Snake – Design Document

# Introduction

Couple sentences

This application is a remake of the classic snake game. It is designed to exist on a 2d plane (made using a 2d array). Upon eating a fruit, the snake will grow an extra tail piece, if the head of the snake collides with a tail piece, the player will lose. The aim of the game is to grow the snake as large as possible, if the player is skilled enough, they should be able to fill the entire map with your tail which activates the win condition and the win screen.

# Snake

The snake itself acts as both a vector and a vector 2 to state the positions of each piece. The head itself is drawn using a vector 2 which is the part the player controls. The player controls the snake by giving the program a direction using either arrow keys or wasd. The player can only input one movement direction per frame. The program will then add the direction of the snake head to the position every frame.

## Tail

The tail of the snake is made using a vector made of vector2 (vector<vector2> tail). The vector is used to make a list of positions for the tail pieces by inputting a position (vector2 tailPosition). Each time the player head moves it will delete the end of the tail and will add a new tail piece right behind the head. Due to how the vector is made, when deleting the end of the tail on movement, the first vector piece is deleted.

# Map

The map is where the main game runs off of, including the draw and movement functions. On each frame, the entire map will be drawn, including the snake and fruit. The map also controls the players ability to teleport to the opposite side of the map when moving offscreen. This is how it is in the original snake. Additionally, I am able to add an option which would stop the player from moving offscreen.

## Fruit

Any time the player collides with the fruit piece, another tail piece is added. Each time the spawnFruit function is called, two random numbers are called, representing an (x,y) position. This position will then be compared to the map and if the map value represents a tail or snake head, generate a new location.

## Colors

Each map piece is represented by a colour, drawn at the defined position. The colour uses a switch case to which returns a colour. Case 0 will return darkGreen (the players head), 1 is green (the tail piece), 2 is gray (an empty tile) and 3 is the fruit piece.