

CSCI 241 Data Structures

Project 2: Literally Loving Linked Lists LOL

Your submission for project 2 will be graded according to the following rubric. You will receive a score in each category. Your total project score will be the sum of all category scores. Categories and their weights may change with each project.

The first category deals with the implementation of your `Linked_List` class and your testing code. Specific points values per section are as follows:

Constructors and `__len__`: 5 points

`append_element`: 7.5 points

`insert_element_at`, `get_element_at`, `remove_element_at`: 22.5 points total

`rotate_left`: 5 pts

`__iter__` and `__next__`: 7.5 points

`__reversed__`: 7.5 points

`__str__`: 7.5 points

Test code in main: 7.5 points

The total score for implementation will include all sub-sections. The rubric on the next page shows the specific grading criteria.

Implementation

	Exemplary A	Good B	Satisfactory C	Marginal D	Unacceptable F
Adherence to Specifications	There are no deviations from the specifications. Items are named correctly. Files are named correctly and all files are present.	Most components adhere to specifications. Some issues remain. Examples may include spelling or case of specified names or other minor issues.	Many components meet specifications, but key elements are incorrect. Examples include algorithmic deviations from what we covered in class, or use of built-in Python features instead of custom implementations.	The submission attempts to follow instructions, but fundamental issues persist. There is a clear lack of understanding of the specifications and how they relate to the material.	The submission may be substantially incomplete. It may be missing specified components. It may implement items in ways other than specified or covered. The submission conveys ignorance or illiteracy of specifications.
Functionality	The submission passes all test cases. Output is formatted correctly, with no unspecified output. Main code conditioned to be separate from definitions.	The submission produces the correct result, but minor issues such as formatting remain.	There is evidence that the implementation does not consider problematic cases or special cases. The program may crash on a few test cases.	The implementation is fundamentally incorrect. Multiple cases fail, or the program crashes in multiple scenarios. The submission fails to convey sufficient understanding.	The program crashes on most inputs. It contains errors that imply it could not have run during development. Of the cases that do not crash, many fail with incorrect results.
Performance	The implementation takes advantage of all opportunities to improve performance.	The implementation is generally efficient, but some steps could be taken to improve performance	The implementation misses many opportunities to improve performance. Steps are taken without regard to their cost.	The implementation is naive with respect to performance. Some sections of code take significantly longer than the algorithms described in class.	The run time interferes with grading. it contains unnecessary loops and/or repetition. If sections of code perform at all, they are extremely inefficient.
Readability	Variable, function, and class names are descriptive. Comments explain sections of code where necessary. Indentation is consistent and deliberate.	Code is well-structured, but names could be more descriptive. Some sections of code are not obvious and could benefit from additional comments.	There is repeated code that should be moved to functions. Names are not reflective of what they reference. Inconsistent white space may be present. Unnecessary special cases are implemented.	Code uses inappropriate features. Names are poorly chosen. Complex implementations are not commented. Repeated code degrades reliability. Functions return values inappropriately.	The implementation conveys a fundamental misunderstanding of basic programming structures. Loops and conditionals are interchanged. Incorrect variables are used for computation. Sections of code are unreachable.
Testing	Incorporates all provided cases and beyond to fully test implementation. Follows examples, including proper exception handling.	Incorporates all provided cases and extends to further test implementation. Follows examples, including proper exception handling.	Incorporates only provided cases. Purpose of some tests may not be clear, and exception handling is not thorough. Shows lack of consideration of provided examples.	Missing some required cases. Purpose of tests is not clear, does not follow examples, and exception handling is not implemented or is implemented incorrectly.	Testing is largely incomplete and missing many required cases. Exception handling not implemented. Tests follow no logical structure or order.
Score /70	<hr/>				

The second category deals with your write-up. Scoring for sub-sections is provided at the bottom of the Writeup rubric.

Writeup					
	Exemplary A	Good B	Satisfactory C	Marginal D	Unacceptable F
Prose Writeup	The writeup is concise and indicates a deep understanding of the material at hand. It integrates and employs references to illustrations, prose explanations, and examples to justify claims. It describes in detail the algorithmic steps that lead to performance variations at current level. The document adheres to length limitations and contains few if any grammatical errors.	The writeup is content-complete and generally correct, but presentation is lacking. It may fail to justify some claims with illustrations or examples. It indicates that the general concepts are clearly understood, but minor improvements in presentation would benefit the reader. Portions of the writeup could be made more concise without sacrificing detail.	The writeup covers the specified material, but in an incomplete way. It draws some incorrect conclusions, or offers incorrect or incomplete justifications. Many portions could be made more concise without sacrificing detail. Minimal effort to distinguish this submission among others is present.	The writeup is not structured as a prose document, or contains only cursory responses to the required items. The writeup appears to have been an afterthought of the implementation instead of an integral component of the project. The prose is unnecessarily verbose. No effort to distinguish this submission among others is present.	The writeup is missing or fails to address the required items. If it is present, it is so brief as to indicate a lack of sufficient time, analysis, and construction of rational argument.
15 pts performance analysis 15 pts comparison Score /30	<hr/> <hr/>				