

Lab Guidance – COMP 499/691

Labs are a major learning opportunity in this course, in terms of practical skills and understanding how the algorithms work. Students are asked to implement some Conversational AI models introduced in the lectures.

How are labs organized? Typically, during the lab, the TA will start showing a tutorial on the topics addressed during the lectures. Then, the TA will introduce the lab assignment of the week that will be made available on Moodle at the beginning of the week. The students can start working on the lab assignment during the lab session and ask clarifying questions to the TA. However, very likely you have to finish the lab at home. You have until the end of the week to submit the lab.

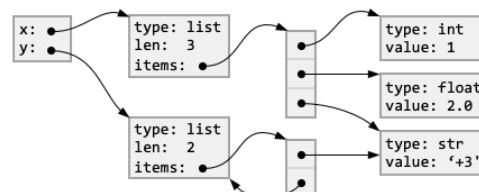
As the lab will be made available at the beginning of the week, you are encouraged to take a look at it (and even start solving it) before your lab session. This way, you can take advantage of the lab session to ask clarification questions to the TA. The deadline will be the same for all the students (i.e., the end of the week), regardless of the day on which the lab session starts.

The lab assignments are Colab notebooks. They contain a list of exercises that requires some code written by the students. For instance:

Exercise 1.1 – Python variables, objects, and references

The [Python language review](#) for this course contains a diagram of how a list object holds references to the other objects that are "in" the list. Go look at the example diagram in the "List" section.

Now consider the diagram below:



It depicts two list objects, an integer object, a float object, and a string object.

Write Python code that creates objects in memory as depicted in the diagram, including the references shown. Write your code in the cell below. Aim for 3 lines of code, and try not to create any extra (temporary) objects not shown.

In []:

Check your answer by running the code cell below.

```
In [ ]: assert 'x' in globals(), "You didn't define variable 'x'"
assert 'y' in globals(), "You didn't define variable 'y'"
assert isinstance(x, list), "Variable 'x' should refer to a list object"
assert isinstance(y, list), "Variable 'y' should refer to a list object"
assert len(x) == 3, "The list object referred to by 'x' should have length 3"
assert len(y) == 2, "The list object referred to by 'y' should have length 2"
assert isinstance(x[0], int) and x[0] == 1, "Slot x[0] should refer to an int object with value 1"
assert isinstance(x[1], float) and x[1] == 2.0, "Slot x[1] should refer to a float object with value 2.0"
assert isinstance(x[2], str) and x[2] == '+3', "Slot x[2] should refer to a string object with value '+3'"
assert y[0] is x[2], "Slot y[0] should refer to the same object as slot x[2]"
assert y[1] is y, "Slot y[1] should refer to the same list object that variable 'y' does"
print('Correct!')
```

You have until the end of the week to conclude all the exercises and submit the notebook filled with your code in Moodle (see the exact deadline on Moodle). We will grade the code submitted.

Why use Colab Notebook and not Jupyter Notebooks?

Colab Notebooks are essentially Jupyter Notebooks that run on a remote machine (and not locally). This has the advantage that all students will have access to the same system, machine, GPU, libraries, etc. This minimizes possible replication issues and other technical issues related to your specific machine or environment (that are very hard to debug as you cannot access your machine).

Submission. In Moodle, we will provide a link to the Colab Notebook. Note that to work with Colab, you need a Google Account. Please, create one if you don't have one. When you are in the specified Colab Notebook, save a local copy of the notebook such that you can start editing it (file=>save a copy in Drive). When you finish the lab, you can download your lab (file>download=>Download .ipynb). Finally, you need to upload your file on Moodle. You have until the end of the week to submit your lab (see the exact deadline on Moodle). Our goal is to provide prompt feedback to the students and grade the lab as soon as we can (e.g., 1 week after the deadline). For this reason, we cannot grant any extension to the specified deadline.

Evaluation. There will be 12 labs worth 1% each up to a maximum of 10% of the total grade. This way, even if one misses two of the labs can potentially get a full grade anyway.

Feedback. The labs are designed such that we test your code along the way or we show the desired outcome. If the test passes or the outcome matches with the targeted one, you are doing well. If tests are not passing, there might be something wrong with your solution. If you have some doubts, you can contact your lab TA (by email and, if really necessary through office hours). Feel free to ask questions during the lab sessions as well. If you are unsure about your grade, please contact the marker.

Lab attendance. We strongly encourage the students to attend the labs in person and take advantage of this opportunity (see the schedule on the Syllabus). This way we give all the students the possibility to have proper support from the TA.

Collaboration. Students are encouraged to help each other with labs verbally, but copying code is forbidden. Feel free to ask for the help of the TA as well. The TA will be helpful to clarify your doubts and provide hints for a solution, but the TA won't provide you with the code. You must be the author of all your submitted answers.