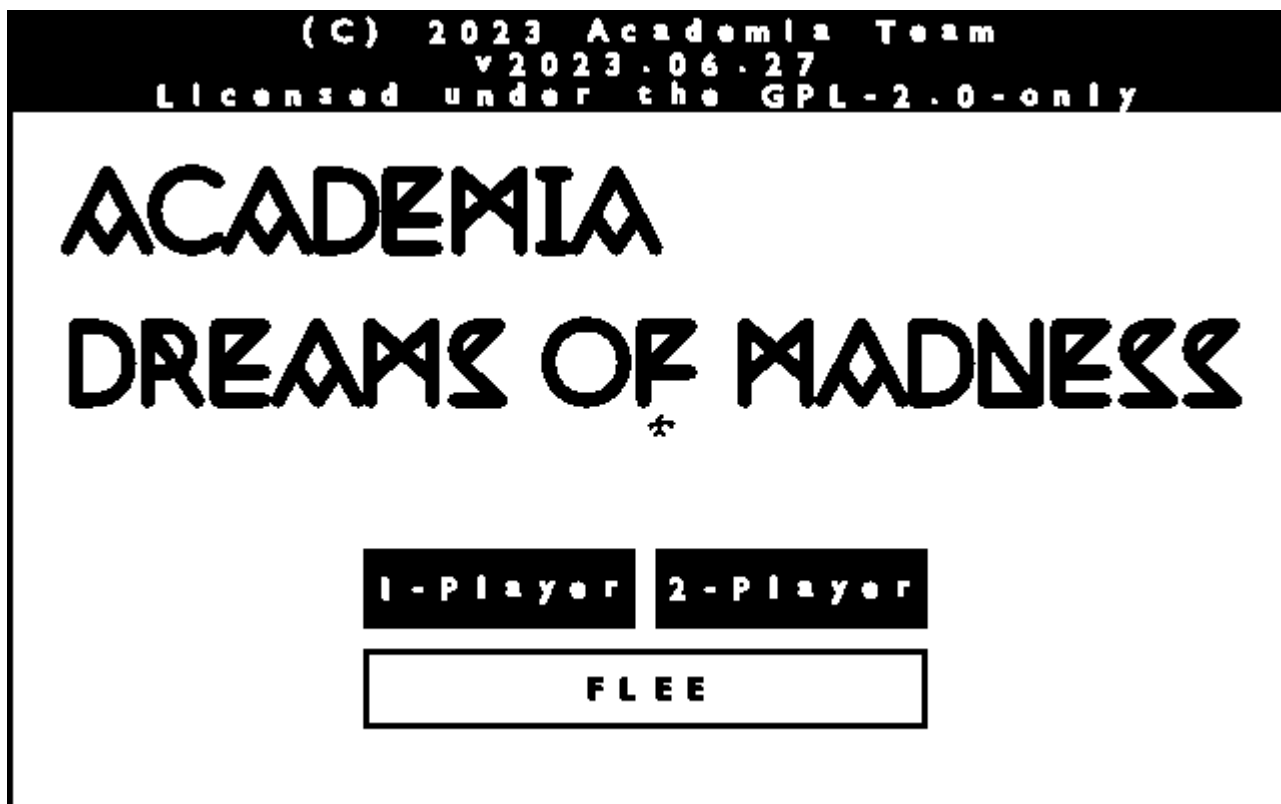


Figure 5. Title Screen with Flee button selected.

2.4. 1-Player Level Start State

The player will spawn on the third row (from the bottom of the gameplay area) and the tenth column. The play area is 11 rows by 19 columns. Spawn is not the third row itself but rather the cells on which the player starts and moves away from. The first bottom three rows, at the very start of a game, will always consist of grass cells with no hazards. Hazards, collectables and safe cells will be generated in all above/forward rows in the world. The player will be able to see their current score and number of lives (starting at zero points and three lives respectively) at the top corners of the screen.

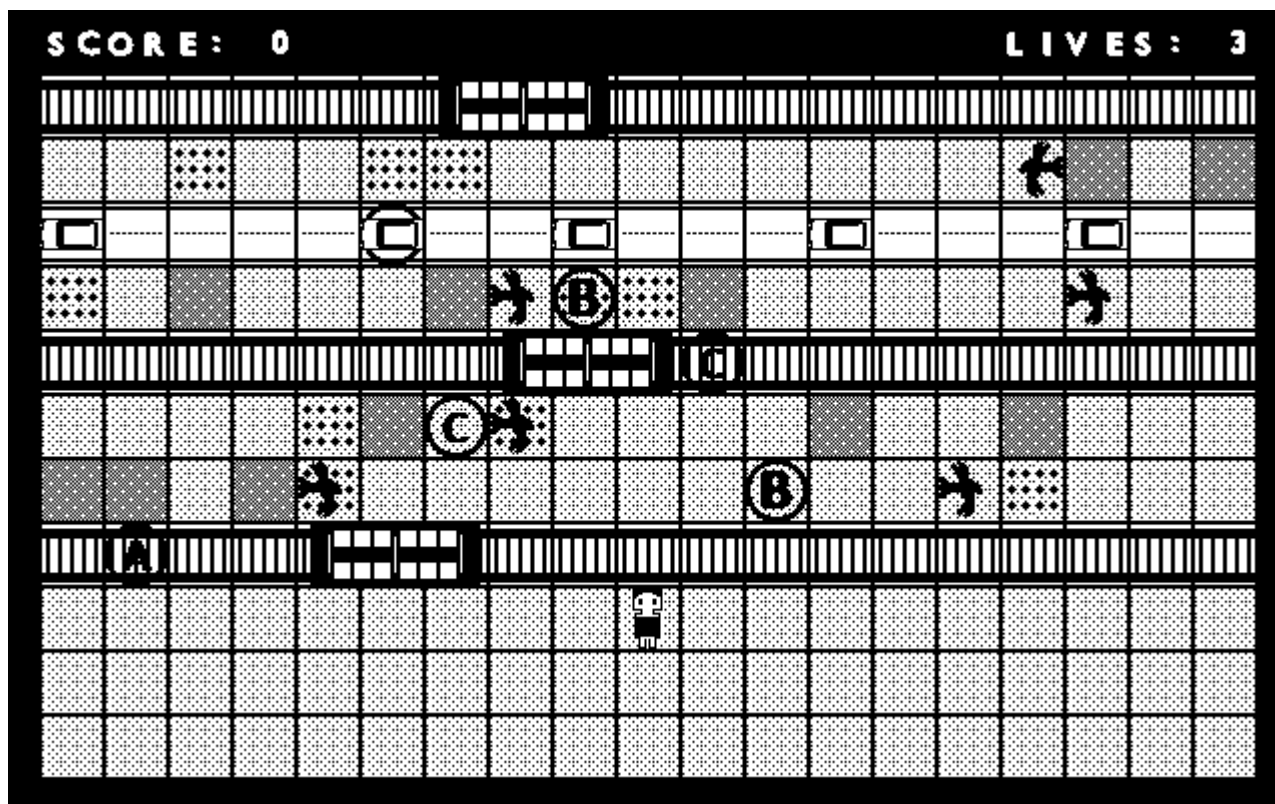


Figure 6. An example of what the player might see in 1-player mode.

2.5. Player Scoring

The player is granted 100 points for moving forward through the level to new rows that have not yet been visited. Points can also be gained by obtaining collectables. A letter grade of 'A' is worth 500 points, 'B' is worth 200 points, and 'C' is worth 100 points.

2.6. Player Death

The losing condition is when the player has lost all three lives they started with. Winning is not possible in the game. Lives will be lost by colliding with a hazard. Collisions occur when the player is standing on the same cell as a hazardous object. Hazards include cars, trains, spike cells, and Mr. Feathers.

2.7. Infinite Scroller

The player can move all the way to the left or right, but if they reach the edge of the screen, the border will prevent further movement. The player can move to the second last or last row without causing the world to shift. When the player creates a move forwards request which would result in them moving onto the fourth row from the bottom in the gameplay area, the player sprite will not move on screen whatsoever. Rather, the world immediately shifts downwards around them. When all is said and done, the player will still visually be in the third row of the gameplay area, but the row the player is on will be incremented in the internal world model. This allows the player to see forward through the world in order to plan their next steps. The row which visually gets shifted off the screen will have its cells, collectables, and hazards regenerated. This will become the new top row in the world ^[1].

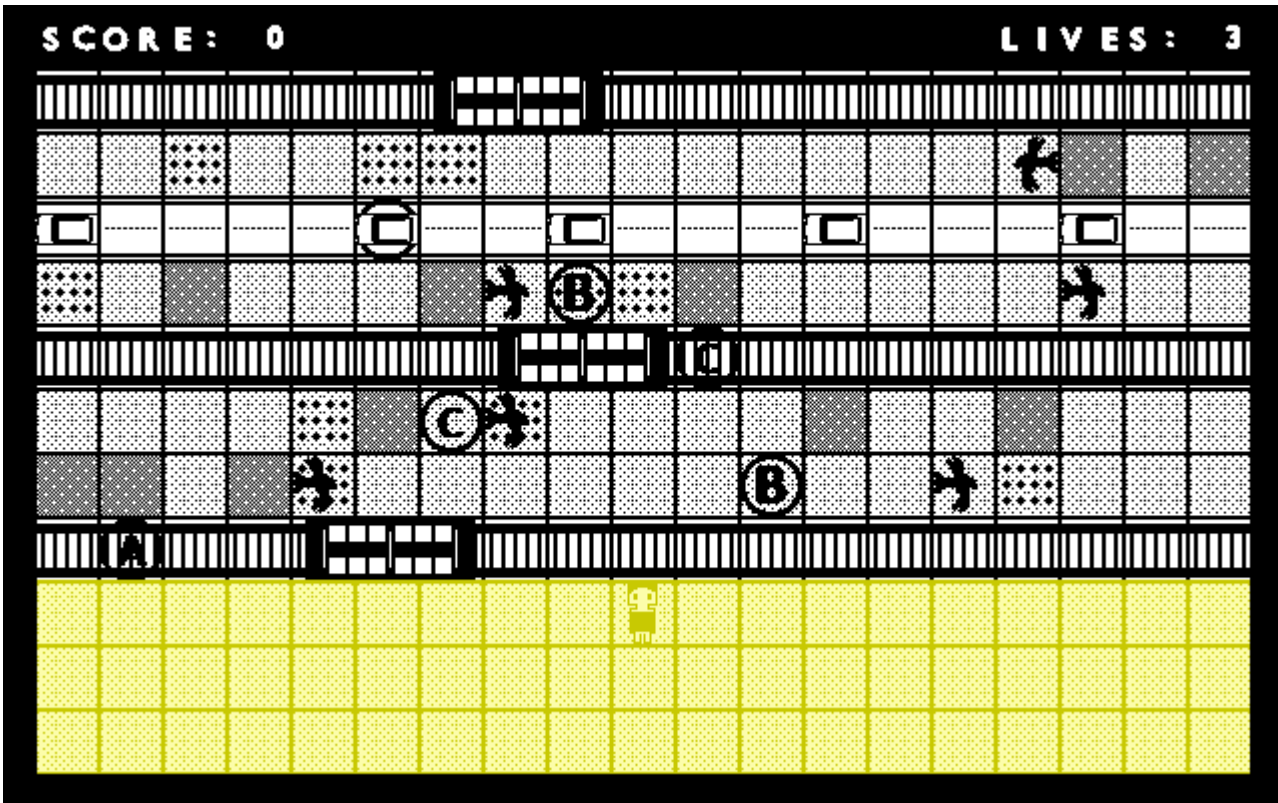


Figure 7. The highlighted cells in the image show where the player can stand without a visual shift occurring.

2.8. Player/Hazard Movement and Collisions

The player can only move straight up, down, left, or right. They cannot move past the left, right, or bottom of the screen, as denoted by the black border, which is 16 pixels wide. Hazard movement is more restricted, as they can only move horizontally left or right across the screen at a constant rate. All hazards on a given row will travel in the same direction. They emerge from the border surrounding the screen. Then, they continue advancing in their chosen direction until they disappear into the border on the other side of the screen.

A hedge cell serves to obstruct player movement, as they cannot stand on the cell. Hedges will not affect the movement of any hazards. Hedges are static in the world and exist only in grassy rows.

Moving hazards are randomly generated onto a row based on the cell type that the row is made up of. A row made up of track cells will spawn trains. Similarly, a row made up of road cells will spawn cars. However, static hazards, such as spike cells, will spawn randomly only during row creation. This gives grass cells two distinct hazards: spikes and an avian enemy named Mr. Feathers. If the player is on the same cell as a hazard, they will lose one life. Upon collision with a hazard, the player will stay where they are and immediately receive immunity from any further loss of lives for only two seconds. Any further collisions during this time will not reset the immunity timer. Collisions with players will not affect hazards in any way; non-static hazards will keep on moving as if nothing has happened.

2.9. Game Over Screen

After a single player game, this screen will display a game over message along with the player's achieved score and two buttons. The *RETREAT* button will send the player back to the title screen, while the *PLAY AGAIN* button will allow the player to start a new single player game. The buttons can be clicked on using a mouse. Keyboard input is also accepted on that screen. Pressing the tab key will cycle between the *PLAY AGAIN* and *RETREAT* buttons. Hitting the enter, space, or return key, as well as clicking the mouse while over a button, will activate the currently selected button. By default, no button is selected. When all buttons have been cycled through, no button will be selected. Hitting tab once more will select the first button.



Figure 8. Game Over Screen in 1-player mode with no buttons selected.



Figure 9. Game Over Screen in 1-player mode with Play Again button selected.



Figure 10. Game Over Screen in 1-player mode with Retreat selected.

When a two player game has been played, the Game Over screen will contain the same message, player score, and buttons from the single player Game Over screen. However, it will also contain the winner

of the game, and the other player's score. In the case of a tie in score, the same statistics will be displayed, but no winner will be declared. Instead, on each other's screen, they will see "TIE". As well, if one player dies while another is still playing, they will have a game over screen displayed for just them. On this screen, the statistics involving the other player will get updated until a winner or tie is declared. Also, both players will have to click *PLAY AGAIN* in order for a new two player game to start. Unless one of the players hits *RETREAT*, no matter how much time has passed since one player has clicked the *PLAY AGAIN* button, a new game will be started as soon as the other player also clicks the button.

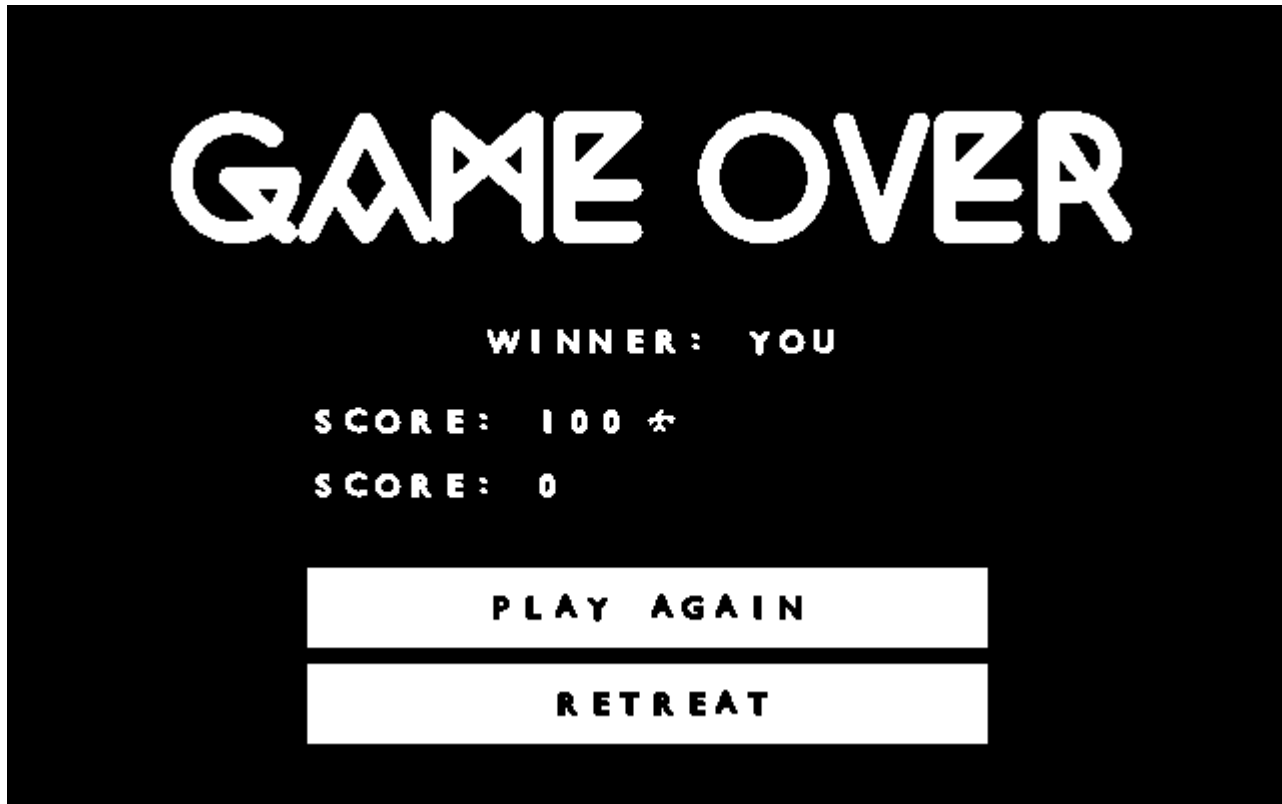


Figure 11. Game Over Screen in 2-player mode with no buttons selected.

2.10. Objects

Name	Properties	Behaviours	Image(s)
Screen	• Dimensions: 640x400 pixels. • 1-Player: 11 rows at 32 pixels tall (play area). 32 pixels on top for score and lives. • 2-Player: 10 rows at 32 pixels tall (play area). 64 pixels on top for score and lives for both players. • 16 pixel border on left, right, and bottom of the screen. • 19 columns at 32 pixels wide. • Score box(es). • Health box(es). • Grid of environment cells. • Shows hazards.	• Displays all objects.	• 1-Player Screen  • 2-Player Screen 

Name	Properties	Behaviours	Image(s)
World (Row Array)	• Location on screen (16,32)px. • 1-Player: 11 rows • 2-Player: 10 rows • Technically not visible to the user. (Its children contain visible elements.) • Circular array of rows, vertically wrapping. i.e top and bottom rows connected together.	• Manages the vertical wrapping of rows. • Manages the collectables count. • Manages all players (including the number of players).	

Name	Properties	Behaviours	Image(s)
Row	• 1-Player: 19 columns. • 2-Player: 19 columns. • Contains an array of cells. • Contains the direction of non-static (car, train, Mr. Feathers) hazard movement on row (left or right). • Contains a count of Hazards in row. • Contains a count of hedges in row. • Contains a count of spikes in row. • Not visible to the user. • Contains its vertical pixel position on screen. Stores hazard (if one currently exists in the row).	• Determines the type of cells in a row. (Grass, Road, or track) • Determines the direction of travel for all moving hazards. • Updates hazard count. • Updates hedge and spike count if and only if cell type is grass. • Updates its vertical position.	

Name	Properties	Behaviours	Image(s)
Cell	• Visible to the user. Actual image shown is defined by the cell type and any hazards and collectables in the cell. • Has a cell type of Grass, Spike, Hedge, Road, or Track. • Contains collectable (if one currently exists in cell). • Contains its horizontal pixel position on screen.	• Toggles if a hazard is in the cell or not. • Toggles if a player is in the cell or not. • Updates its vertical position.	

Name	Properties	Behaviours	Image(s)
Mouse Cursor	• Discrete movement. • Can remain unmoving for an infinite amount of time. • Can move up, down, left, right, top-left, top-right, down-left, and down-right at a rate that's upper limit is how fast mouse input can be handled. • 32x16px on screen. • Max 1 cursor on screen.	• Indicates the current relative mouse position on screen. • Changes its color to better contrast between black and white areas of the screen.	• Cursor on white background.  • Cursor on black background. 
Button	• Integer pixel location. • Has a label. • Is rectangular. • Contains states selected and unselected. • Is visible, but actual content and size varies.	• Can be selected. • Can be activated to invoke a specific action. • Color of the button varies depending on its background.	•  •  •  •  • 

Name	Properties	Behaviours	Image(s)
Info Bar	• y integer pixel location. • Holds labels. • Is visible. • Takes up the entire length of the screen.	• Can add a label. • Can remove a label. • Horizontally centers all labels it manages on screen.	• 
Score Box	• Static integer location 1-player: (16,8)px, 2-Player: (128,8)px score box location for whoever is playing on the computer and (128,39)px for the other player's score box. • The font utilized is 16 by 16 pixels. • Max score is Long Word. (unsigned integer bit limit) • Has a label containing the text "SCORE:".	• Displays Score. • Updates Score.	• 

Name	Properties	Behaviours	Image(s)
Health Box	- Static Integer location 1-player: (495,8)px, 2-Player: (495,8)px score box location for whoever is playing on the computer and (495,39)px for the other player's score box. - The font utilized is 16 by 16 pixels. - Integer health. - Maximum: 3 lives. - Minimum: 1 life. - Has a label containing the text "LIVES:".	- Displays Health. - Updates Health.	LIVES: 3
Label	- Integer pixel location. - Stores text. - Is visible, but actual content varies. - Font utilized is 16 by 16 pixels.	Displays label text.	

Name	Properties	Behaviours	Image(s)
Player Character (Student)	• Has a position on screen (which starts at (304,288) pixel coordinates). • Discrete movement. • Move direction up, down, left, right and none at a rate of 32/70 px/tick by 1 cell. • Is alive. • Has 3 lives. • Has score. • Has orientation: South (out of the screen, facing the player), North (into the screen, with back facing the player), East (facing to the right), and West (facing to the left). • 32x32px on screen. • Max 1 player on screen. (Users only see their own player).	• Moves when unobstructed and a relevant key is being pressed (up, down, left, and right arrow keys). • Dies at 0 lives.	•  •  •  • 

Name	Properties	Behaviours	Image(s)
Other Player	• Not visible to the user. • Has 3 lives. • Has a score. • Is alive.	• Dies at 0 lives.	
Car	• Continuous movement. • Location and direction determined by row and cell. Moves horizontally left and right at a rate of 32/56 px/tick. Has orientation: West (facing to the left) and East (facing to the right). • 32x32px on screen. • Max five cars per row.	• Hazard for player, damages player by 1 health.	•  • 

Name	Properties	Behaviours	Image(s)
Train	• Continuous movement. • Location and direction determined by row and cell. • Moves horizontally left and right at a rate of 32/14 px/tick. • 96x32px on screen. • Max two trains per row.	• Hazard for player, damages player by 1 health.	• 
Mr. Feathers	• Continuous movement. • Location and direction determined by row and cell. • Moves horizontally left and right at a rate of 32/42 px/tick. • Has orientation: West (facing to the left) and East (facing to the right). • 32x32px on screen. • Max two Mr. Feathers per row.	• Hazard for player, damages player by 1 health.	•  • 

Name	Properties	Behaviours	Image(s)
Spike Cell	• Has a static x integer location. y integer location determined by the row. • 32x32px on screen. • Max three spike cells per row. • Type of cell.	• Hazard for player, damages player by 1 health.	• 
Hedge Cell	• Has a static x integer location. y integer location determined by the row. • 32x32px on screen. • Max three hedge cells per row. • Type of cell.	• Wall, stops player from moving onto block, no damage taken.	• 

Name	Properties	Behaviours	Image(s)
Road Cell	• Has a static x integer location. y integer location determined by the row. • 32x32px on screen. • Always surrounded by road cells or world border on either side. • Type of row cell.	• Environment (ground visual).	• 
Track Cell	• Has a static x integer location. y integer location determined by the row. • 32x32px on screen. • Always surrounded by track cells or world border on either side. • Type of row cell.	• Environment (ground visual).	• 

Name	Properties	Behaviours	Image(s)
Grass Cell	• Has a static x integer location. y integer location determined by the row. • 32x32px on screen. • Type of row cell.	• Environment (ground visual).	• 
A Collectable	• Static integer location (x,y). Location determined by row and cell. • 32x32px on screen. • Value: 500 points. • Max of one A collectable on the play area at a time. • Max of one collectable per row.	• When collected grants the player 500 score.	• 

Name	Properties	Behaviours	Image(s)
B Collectable	• Static integer location (x,y). Location determined by row and cell. • 32x32px on screen. • Value: 200 points. • Max of two B collectables on the play area at a time. • Max of one collectable per row.	• When collected grants the player 200 score.	• 
C Collectable	• Static integer location (x,y). Location determined by row and cell. • 32x32px on screen. • Value: 100 points. • Max of three C collectables on the play area at a time. • Max of one collectable per row.	• When collected grants the player 100 score.	• 

2.11. Asynchronous (Input) Events

Event Name	Triggering Input Event	Description
Request player forwards.	<i>Up Arrow</i> key is pressed.	Sets player movement direction to forwards. Makes the player face forwards (facing away from the user). Triggers 'Player moving to fourth row'.
Request player backwards.	<i>Down Arrow</i> key is pressed.	Sets player movement direction to backwards. Makes the player face backwards (facing the user).
Request player leftwards.	<i>Left Arrow</i> key is pressed.	Sets player movement direction to left. Makes the player face leftwards.
Request player rightwards.	<i>Right Arrow</i> key is pressed.	Sets player movement direction to right. Makes the player face rightwards.
Button select.	Mouse cursor moves over a button.	Selects a button.
Button cycle.	<i>Tab</i> key is pressed.	Changes the currently selected button or selects a button if none are currently selected. It cycles through them in order from top-left to bottom-right.
Mouse Button Activation.	Left/Right mouse button is pressed.	If one of the two mouse buttons is pressed and a button is currently selected, activate the functionality of that button.
Keyboard Button Activation.	<i>Enter</i> , <i>Return</i> , or <i>Space</i> key is pressed.	Activates the functionality of any currently selected button.
Cursor movement.	Mouse movement.	Moves the position of the mouse cursor on the title / game over screen at a variable rate of speed (depending on how fast the mouse is moving).
Pause/ Unpause Game.	<i>Esc</i> key is pressed (in 1-player mode).	Prevents player and hazard movement. Any requests to move the player in any direction will be ignored. If the game is currently paused, pressing the key will unpause the game.

Event Name	Triggering Input Event	Description
Quit Game.	Q key is pressed while playing the game or in the Game Over Screen.	This will quit the current game or leave the Game Over screen. Either way, the player will go back to the Title screen.
Quit Program.	Q key is pressed while in the Tile screen.	This will quit the program.

2.12. Synchronous (Timed) Events

Name	Trigger Timing	Description
Move player	0.5 second (35 ticks)	Player moves in the set direction at 32/35 pixels/tick. (1 cell/0.5 seconds)
Spawn car	1 second (70 ticks)	Will have a 1/5 chance of spawning a car at the border of a row. (start/end of row).
Move car	0.8 second (56 ticks)	Car moves forward (horizontally) at 32/56 pixels/tick. (1 cell/0.8 seconds)
Spawn train	1 second (70 ticks)	Will have a 1/10 chance of spawning a train at the border of a row. (start/end of row).
Move train	0.2 seconds (14 ticks)	Train moves forward (horizontally) at 32/14 pixels/tick. (1 cells/0.2 seconds)
Spawn Mr. Feathers	1 second (70 ticks)	Will have a 1/10 chance of spawning Mr. Feathers at the border of a row. (start/end of row).
Move Mr. Feathers	0.6 seconds (42 ticks)	Mr. Feathers' moves forward (horizontally) at 32/42 pixels/tick. (1 cells/0.6 seconds)
Render Screen	0.2 seconds (14 ticks) - timing based around the fastest moving object in the game (the train).	Renders the world to the screen.

2.13. Condition-Based (Cascaded) Events

Name	Triggering Condition	Description
Player Collides with Car.	A Player has moved into a cell containing a car.	The player will lose a life. Triggers the 'Player Immunity to Harm' and 'Update Lives' events.
Car Collides with Player.	A car has moved into a cell containing a Player.	The player will lose a life. Triggers the 'Player Immunity to Harm' and 'Update Lives' events.
Player Collides with Train.	A Player has moved into a cell containing a train.	The player will lose a life. Triggers the 'Player Immunity to Harm' and 'Update Lives' events.
Train Collides with Player.	A train has moved into a cell containing a Player.	The player will lose a life. Triggers the 'Player Immunity to Harm' and 'Update Lives' events.
Player Collides with Mr. Feathers.	A Player has moved into a cell containing Mr. Feathers.	The player will lose a life. Triggers the 'Player Immunity to Harm' and 'Update Lives' events.
Mr. Feathers Collides with Player.	Mr. Feathers has moved into a cell containing a Player.	The player will lose a life. Triggers the 'Player Immunity to Harm' and 'Update Lives' events.
Player Collides with Spike Cell.	A Player has moved into a Spike Cell.	The player will lose a life. Triggers the 'Player Immunity to Harm' and 'Update Lives' events.
Player Will Collide with a Hedge.	A Player will attempt to move into a cell containing a Hedge.	The attempt to move will be canceled.
Player Will Collide with the Bottom Border.	A Player will attempt to move into the Bottom Border.	The attempt to move will be canceled.
Player Will Collide with the Left Border.	A Player will attempt to move into the Left Border.	The attempt to move will be canceled.
Player Will Collide with the Right Border.	A Player will attempt to move into the Right Border.	The attempt to move will be canceled.
Car Moves from the Left Border.	A car is moving starting from the Left Border and is actively moving away.	The car will be cropped from the left, but the amount of the car that is cropped decreases as it moves.

Name	Triggering Condition	Description
Car Moves into the Left Border.	A car is actively moving towards the Left Border and is colliding with it.	The car will start to be cropped from the left. The amount of the car that is cropped increases as it moves.
Car Moves from the Right Border.	A car is moving starting from the Right Border and is actively moving away.	The car will be cropped from the right, but the amount of the car that is cropped decreases as it moves.
Car Moves into the Right Border.	A car is actively moving towards the Right Border and is colliding with it.	The car will start to be cropped from the right. The amount of the car that is cropped increases as it moves.
Train Moves from the Left Border.	A train is moving starting from the Left Border and is actively moving away.	The train will be cropped from the left, but the amount of the train that is cropped decreases as it moves.
Train Moves into the Left Border.	A train is actively moving towards the Left Border and is colliding with it.	The train will start to be cropped from the left. The amount of the train that is cropped increases as it moves.
Train Moves from the Right Border.	A train is moving starting from the Right Border and is actively moving away.	The train will be cropped from the right, but the amount of the train that is cropped decreases as it moves.
Train Moves into the Right Border.	A train is actively moving towards the Right Border and is colliding with it.	The train will start to be cropped from the right. The amount of the train that is cropped increases as it moves.
Mr. Feathers Moves from the Left Border.	A Mr. Feathers is moving starting from the Left Border and is actively moving away.	The Mr. Feathers will be cropped from the left, but the amount of the Mr. Feathers that is cropped decreases as it moves.
Mr. Feathers Moves into the Left Border.	A Mr. Feathers is actively moving towards the Left Border and is colliding with it.	The Mr. Feathers will start to be cropped from the left. The amount of the Mr. Feathers that is cropped increases as it moves.

Name	Triggering Condition	Description
Mr. Feathers Moves from the Right Border.	A Mr. Feathers is moving starting from the Right Border and is actively moving away.	The Mr. Feathers will be cropped from the right, but the amount of the Mr. Feathers that is cropped decreases as it moves.
Mr. Feathers Moves into the Right Border.	A Mr. Feathers is actively moving towards the Right Border and is colliding with it.	The Mr. Feathers will start to be cropped from the right. The amount of the Mr. Feathers that is cropped increases as it moves.
Player Immunity to Harm.	Player collides with hazard(s).	The player will not lose lives if they cross paths with or remain on a hazard for two seconds.
Player Collides with collectable.	The player's bounding box intersects with the bounding box of a collectable, such as a letter grade 'A'.	The player will gain 100, 200, or 500 pts respectively if they walk over a 'A', 'B', or a 'C' letter grade. After the collectable has been collected it will disappear forever. Calls 'Update Score'.
Player Moves to Fourth Row.	Player moves forwards (<i>Up Arrow</i>).	Checks to see if the player is on the 3rd row from the bottom and if they are on that row, trigger 'World shift' and 'New Row'.
Player Moves Forwards to Unexplored Rows.	Occurs when the player moves forwards onto a previously untouched row.	Grants the player 100 points for entering a new row. Calls 'Update Score'.
World Shift.	Occurs when the player is in the 3rd row and moves forward.	The world will shift down one row and trigger 'Generate Row' to generate the new row at the top of the screen.
Generate Row.	Occurs upon initial world creation and subsequently, when 'World Shift' has shifted the world.	Generate will generate a new row with safe spaces, collectables, and/or hazards. (May spawn vehicles based off of the spawn chances for each type of vehicle as listed in the Synchronous Timed Events. [Chance for car: 1/5, train: 1/10]. Mr. Feathers, spikes, and hedges may also have a chance at spawning at 1/10 and 1/6, respectively.)

Name	Triggering Condition	Description
Generate Cell.	Occurs when a row is generated.	Can generate a grass, road, or track cell which is determined by the row.
Generate Spikes.	Occurs when a row is generated.	A spike cell has a 1/6 probability of being generated in place of a grass cell.
Generate Hedges.	Occurs when a row is generated.	A hedge cell has a 1/6 probability of being generated in place of a grass cell.
Generate Collectable.	Occurs when a row is generated.	Can generate a collectable onto a cell so long as it is not a hedge cell. Collectable 'A' has a 1/40 chance of spawning, collectable 'B' has a 1/30 chance of spawning, and collectable 'C' has a 1/20 chance of spawning.
Generate Car.	Occurs when a row is generated and every second after generation.	A car can be generated on road cells with a chance of 1/5.
Generate Train.	Occurs when a row is generated and every second after generation.	A train can be generated on track cells with a chance of 1/10.
Generate Mr. Feathers.	Occurs when a row is generated and every second after generation.	Mr Feathers can be generated on grass cells with a chance of 1/10.
Update Score	Occurs when the score has been added to.	Updates the score in the model.
Update Lives	Occurs when the player has lost a life.	Updates the amount of lives in the model.
Gameover	Occurs when the main player is no longer alive.	Causes a Game Over screen to be displayed.

2.14. 1-Player Hypothetical Gaming Session

When our user first opens the program, they are greeted with the title screen proudly displaying the title of the game: “Academia: Dreams of Madness”. It also provides white rectangular buttons such as *1-Player*, *2-Player*, and *Flee*. Say the user decides to start a one player game. Upon clicking on the *1-Player* button, they will be presented with the game window and a section on the top of the screen where score and lives are located. The user will start with a score of zero and three lives. First, they attempt to

move down with the *s* key in order to thoroughly explore the game world. However, the user is perplexed, as every time he presses the key, his ears are assailed with a hollow beeping sound. He eventually realizes that the arrow keys are used in the game and that the beeping is probably telling him that his key press is not recognized. So, he moves down twice with the *down-arrow* key. However, he finds himself unable to move down a third time, as he cannot move past the black border surrounding the screen. So, the user tries moving forward with the *up-arrow* key. Eventually, they will find their player on what looks like train tracks. Hearing the sound of a train horn, they hit the *Esc* key in order to pause the game and look around to see where the sound is originating from. Eventually, after not hearing the sound for a little bit, the user decides that the sound must have come from the game (how surprising). Turning their attention back to the screen, they noticed that there is a train on the row where his character is. Pressing the *Esc* key to resume the game, the train quickly zooms by, striking the user's character and causing the number of lives to be reduced to two. Following that experience, they panic and move forward with the *up-arrow* key lest another train comes along. The new cell they have settled on seems safe for a second or two. However, their character loses another life. Befuddled, they move to the left with the *left-arrow* key and glance over at the cell they were at before. They realize that the cell their character was on must have been a spike cell. They try to advance forwards, but there is a cell blocking their way. This must be a hedge the player thought to themselves. So, they decided to move right with the *right-arrow* key, forgetting that the spike cell lives there. So, the player is forced to stare in utter horror at their lives count which is now tragically displaying 0. Mercifully, after a short death noise (comprised of some do-do-dos), they found themselves facing the Game Over Screen. On the screen, there were two white rectangular buttons: *Play Again* and *Retreat*. Above the buttons was the player's score of 600 pts, which the user surmises that it means he was only able to advance six rows in the world. Deciding that they've learned from their initial attempts at the game and wanting another shot, they move their mouse over the *Play Again* button and their player is spawned anew in a different game world.

Wanting to compare this game session from the previous session, the player notices that his character spawned in the same position on the screen as last time he started. As well, the row the character was on (and the two preceding rows) consisted exclusively of grass cells and had no hazards on them. The user noticed that the row above his character consisted entirely of train track cells. Indeed, he actually saw a train go steaming off the row into the abyss (or whatever magical place the trains disappear to). After the hazard had passed, he moved his character up a row and noticed how he had just gained 100 pts for doing so. Thinking that his character would be safe for a few seconds, he decides he has time to drink a bit of water. Unfortunately, while he was doing so, the train horn sound effect came unexpectedly and he helplessly watched his character get run over by the train (and lose a life as a result). The player panicked (yet again) and moved his character up one row. Gazing at the top of the screen, he notices that on the newly visible row, there is a letter grade he can collect—a letter 'A'. Remembering that particular collectable is particularly valuable for improving one's score, he plots a course to snag that collectable. Unfortunately, he noticed too late that he was on one of several rows consisting entirely of road cells and swarms of cars. A car precedes to run over the player, thus causing the life count to decrease yet again. He quickly advances across the many roadways packed together and he notices that he hasn't received any more damage even though he is touching cars repeatedly. He surmises that after taking damage, his player is immune for a short period of time. He manages to avoid the cars before his immunity runs out and continues navigating the onslaught of hedges, spikes, and trains. Eventually, he makes it to the collectable and becomes overjoyed as the collectable

disappears and a nice little collecting sound is uttered by the underlying hardware. He remained overjoyed for a few seconds, blissfully unaware of Mr. Feathers (the speedy avian enemy heading towards him). He loses his final life and goes back to the Game Over screen, this time with a score of 1,500 pts. He realized this means that he traveled to 10 new rows and that he snagged a collectable. The user, starting to understand the game even better now, decides to hit the Play Again button for the second time. Shortly afterwards, however, he decides that maybe he's accomplished enough for the day. So, he presses the *q* key on his keyboard in order to get magically transported back to the Title Screen.

When the user sees the main menu again, they click on the Flee button, and then watch the game close. After that, they sit and reflect on what a wonderful enlightening experience it was. And you can enjoy it too! It is one of the greatest games out there for the Atari ST. There is nothing the team behind this wonderful experience can't do. Order the game today!

[1] Implemented as a circular array.

Chapter 3. Game Play Details for Core 2-Player Version



This part of the manual has not yet been implemented in the current stable version of the game. It is preserved to provide guidance for future updates to the game.

The game plays and looks almost identical to 1-Player mode with some differences. One difference is that the world is one row shorter in order to provide room for the opponent's score and health. Also, the game will be played simultaneously (competitively) by both players. A player can only see themselves and their own world; they are unable to see the other player or the other player's world. Actions undertaken by one player won't affect the other player or their world.

Additionally, there is a winning condition for the 2-Player mode. The player with the highest score when both of the players have lost all their lives will be the winner. Since players gain points every forward step they take onto a previously unvisited row, the player who lasts the longest might gain the most points and therefore, become the winner. However, that isn't always the case due to the fact that there are score-increasing collectables.

At the end of a two player game, the score of both players will be listed with a declaration of who has won the game. It is possible, although highly unlikely, for the players' scores to be tied. In that case, the game over screen will state "WINNER: TIE". Additionally, the Game Over screen will not specify a winner while there is still a player playing the game. Instead, it will just update the scores and other statistics on the page. All the buttons on the Game Over screen will still be selectable during this time. For instance, one player—while waiting for the other to finish—can click on the *PLAY AGAIN* button, which will create a new game world once the other player finishes and also clicks that button. Players are unable to pause the game in 2-player mode in order to avoid holding the results hostage. However, they are still allowed to quit. In the case of someone quitting, the person remaining will win by default.

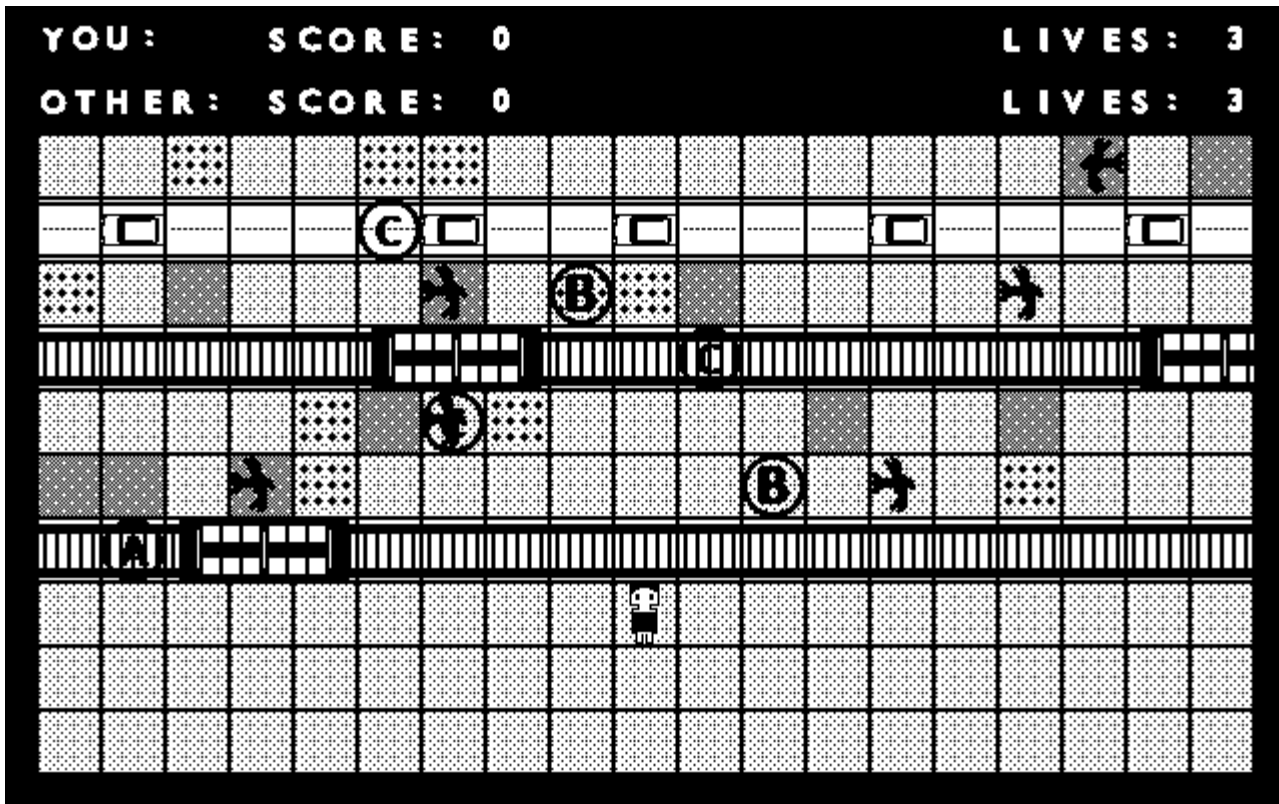


Figure 12. An example of what a player might see in a 2-player gaming session.

3.1. 2-Player Hypothetical Gaming Session

When our user (henceforth known as Player One) first opens the program, they are greeted with the title menu providing buttons such as *1-Player*, *2-Player*, and *Flee*. Noticing the *2-Player* button, he calls his friend over (henceforth known as Player Two), who just happens to have an Atari ST (who knew). Player Two installs the game and afterwards, Player One clicks the *2-Player* button to begin the game. The layout on both their devices are similar to the 1-Player layout, except with each others' scores and lives being displayed on the top of the screen below their own character's information. Anyway, both players have a rocky start. They both decide to advance forward with an *up-arrow* key only to find their characters smacking against the side of a car. They both lose a life as a result. After that, Player One sees a bird a few rows above their position. So, they decide to check out this bird. Eventually, Player One meets the bird head on, which turns out to be the evil fowl Mr. Feathers, and loses a life as a result. In the meantime, Player Two is moving up through the world until he suddenly hears what sounds like a train horn noise. Unfortunately for him, he is currently on the train tracks where a train is emerging from. As a result, he loses a life. He is briefly disappointed, but noticing that his friend's life counter has also decreased, he carries on, knowing that he still has a chance of winning. In the meantime, Player One notices a circled letter 'A' and thinks to himself that this must be one of those letter grades you can collect for additional points. Knowing it was worth the risk, Player One heads over to collect the grade. He moves left two steps and then advances forwards four steps in order to collect the letter grade. So overjoyed by the increase in his score, he fails to notice the movements of Mr. Feathers on the row he is situated on. Player Two notices that Player One is now dead as signified by the zero lives remaining. Taking joy that he is the last man standing, he proudly takes a step and then proceeds to get run over by a train. At the Game Over screen, both our players are surprised to

find that Player One had won since he had the most points. At that stage, the two players parted ways. That left our users clicking on the Retreat button.

Chapter 4. Sound Effects

Name	Description	Event which Triggers Playback
Pain Noise	"oof"	Player loses a life.
Death Noise	"wa wa wa" (decreasing in frequency every time).	Player has no lives remaining.
Walking Noise	"bloop"	Player is moving.
Train Warning	The train will sound its horn before entering the screen.	A train has been spawned.
Collecting Collectable	Will make a rewarding ding/bling sound (Mario collecting a coin).	Player collides with a collectable.
Beep	A hallow beeping noise.	Indicates that an invalid key was pressed.

The background music can be classified as horror melancholy. It utilizes the minor scale of tones and holds notes for a long period to contribute to a depressing atmosphere of impending doom.