

# Grewh2D

Vilém Zouhar

*This document is intellectual property of the author.  
Created for classes Programming 2 (NPRG031) at [MFF UK](#).*

## Revisions

Creation date: 24. 02. 2018  
Document version: 02  
Last updated: 16. 03. 2018

## Notes:

24. 02. 2018 - Vilém Zouhar: document created  
16. 03. 2018 - Vilém Zouhar: first revision  
16. 03. 2018 - Vilém Zouhar: second revision

## Product Goal

Grewh2D (Genetically Refined Wheels in 2D) is semi-interactive visualisation of genetic based evolution of multi-wheeled vehicles. It is intended for students with eager interest in genetic algorithms as well as for teachers to demonstrate the capabilities of disparate approaches to algorithm design. Grewh2D serves only as a tool. No gamification is intended. There already exists a similar product, Genetic Cars 2. Grewh2D will offer better UI as well as some new features (terrain, goal and creature customization).

## Functional description

1. Menu + basic explanation of genetic algorithm approach used in this project
2. Randomization of the initial vehicle design and the environment, eg. gravity and shape. (extra functionality)
3. Task configuration (whether the goal is to carry as much luggage - other independent objects - as possible or to be the fastest)\*, genetic algorithm parametrization\*\*
4. Computation of the best movement strategy + displays of some generations trying to get as far as possible (part of the metric used), fast forward
5. Statistics displaying evolution progress (only basic info - best car so far, not as complete as in *genetic\_cars\_2*)

\*Only one of those will be actually implemented. The program should be extendable enough to allow for different metrics.

\*\*At least two parameters (population size, number of entities that live on)

## User interface

User will be able to affect (on very limited scale) the initial creature and the environment. Interactivity is not the main goal of this program as unrepeatability is achieved by randomness.

## Functional requirements

- A standalone application implemented in C# and Unity (the choice of C# 2D physics engines is limited)
- Published either as an executable (Windows, Linux, Mac) or, if not prevented by technical implementation, WebGL
- 2D vehicle and terrain generation (mostly terrain will be randomized)
- Robust evolution system (core of the application)
- Generation race simulated with a physics engine (determines the fitness of the individual)
- Statistics screen

## Data inputs

Very limited data input. At most a seed for pseudorandom generation. See *User interface*.

## Deadline

First draft: 01. 05. 2018 (at the latest)

Final release: 20. 05. 2018 (at the latest)