Project 0: Python Tutorial

CS 4300: Artificial Intelligence
University of Utah

This project serves as an introductory material to the primitives in Python programming. The objective of this project is to get your hands wet in Python which would be the programming language that would be used throughout all the projects in this course. Do read through this tutorial for a refresher on Python and for basic instructions for installation(detailed instructions are provided in this pdf).

The instructions below are for Python 3.6, however versions newer than 3.6 should work as well.

1 Python Installation

The right version can be installed from the Python website (link). The suggested alternative is to use an environment manager like **conda**. Conda comes installed with the distribution platform **Anaconda** and **Miniconda**(bootstrap version of Anaconda). An environment manager can help isolate project environments from each other which means that you can have multiple versions of Python and other packages installed without any conflicts. Instructions for using conda are provided below:

1. Install Anaconda or Miniconda. I suggest the latter since it is a lightweight version of the former and does not install all the packages that Anaconda installs. Basically you have more control of what is installed. This is the link to their installation instructions (link). Choose the latest installer for your OS.

2. Open your terminal(On windows, open the Anaconda prompt which is installed along with your conda installation. This sets all the environment variables so that you can access all the required commands).

3. Create your environment using the command:
conda create −−name <name of environment> python=3.6

4. Once it is created, you can activate the environment using the command:
activate <name of environment>(Windows)
conda activate <name of environment>(Linux)

5. Now within this terminal, you have access to Python 3.6 for your project ie you can run all your scripts(.py) for the projects from within this terminal using the command:
python <name of script>.

6. You can deactivate the environment by using the command:
conda deactivate

If you have any issues with installing Python on your local machines or have other con-

straints, you can make use of the CADE lab machines which have Python installed.

2 (OPTIONAL) Creation of Virtual Environment using venv

While conda is the suggested environment manager, virtual environments can be created using the package venv also. This step isn't necessary if you are using conda environments. It(venv) comes pre-installed in python3. Please follow the instructions below to set up the virtual environment.

- Navigate to desired virtual environment installation path (where you want to install it)
  cd <Folder name> or <path>
- **Creating virtual environment:**
  python3 -m venv venv_test
  Note: If you only have python 3 and do not have python 2, you need to use the command python -m venv venv_test
  Here venv_test is the name of the environment that you are creating.
- **Activate virtual environment:**
  Linux: source venv_test/bin/activate
  Windows: call venv_test/Scripts/activate
- **Deactivate virtual environment:**
  deactivate

3 Introductory Python tutorial

For those who are new to Python, please go through all the examples and code snippets under the section titled Python Basics from the tutorial. It also includes some UNIX commands. For those who have already been programming in Python, this could be a good time to refresh yourself on some of the topics.

4 Project 0 Source files

Download p00.zip and unzip it. The project directory has many files including an **autograder.py** file which will be used to check the correctness of your code.

5 Python programs (3 pts)

As part of Project 0 you would have to complete three questions.

1. Question 1

2. Question 2
3. Question 3

described in the tutorial. The instructions in each question clearly tells you which file(s) would you need to modify for the corresponding question. Complete the code for all the three questions and include comments in your code wherever necessary. All questions are equally weighted.

6 Self Analysis (1 pt)

5.1 What was the hardest part of the assignment for you?

5.2 What was the easiest part of the assignment for you?

5.3 What problem(s) helped further your understanding of the course material?

5.4 Did you feel any problems were tedious and not helpful to your understanding of the material?

5.5 What other feedback do you have about this homework?

7 Evaluation

Your code will be auto-graded for technical correctness. Please do not change the names of any provided functions or classes within the code, or you will wreak havoc on the autograder. However, the correctness of your implementation – not the autograder's judgements – will be the final judge of your score. If necessary, we will review and grade assignments individually to ensure that you receive due credit for your work.

8 Submission Instructions

- For the final submission you will be turning in the following items (please do not zip them together but instead submit them separately):

  1) **addition.py**

  2) **buyLotsOfFruit.py**

  3) **shopSmart.py**

  4) a PDF document containing your responses to questions.

- Please ensure all the submissions are done through Gradescope. Please do not email the instructor or the TA's with your submission. Submissions made via email will not be considered for grading.

- **Written Answers:** Place all your written answers in a single PDF document. This should be clearly named in the format ⟨uid1⟩-Proj⟨number⟩-answers.pdf, where ⟨uid1⟩ is your Utah uID and ⟨number⟩ is the Project number. Ex: u0004300-Proj0-answers.pdf. Please make sure to write your and your partner's name at the top of the document!
- **Naming:** Your python file(s) upload should be kept under the original name(s), i.e. **addition.py, buyLotsOfFruit.py, shopSmart.py**
- For this project fill in portions of the files to edit. Once you have completed the code, name them as per the conventions stated above and submit the requested files via Gradescope.
- In the pdf, each group member is expected to provide their own responses to the self analysis, but can work together on the code and other written responses.
- Do **NOT** install/use additional libraries or packages other than what is provided and the python standard library.