Simple Book Example

TeXstudio Team

January 2013

Contents

Chapter 1

Introduction

Abstract

It is about the process of starting the game in an idea, to laying it down to be an actual project that can run, it will be separated into main 3 section, the first one is the wave class, second is ball and third is combine the game altogether in the main class

Choosing python as a programming language had a lots of good options in it:

1 Most of AI libraries are made in python

2 The numbers start at 1 with each use of the `enumerate` environment.

3 Another entry in the list

the first step of the game was was to generate a wave and duplicate it on both sides of the screen, not to be hard-codded as the game is made to be infinite function that can generate a wave for me and at the same time i can change in the variables of wave to make it harder for the players, these variable are the wave amplitude[1] or the wave frequency[2] with all of this in calculation, it means that i can make the game harder by making the behaviour unexpected for the next move, also to go extra step, the wave will decrees the gap between each other to limit the player movement

[1] is the maximum or lowest height the wave can go in one point to up or down

[2] number of waves that can go through a fixed distance in amount of time

Chapter 2

Related work

Chapter 3

Technologies

Chapter 4

Timeline

The first step of developing the game was choosing a programming language that would be helpful in both showing graphics on screen and implement an AI algorithm. The only option that was convenient enough was python as:

1 The most famous programming language with AI libraries that have a good documentation.

2 It is OOP, which means I can have a class for each component of the game easily to make an instance for the AI to train from.

3 A library to draw graphic on screen while it won't need heavy CPU usage that won't throttle the process of AI learning

Choosing a library for the game was the first step as it would define the characteristics. I went for **PyGame** because it was nearly the only one that is good enough with documentation to start with, and quit enough, the main focus isn't about making a game that will have that much of physics in it and 3d animation. For now, the imagined picture of the game is a rectangle as a screen that will have two main component of the game, a wave that is on one side of the screen vertically and one on the other side with the only difference is 350px (the starting amplitude is 50px and the screen width is 400px) and a ball that the **only purpose for it is to survive as much as it can, without hitting any of the sides of the wave**.

I'm minimalist when it comes to developing things, like there wouldn't be splash screen all over, and hard controls, not even hard rules, and it will be the approach I'm following with the game, to break it into steps:

1 The accent colour of the game will be black as a background

2 White will be used to show elements

3 Score counter on top

    3.1 AI mode: show generation and genome number
    3.2 AI mode: show total runtime
    3.3 AI mode: show the vision of the ball

During the writing here, I will raise some questions that might come to your mind while working on some parts (as they might have came to me too) and will try to answer them at the end of every section in the chapter.

4.1 Develop the game

4.2 Display the game

4.2.1 Draw

4.2.2 Collision

4.2.3 Move ball

4.2.4 Count distance

4.2.5 Show vision

4.2.6 Loop

4.2.7 Reset

Chapter 5

Testing

Chapter 6

Summary