

# Lab 5: Linked Lists Solution

## 4) Additional exercises

### Generalizing `--getitem--`

The implementation we've provided for `--getitem--` has many shortcomings compared to Python's built-in lists.

Two features that it doesn't currently support are negative indexes and slices (e.g., `my_list[2:5]`).

Your first task here is to investigate the different ways in which Python supports these operations for built-in Python lists; you can do this by experimenting yourself in the Python console, or by doing some reading online.

Then, modify the linked list implementation of `--getitem--` so that it handles both negative indexes and slices.

Note that a slice in Python is actually a class: the expression `my_list[2:5]` is equivalent to `my_list.__getitem__(slice(2, 5))`.

Use `isinstance` to determine whether the input to `--getitem--` is an integer or a slice.

The fully general method signature of `--getitem--` should become:

```
1 def __getitem__(self, index: Union[int, slice]) -> Union[Any,
    LinkedList]
```

Note: slicing should always return a new *LinkedList* object.

This means that for a given slice, you'll need to create a *LinkedList* and new *\_Nodes* as well, in a similar manner to how you implemented the more powerful initializer at the end of Task 1.

## Matplotlib Practice

Use *matplotlib* to plot the results of your timing experiments, using the same approach as last week (See matplotlib section in lab 4).