Last week:

*list comprehension:*

* *a compact/concise way to produce lists. We will use this often.*

*matrices*:

* on paper element , in python [[10,8,6],[4,2,9],[7,5,3]], element

*main guard*:

* if \_\_name\_\_ == ‘\_\_main\_\_’:

*function overloading:*

* Actually, python does not allow proper function overloading.

*some PyCharm tricks:*

* *Folding: Ctrl+-, Unfolding: Ctrl++, Folding All: Shift+Ctrl+-, Unfolding all: Shift+Ctrl++*
* *Go to function: Ctrl+click*
* Commenting blocks: *Ctrl+/*
* *Uncommenting blocks: Ctrl+/*

*Pop and insert* (Determinant.py, Gauss\_Elim.py)

* *pop*: pulls a row (element) out of a list and all elements “below” that element decrease their index by 1.
* *Insert*: pushes a row (element) into a list at a given index and causes all elements “below” that index to increase their index by 1.

*lambda functions (*[*https://en.wikipedia.org/wiki/Lambda\_calculus*](https://en.wikipedia.org/wiki/Lambda_calculus)*)* (Functions passed to Functions and Lambda.py)

* A small, anonymous function.

*callbacks - (*[*https://en.wikipedia.org/wiki/Callback\_(computer\_programming)*](https://en.wikipedia.org/wiki/Callback_(computer_programming)) *)*

* A function can be passed as an argument (a.k.a., a *callback function*) to another function (a.k.a., the *called function*)
* The *callback function* is executed from within the *called function*.
* Re-visit ‘Second Program – Functions and Arguments.py’ from Week 1.

Revisiting:

*List comprehension* (<https://www.w3schools.com/python/python_lists_comprehension.asp>)

* A flexible way to build a list using an iterator and predicate (logical condition).
* e.g., build a 10×10 array of integers where all the odd numbers are negative, and all even numbers are positive.

*Using recursion to find a determinant of a matrix*

* A matrix is singular if det(A) = 0
* *recursion* refers to a function calling itself.

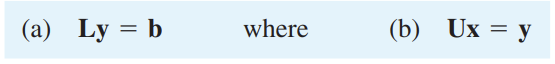
New:

*Input:* Interacting with the user through the command line interface (cli).

* So far, we have been “hard coding” our inputs (e.g., matrices) without allowing the user to provide input.
* We can get input from the user in the form of strings.
* We need to properly convert the input into float, int, bool, etc.
* We can provide default responses.

*A white background with black text

Description automatically generated*

*A white paper with black text and numbers

Description automatically generated*