

CSCI E-23a

Introduction to Game Development

produced by CS50

Colton Ogden
David J. Malan

Harvard Extension School
Spring 2018

Assignments

Lectures

Live Stream

Office Hours

Projects

Sections

Slack

Staff

Syllabus

Assignment 0: “Pong, The AI Update”

Objectives

- Read and understand all of the Pong source code from Lecture 0.
- Implement a basic AI for either Player 1 or 2 (or both!).

Getting Started

`git` and GitHub

First of all, welcome to GD50!

If you don't already have a GitHub account, head over to <https://github.com/join> and create an account. You'll use this account throughout the course to work on assignments, submit your work, and receive feedback.

You'll also need to make sure you have `git` installed on your computer. Go to GitHub's [instructions](#) for setting up and configuring `git` on your machine, and follow the steps there to install `git` (if you didn't already have it installed) and configure `git` with your GitHub username and email address.

GitHub Classroom

In this course, we'll use GitHub Classroom to distribute projects and collect submissions. To begin Assignment 0:

1. [Click here](#) to go to the GitHub Classroom page for starting the assignment.
2. Click the green "Accept this assignment" button. This will create a GitHub repository for your project. Recall that a git repository is just a location where your code will be stored and which can be used to keep track of changes you make to your code over time.
3. Click on the link that follows "Your assignment has been created here", which will direct you to the GitHub repository page for your project. It may take a few seconds for GitHub to finish creating your repository.
4. In the upper-right corner of the repository page, click the "Fork" button, and then (if prompted) click on your username. This will create a fork of your project repository, a version of the repository that belongs to your GitHub account.
5. Now, you should be looking at a GitHub repository titled **username/assignment0-username**, where **username** is your GitHub username. This will be the repository to which you will push all of your code while working on your assignment. When working on the assignment, do not directly push to the **games50/assignment0-username** repository: always push your code to your **username/assignment0-username** repository.
6. Next, let's set up GitHub Pages for this repository. Click on the "Settings" tab on the repository page. Scroll down until you see "GitHub Pages", set the "Source" to "master branch", and click "Save."

7. If you scroll down on the page again to “GitHub Pages”, you should see the URL at which your GitHub pages website will (soon) live! But first, we’ll need to add some a simple game to your repository.

Your First Game

Okay, let’s add a simple game to your repository. First, on your main repository page (<https://github.com/username/assignment0-username>), click on the green “Clone or download” button. Copy the “Clone with HTTPS” link to your clipboard (if familiar with SSH, you can use that instead).

Then, in a terminal window (located in `/Applications/Utilities` on Mac or by typing `cmd` in the Windows task bar), move to the directory where you want to store your project on your computer (recall that the `cd` command can change your current directory), and run

```
git clone repository_url assignment0
```

where `repository_url` is the link you just copied from GitHub. You will be prompted for your GitHub username and password

If you don’t want to enter your GitHub username and password every time you clone a repository, you can also optionally connect to GitHub using SSH. More information about that [here](#).

Go ahead and run `cd assignment0` to enter your repository.

It's Game Time

Your first assignment in the course will be a fairly easy one, since the dive into game programming can be deep enough as it is without having to implement an entire code base from scratch! Instead, we'll take the Pong example we covered in class and extend it in a small but fun way by giving one of the paddles (or perhaps both) logic for playing the game so that you don't always need a buddy to play the game with you! We'll approach problem sets in the course this way generally, taking the full code bases we've used in lecture and extending them so that you'll get plenty of experience interacting with fully implemented games. You can even use these projects as templates and jumping boards for your own games!

Of course, the code won't run if you don't have LÖVE2D installed, so we'll have to tackle that in addition to grabbing the code; choose whichever distribution is appropriate for your system here:

<https://love2d.org/>

For further information on how to actually run games, do just visit the following page:

https://love2d.org/wiki/Getting_Started

Once the code and LÖVE2D have been downloaded and installed, the actual change you'll be making to the code base is small, but it will require you to understand

what many of the pieces do, so be sure to watch Lecture 0 and read through the code so you have a firm understanding of how it works before diving in! In particular, take note of how paddle movement works, reading both the `Paddle` class as well as the code in `main.lua` that actually drives the movement, located in the `update` function (currently done using keyboard input for each). If our agent's goal is just to deflect the ball back toward the player, what needs to drive its movement?

Your goal:

- Implement an AI-controlled paddle (either the left or the right will do) such that it will try to deflect the ball at all times. Since the paddle can move on only one axis (the Y axis), you will need to determine how to keep the paddle moving in relation to the ball. Currently, each paddle has its own chunk of code where input is detected by the keyboard; this feels like an excellent place to put the code we need! Once either the left or right paddle (or both, if desired) try to deflect the paddle on their own, you've done it!

How to Submit

Step 1 of 2

1. Go to the GitHub page for your **username/assignment0-username** repository (note: this is different from the **games50/assignment0-username** repository).
2. On the right side of the screen, click the Pull request button.

3. Make sure that the “base fork” is `games50/assignment0-username` , and the “head fork” is `username/assignment0-username` .
4. Click “Create pull request”.
5. On the next page, click the “Create pull request” button again.

Step 2 of 2

Fill out [this form](#)!

Congratulations! You’ve completed Assignment 0.

