SNAKE XENIA

Game Manual

Huzaifa Shabbir
MAWAHID ABBAS

Partners:

- Huzaifa Shabbir (23L-0647)
- Mawahid Abbas (23L-0613)

Description:

The player controls a long, thin creature, resembling a snake, which roams around on a 2d plane, picking up food (or some other item), trying to avoid hitting its own body in the area. Each time the snake eats a piece of food, its body grows longer, making the game increasingly difficult.

Libraries used:

SFML **(** Simple and Fast Multimedia Library **)**

Version : 2.6.1

Features:

- **Menu:**

The game features a simple menu which can be moved forward from by clicking the "play" button with the mouse or pressing the P key on the keyboard (controlled by the use of the *gameplaying* bool)

- **Score counter:**

Score counter is shown on the top left and controls the size of the Bodies vector (vector of the sprite snake_sprite_middle). Touching the food_sprite1 sprite increases the score and resets the position of the food.

- **Input detection:**

The input detection is done by using the pollevents() function which can be used to detect inputs using the "Keyboard" and the "Mouse" classes and an if condition, it compares the event IDs checking event type is also necessary as the key press of A and the left mouse button have the same event ID. Prevention of multiple direction input between each tick by the use of *Direction_changed* bool.

- **Direction:**

The direction is changed using the pollevents function which can be used to get inputs from the keyboard which are then used to set the direction using a global int variable "direction"

1=left

2=up

3=right

4=down

These directions are set using the arrow and WASD keys on the keyboard which then move the snake's head using the sprite.move() function

For instance:

When user enters ↑ or W, snake move upwards

When user enters ↓ or S, snake move downwards

When user enters → or D, snake move rightwards.

When user enters ← or A, snake move leftwards.

- **Tick system:**

The game features a tick system after which the snake moves and the body of the snake follows it. Without this tick system, the movement of the bodies would be impossible.

The tick system is implemented by using an object of the class "Clock" which calculates the time on its own using restarts() function.

- **Movement:**

The movement occurs using the class "vector2f()" and it affects the snake_sprite_head's movement which occurs after each tick and at the end of the tick, the rest of the body of the snake follows the previous position of the snake_sprite_head sprite

the objects in the Bodies vector and then the snake_sprite_tail follows.

The movement only occurs if the IsGameOver bool is set to false and the gameplaying bool is set to true.

- **Collision detection:**

Collision detection of the snake_sprite_head sprite with the tail , middle or the borders is detected using sprite.getGlobalbounds().intersects() which sets the IsGameOver bool to true.

- **SFX:**

The game features 3 sound effects the sound of dying, (death_sfx) , the sound of collecting the apple (eat_sound) and the mouse click sound (click_sfx) .

- **Restarting:**

After the *IsGameOver* bool is set to true, the user can press Enter or click on the restart "button" on the screen to restart the game or escape to exit. Restarting involves the use of the *retry* bool which is then used to reset all sorts of things such as the score, resetting the vector, resetting bools etc.

- **Exiting:**

Pressing escape anytime will close the game.

- **Endnotes**:

In concluding this project, we've successfully reimagined Snake Xenia, blending classic nostalgia with contemporary gaming excellence. The refined controls and captivating graphics, offer players an enhanced and engaging experience. As we wrap up development, our team takes pride in delivering a modernized version of a beloved classic, ready to captivate both new and seasoned players around the globe. The timeless allure of Snake Xenia lives on in this digital evolution.

Thankyou!!!