Jessica Lee

E-mail: jessicaleesthlm@outlook.com
Github: github.com/leejesslee

Downloadable source code, screenshots and executable files are available at: https://leejesslee.github.io/portfolioEng.html

Portfolio

Bachelor's thesis

Language: C#
IDE: Unity3D

My bachelor's thesis with a major in computer science. The thesis investigates wether Artificial Bee Colony Algorithm is a suitable algorithm for pathfinding in computer games by comparing the algorithm's performance and paths with the results from the A* algorithm. Both algorithms were implemented and tested on four different testing environments. The entire report and experiment is completed by me.

Alpha version of mobile game

Language: C#
IDE: Unity3D

An alpha version of a mobile game that works for Android and PC. The game uses touch interactions and allows the user to control the game with their finger. This alpha is a group project created by game designers and other programmers. For this project I worked with the camera.

2D PC game

Language: C++
IDE: Microsoft Visual Studio

Side scrolling game named "Flykten" is created by a group of 12 people, including graphic artists, sound designers, game writers and programmers. My role in this project was mainly lead programmer where I managed the communication between programmers and the project manager and also made sure that us programmers met the deadline. I also wrote a big part of the code to the game, some of it being GUI interaction, states and 2D animations.

3D PC game

Language: C#
IDE: Unity3D

Demo for a 3D game in third person perspective called "Räv". The demo was created by a group of 16 people by the course of 10 weeks. My role for this project was to write code together with four other programmers. I mainly worked with states and physics, but also with some GUI interactions.

Website

Language: HTML, CSS, JavaScript
IDE: Notepad++

A website for the game "Räv". The website's layout and design is coded in CSS while its content is coded in HTML. The gallery and the error message in the contact form are both coded in JavaScript. This is not an offical website for the game, but a project that I created on my leisure.

Evolutionary algorithm

Language: C#
IDE: Unity3D

A program which generates a desired color and prints out its RGB value. The program generates 6 random colors at a time. The user then gets to pick the color that is the closest to the desired color. The program then generates 6 new colors based on the selected color. This step is repeated until the desired colored is achieved. The algorithm uses evolutionary techniques by assigning the highest fitness value to the chosen colors from previous generations and then generating new colors based on which colors have the highest fitness value. In order to not get stuck in a loop a mutation will ocassionally occur which randomly generates a new color that is not related to the fitness values.