Principles of Robot Autonomy I

Section Logistics
Sections

• Modeled after sections in CS 106A/B/X/L
• Provide hands-on experience for commonly-used tools in robotics
  • AKA tools you’ll be using for your homework and final projects
• Taking feedback from previous years to heart
Section Logistics

• First 15-30 minutes will be a presentation about the aims of the section, references, and a description of the hands-on activity you’ll be doing

• Rest of the time (1.5+ hours) will be for you and a partner (your tablemate) to complete the hands-on activity

• You submit your results on Gradescope when you’re done
Do I have to stay the whole time?

• Once you complete the activity and submit your results, you can leave
Do I have to arrive on time?

• Yes

• ... unless you have an overlapping class conflict. In that case, you should still arrive ASAP and make a group with someone else that is arriving similarly late
  • If you’re the only one that arrives late, then you can join an existing group

• Section slides and the activity handout will be posted online, so you can still catch up

• However, we will not stay after hours
Questions about Section Logistics?
Principles of Robot Autonomy I

Section 1: Introduction to Python2.7, Git, and Installing VMs
Aims

• Install Ubuntu virtual machine (VM) locally
  • Will be very useful for interfacing with your robots later!
• Learn how to use Git for version control
• Start working with Python 2.7 and some of its most common packages
Docker

• Docker is a lightweight application that runs software in containers

• A container is a standalone unit of software that “contains code and all its dependencies so the application runs quickly and reliably from one computing environment to another”

• In case you don’t have a Linux-based machine, it comes with a Linux OS and uses it to run these containers

• More information available at: https://www.docker.com/resources/what-container
Git

• Popular source code version control system
• You probably already use it!
  • Github, BitBucket, etc. all support Git
• Replaces the days of
  • Important_doc.docx
  • Important_doc_v2.docx
  • Important_doc_final.docx
  • Important_doc_final2.docx
  • Important_doc_final2_USE_THIS_ONE.docx
Python 2.7

• We assume you already have some programming experience at the level of CS 106A

• As a result, rather than providing a full-blown tutorial about Python, we’ll direct you to last year’s Python + NumPy tutorial (hands-on!)

• It can be found online at: http://asl.stanford.edu/aa274_win1819/pdfs/recitation/Tut3_NumPy.pdf
Section 1

• Focuses on Python and common use-cases for it in this course
• We’ll ask you to perform a few basic mathematical operations and plot the results