

CSC108 Lab 6: While Loops & Lists

October 13th – October 21st 2025

1 Overview

Welcome to your sixth lab in CSC108. This lab will focus on using while loops and give you some practice with lists. Much like how you went from for loops to nested for loops, you should expect some form of nested lists next week :)

2 Lab Tasks

You will have four functions to implement for this lab, and one of them will just be to re-implement `loopy_madness` with while loops instead of for loops. Aside from this change, all other specifications and restrictions on the function are otherwise the same as last week (i.e., Lab 5).

The specifications for the functions have already been laid out in `lab6.py`, so in order to prevent duplicate information, please download `lab6.py` and complete the functions according to their docstrings. You will not need functions from the previous lab in this one, besides the re-implementation described above.

Lastly, this is already noted in the lab, but we are noting it here just to be extra sure you've seen it. There are restrictions on what operators and constructs you are allowed to use for this lab (and possibly for future labs). For this lab in particular, you may not use any dictionary or dictionary methods, or any `try-except` statements. We are restricting the use of these statements for two reasons: first, we haven't covered them in class yet, and as a result we want you to work with the material that you have already learned in class to solve the problems set out in this lab. Second, some of these constructs like `try-except` are often used improperly, and for the purposes of this lab, there is no need to use them.

3 QuickTA

We are excited to bring back QuickTA, our interactive chat-based tool designed to provide real-time answers and clarifications while you are problem solving. It can be used to assist you with this lab, while you implement the functions in this lab.

Access QuickTA for immediate responses and guidance using this link: <https://quickta.utm.utoronto.ca>

Note please attempt to complete each post-conversation survey. Your responses will help improve the future implementations of our chat-based tool!

QuickTA is here to aid you on your Python programming learning journey at UTM, but it should not be used to obtain direct solutions. The responses from QuickTA should not be regarded as those from a member of the CSC108 Teaching Team. If you find yourself uncertain about any information, it is essential to seek clarification from a human!

Feel free to use QuickTA when you require assistance solving your functions, but remember to use it responsibly and within the guidelines. All work that you submit must be created and understood by you!

QuickTA is an experimental tool, and as it leverages the power of generative AI, which means that it may produce inaccurate information. If you notice it produces an error, please report the error using QuickTA reporting feature.

4 Sanity Checks

Often we forget to do our own sanity checks, so this is a reminder. Have you:

1. written at least 3 doctests per function and ensured that they pass? (you **must** do this)
2. considered edge cases in your doctests?
3. uploaded your file to [MarkUs](#) and run the student tester?

Remember: you should be running the student tester and using the student correctness and style results to refine your submission. Submit your code often and ahead of time!

5 Final Check!

Once you are finished, submit your final version of `lab6.py` to [MarkUs](#). **Please check to make sure that it's the right file!** You'll receive a grade once everyone has submitted and automarking has been done. See you next week!