

Homework 3 – ADT

Exercise 1

Implement the classes **Node** and **LinkedList** and provide the following functionalities:

- **Node:**
 - `get_data()`
 - `get_next()`
 - `set_data(new_data)`
 - `set_next(new_next)`
- **LinkedList:**
 - `add(item)` – adds item to the front of the linked list.
 - `append(item)` – adds item to the end of the list.
 - `insert(pos, item)` – adds item to position pos.
 - `remove(item)` – deletes item from the list.
 - `pop()` – removes and returns the last item in the list.
 - `pop(pos)` – removes and returns the item at position pos.
 - `index(item)` – returns the position of item in the list.
 - `search(item)` – returns the pointer to the first occurrence (if any) of item.
 - `is_empty()` – returns true if there are no items in the list.
 - `size()`

For each class you have to implement the constructor (or more than one if you want to have different variations) and utilities the methods like `__str__`

Exercise 2

Define a class **Library** that contains a collection of objects of the class **Book**.

A **Book** is defined by:

- A title
- An author
- An ISBN code
- Number of pages

A **Library** provide a collection of books and allows:

- Add a new book
- Find a book using the author, title or ISBN as search key
- Remove the books of an author
- Remove the book with a specific ISBN
- Remove the book with a specific title
- Return a collection of the books sorted by
 - Alphabetical order using the title
 - Alphabetical order using the author name

- Number of pages
- Print to screen the content of the library

For implementing the classes **Library** and **Book** you can decide which data structure is more convenient and make your own implementation choices that you should be able to motivate. Each class should have convenience methods to access the single fields (i.e., author, title, etc.), to print to screen an object, and (at least) a constructor.

- Discuss (and implement if you have time) which changes (if any) you would make to your class **Library** if the books in it have to be stored in alphabetical order using the author name (in case of same name use the title as second criteria). What would happen for insertion and removal operations?