

Lab 11: Alien Python

Version as of 2/1/2017

For the most updated assignment, please see this link:

<https://www.cs.hmc.edu/twiki/bin/view/CS5/Lab11>

CS5 Web > Homework11Gold > Lab11

Next HW: **Homework 2: *Functioning recursively*** will be due on: **Mon., Feb. 6, 11:59pm**

Next Lab: **Lab 2: Turtle!!** will be held on: Tue./Wed. evening, Jan. 31-Feb. 1

Submissions: [CS submission site](#)

Lab 11: Alien Python!

(30 points; individual or pair)

This problem invites you to explore the object-oriented VPython graphics package, also known as `vpython`. The goal of this lab is to explore the 3d library and to submit a zip file `hw11pr1.zip` with your code and at least two screenshots of your 3d models in action.

For those who would like to use VPython more extensively in cs5, one of the three final-project options uses VPython (to create a game or interaction of your own design).

Running VPython

VPython does not run with Python 3 at the command-line. That said, there are three different ways you can run VPython for this lab and, if you wish, for the final project:

- **Classic VPython** You can install Python 2 and its VPython library. This is the most complete and best-supported approach. It allows keyboard interactions and can happily co-exist with Python 3 on your machine. (This is probably the best option - at least eventually - if you think you might want to try the VPython final project.)

[Here are the instructions for installing and completing the lab with this "classic" version of VPython.](#)

- **VPython on our Lab Macs** The HMC CS Lab macs have VPython installed. Though not as convenient as your own machine, this is a great way to complete the lab and try out 3d graphics, without worrying about installing something else on your own laptop.
[Here are the instructions for running VPython on the Lab Macs.](#)
- **Jupyter VPython** is the browser-based version of VPython that is in the process of replacing the classic VPython. This *does* use Python 3. It does, also, require an install of the VPython libraries. These instructions have been tested with the version of Anaconda Python we're using this term.
[Here are the instructions for running Jupyter VPython](#)

Note! If one of these options does not work for you, seek out help -- and it may turn out that you'll need to try another of the options... . To date, we've always been able to get classic VPython to work on every student's computer *eventually*

Completing Lab 11

Each of the above approaches to VPython includes starter code in three files:

- `cylinder.py`
- `bounce.py`
- `billiard_bounce.py`

and there are several challenges for running and modifying these (especially the last one). The results, including two screenshots of your custom-designed 2d models, should be zipped into an archive named `hw11pr1.zip`

When you've made those changes and created your `hw11pr1.zip` archive, submit that to the [usual place](#) .

Then, you've completed Lab 11. If you liked VPython, consider using it for your final project... !