Lab 3: `raylib` Pong

Warm-up (10 mins)

Read through the following article. Then have a look at Python documentation, it should make more sense.

In the reflection box on the back, write down the two main takeaways for you.

Pong!

Play a simple Pong version. *Used to work great couple of weeks ago... now thx time machine!*
From Ramon's course, we have c code for a basic Pong!. It uses a lot of the concepts we covered. Watch Ramon's Pong core running.

Getting `raylib` compiled for your machine, and then linked to a c game, is more complicated than Python... If you are up for the challenge, go for it. You need to compile `raylib` for your architecture, look at the wiki, and then I suggest using a template.

Implementation

Your `starter_code` is last class demo. We use it to develop our game in a organized/good form.
Read through it: how is it different from yesterday?

Our focus is to get the main game play, using the provided `c_code` reference: your core task is to implement the gameplay screen. Working in *pair*, you should iterate through the following steps.

1. Each of you should identify the next smallest and logical feature to add next in term of expected gameplay/gameloop. A **goal** that you can define, implement and test right away.
2. Talk and agree with your partner on which one of your possible different goals, you should tackle next.
3. Write down in your own words that goal on your roadmap. (Build a bullet list at the back of this sheet).
4. Identify and **annotate** in Ramon's code/print-out the relevant/inspiring lines you are working on. You should have a definite plan and check the cheatsheet relevant functions.
5. Discuss with your partner, where and how the equivalent Python code is integrated in your code base.
6. Use pair programming (rotating each time who is navigator/driver) to implement and test that new feature (update/draw/..).

Feel free to refer to other examples we wrote in class.

Requirements

- Use our additional code for inspiration, but comment it out, and delete it when appropriate.
- By the end, everything (ball and players) should moved using `get_frame_rate` (as we did for the cat).
- You are expected to improve on c code. It is there to get your started; you should make use of everything we have learn.

Submit

- Hand-in this paper sheet (each of you)
- Upload files on Moodle (one per group)

Make sure to show me your work before the end of the lab.

Before tomorrow's lecture

Watch and take notes on this video. Link also on our schedule.

Fill-in below

Name: _____ Partner: _____

Reflection

Roadmap