Progress report

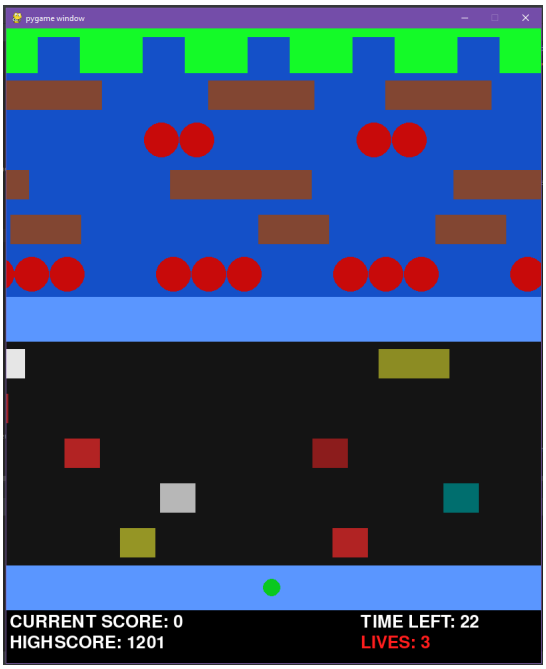
Dominik Trautman, Antoni Dąbrowski

30 stycznia 2022

1 Project overview

Our project contains two main parts which we are developing simultaneously. On one hand, we are creating a fast and visually attractive game environment. On the other hand, we are implementing a NEAT algorithm, in which a population of agents struggles with provided task.

2 Frogger



Implementation of the game our neural network will be learning to play is pretty much complete. The inputs we're planning to pass consist of a 15×14 board of short integers. Each tile might be either safe, deadly, goal or frog. Visualization of the game can be turned off. The speed at which the game runs can be adjusted. It's possible to simulate a 30fps game with about 366-time speedup, which should prove very useful for training.

3 NEAT

Briefly, the whole learning algorithm could be described as below. However, there are lots of nuances and details that make this approach pretty complex.

Algorithm 1 Neuroevolution of augmenting topologies

```
1:  procedure NEAT
2:      population ← initializePopulation()                    ▷ done
3:      species ← speciation(population)                       ▷ done
4:      fitnessValues ← measureFitness(species)                ▷ done
5:      while not terminationCondition(fitnessValues) do
6:          species ← selection(species)                       ▷ done
7:          species ← crossover(species)                       ▷ done
8:          species ← weightMutation(species)
9:          species ← topologyMutation(species)
10:         fitnessValues ← measureFitness(species)            ▷ done
11:     return species.bestIndividual
```

We are still developing some evolutionary operators, but our job is mostly done here. This leads to the next task, which is testing and tuning hyperparameters. It might be quite time-consuming so we are trying to come to this part as soon as possible.