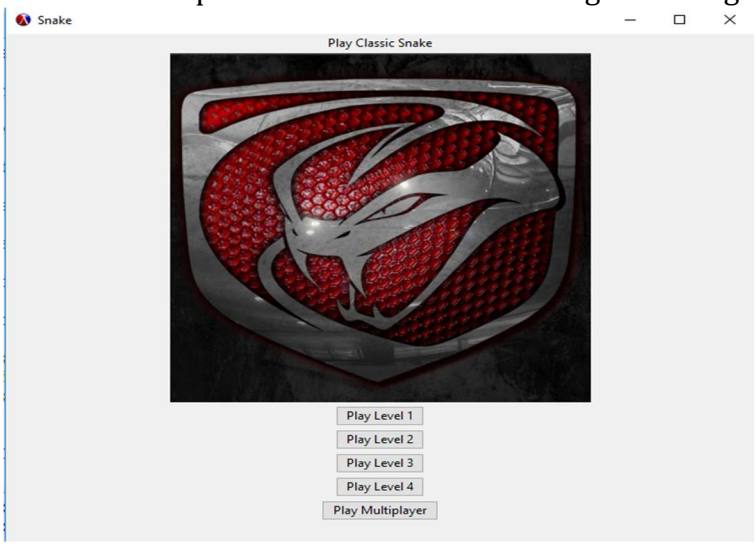
Racket Snake: A CS 154 Project

Created by:

Meet Kathiriya	Rajat Majoka	Phuntsog Wangchuk
160050001	160050031	160050109

Introduction

We have implemented the classic Snake game using DrRacket.



Opening screen of the game.

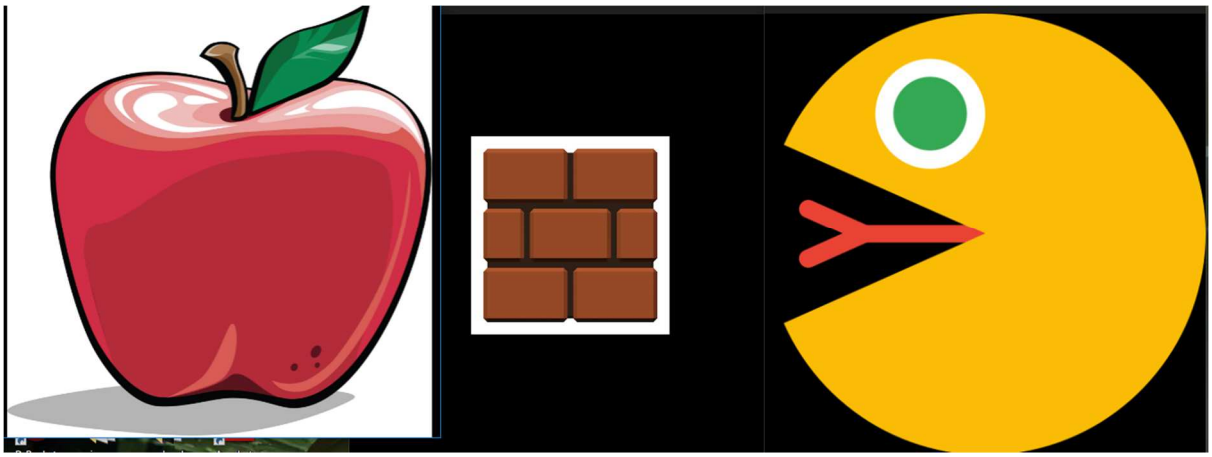
The main objective of this project was to practise and learn more about higher-order functions, abstractions and structural programming as well as to explore the various aspects and uses of the graphical library in Racket. Our program extensively uses the "universe" and "image" teachpacks to achieve its purpose.

Overall Design

Using multiple inter-dependent higher-order functions and some ingenuity, different level maps and multiplayer game modes have been created as well to add further depth to the game.

Graphics

All basic wall, food, snake-head and snake-body element are rendered as simple images arranged appropriately to simulate the snake's motion and gameplay.



Images used as Food, Wall and Snake-head respectively.

The image library was used to tie a particular file to each of these elements, enabling us to provide different appearances to different pieces.

Gameplay

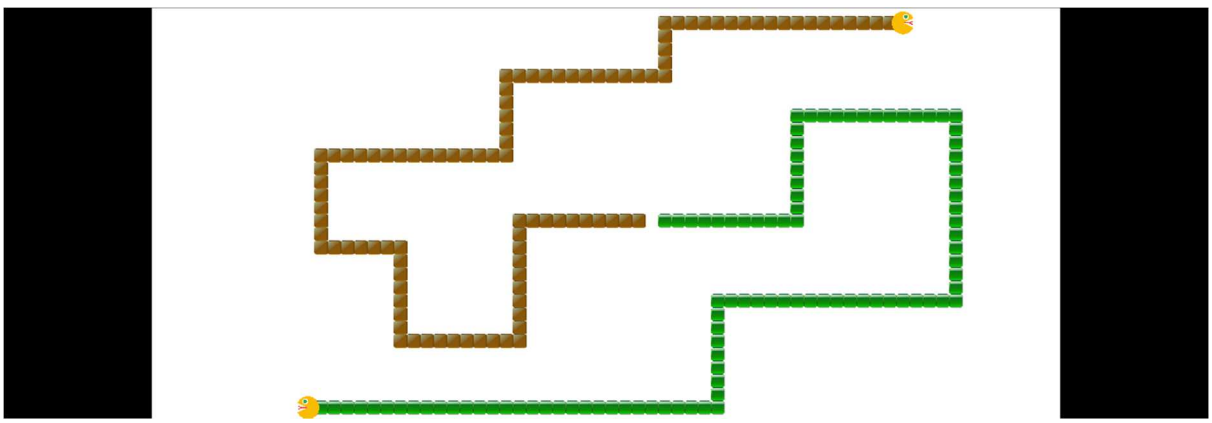
The "big-bang" function (provided in 2htdp/universe) enables us to provide keyboard input and progress through the game.

The main sub-functions include:

1. "pit": Defines the initial state of the screen.
2. "next-pit": Includes functions to be executed with each "tick".
3. "direct-snake": Describes the response to key inputs.
4. "render-pit": Helps to create the game screen with each tick.
5. "dead?" and "render-end": Decides whether Snake is dead and shows the game-over screen.

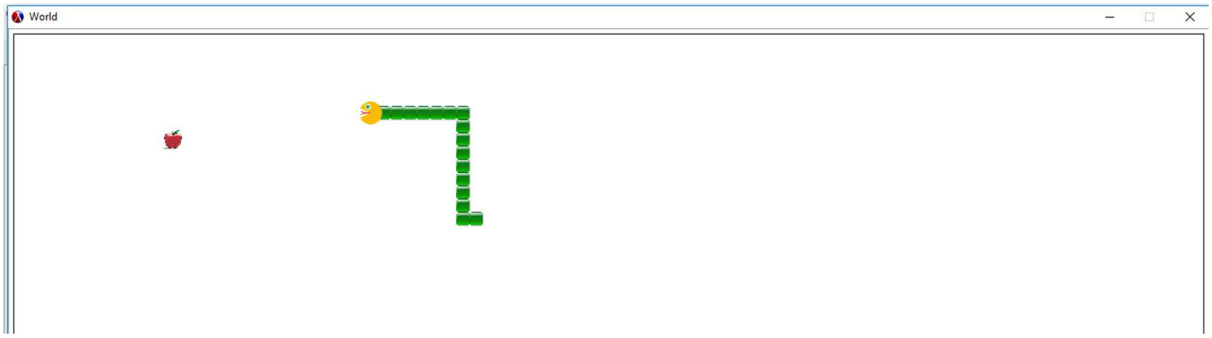
Multiplayer

Our version of Multiplayer is somewhat different from the regular game. Here, both snakes start from the centre and keep extending indefinitely in whatever direction the player wishes, filling up the screen-space. No food is involved. The objective of the game is to avoid biting the opposition and forcing them to do the same.

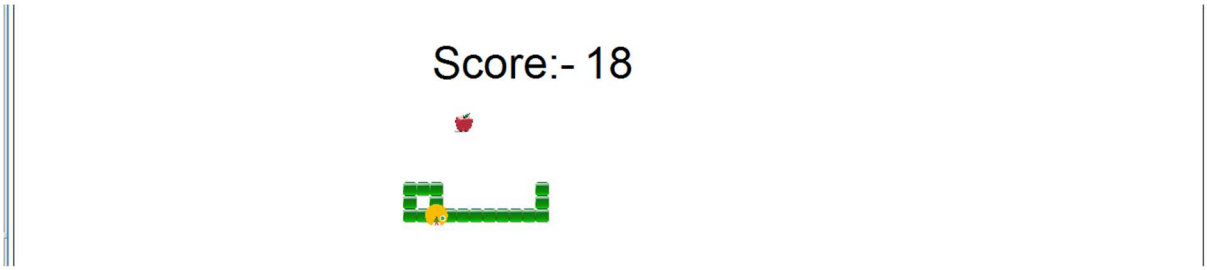


Multi-player gameplay screenshot

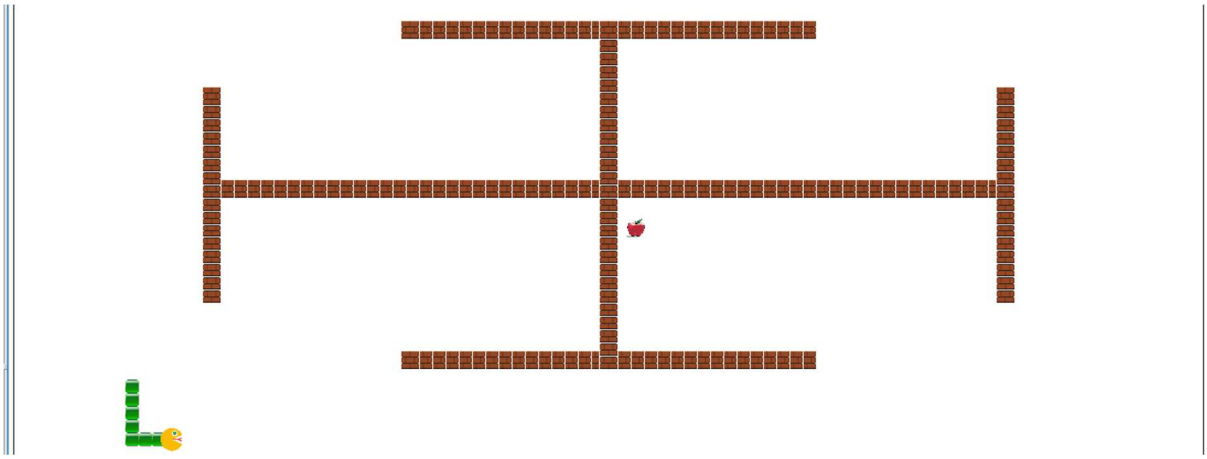
Sample Inputs and Outputs



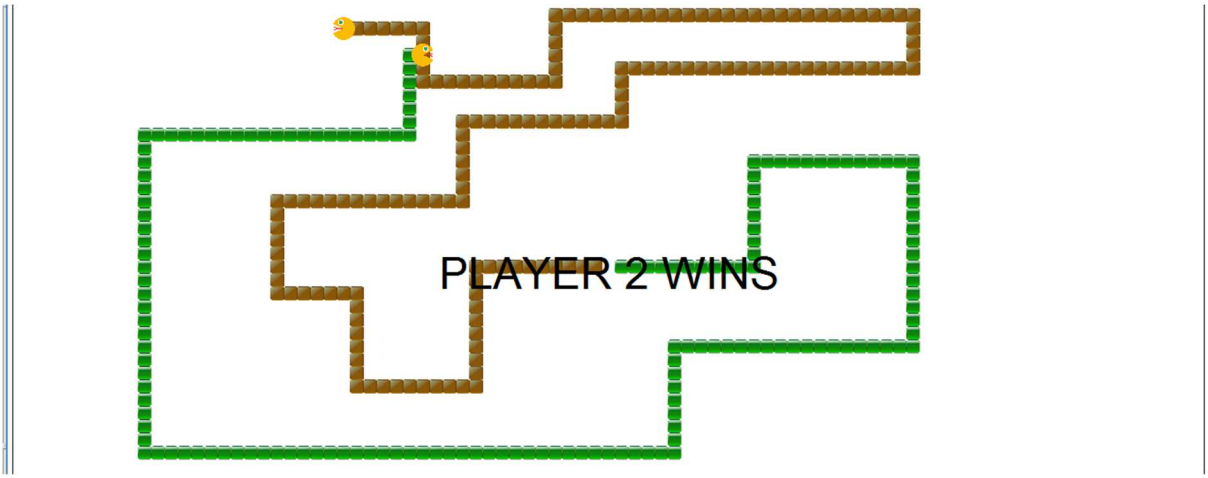
Level 1 gameplay screenshot



Game-over screen



One of the Map levels



Multi-player game over screen

Challenges

1. Modelling the initial, basic code into map-based and multiplayer modes required serious reprogramming as well as extensive repetition of code. We largely minimized this repetition in the map-level code by using multiple, smaller higher-order functions, but the multiplayer mode had to be remade.
2. There was a problem of partial overlapping of images when the snake moved next to walls and food. This was fixed by scaling every image as well as the entire screen-space by a common factor to minimize overlapping.
3. Food was often randomly spawned onto walls, rendering them out of reach. The (new-food) function was modified so as to avoid this. However, for simplicity, it only spawns food in positions where walls are absent in ALL MAPS instead of the current one.
4. Head-on or simultaneous collisions in the multiplayer mode always announce Player 2 as the winner, as the code checks whether Player 1 collided with anything before it checks for Player 2.
5. There is an initial hiccup at the launch of each game due to some unknown reason.