Dominic Fantauzzo
Ever Sardoth
Arnetia Bishop-Kent
Carlos Madera

Assignment 3 - Group Analysis

1. **Which program did your group decide on?**

   The group selected Dominic's code as the best one.

2. **What advantages do you think that program has over the others? (be detailed).**

The selected code had the following advantages over other options:

- Better documentation. The selected program had the most detailed docstrings and included docstrings for all functions. Notably, the program's docstrings contained everything that was needed to write the code itself. Furthermore, the program contained the most numerous and detailed comments, which made comparisons easier.
- More get/set methods. The selected program had get/set methods for all data members instead of the necessary data members. If the program were to be developed further, this would make coding from the existing classes more intuitive. No need to go back and check methods. The methods also follow a consistent naming pattern.
- No logical errors. Our group adjusted our function naming and deployed a test file to test all options. The selected program was the only one that passed all tests and did not contain a logical error. This is key as functionality is the most important part of a program. Incorrect returns could cause chaos at the library that implements our program. Hence, all the programs were debugged and evaluated for functionality. Other programs encountered the following problems:
    - Logical error with the return_library_item() method: All library books being returned had their location set to "ON_HOLD_SHELF" or not updated at all.
    - Logical error with the check_out_library_item(): The conditional that checked if an item was on hold by another person called the .get_checked_out_by() method instead of the proper get_requested_by()

method. There were also some minor misspellings in the returns that didn't match the ReadMe.
    - Logical error with the return_library_item() method: When an item was returned, and not requested, the location is never updated.
    - Logical error with the return_library_item() method: The method never removes a patron's items, causing them to incur fines for an item forever.
    - Logical error with the return_library_method(): The method did not include branches for the cases when a patron or item cannot be found.
    - Logical error with the check_out_library_item(): If the item being checked out is on hold, this method does not check if the patron requesting the item is the same as the patron attempting to check out. This would cause the requesting patron's book to be given away.

3. **What improvements do you think could be made to that program? (be detailed).**

   As for suggested improvements, there were none related to the logic of the program, but rather related to code readability. Thus, implementing shortcut operators to shorten the code's length was the only significant suggested enhancement. While not directly changing the code, adding more comments to describe the operation of the return and check_out methods is also recommended. For the return method, the comment "# first checks if the book has been requested, then adjusts location" would have been helpful. The experience of reading other people's code really emphasizes the importance of good comments, something we can all improve on.

4. **Did other submissions have any advantages over the one the group decided on, and if so, what were they? (be detailed).**

   Ever and Arnetias's Library class amend_fine() method version uses a shortcut operator that simplifies the code. By replacing two lines of code with the += operator, we can make the code slightly more readable.

   Arnetia's branching structure for the checkout library item method in the Library class checks for " is not" instead of " is" or "==". This approach was unique amongst all the group implementations and makes the code look more clear. Visually, the user is able to look straight down from the conditional statement to

the respective else condition and understand the result. While there is no performance advantage, this format does improve readability when compared to the more scrambled series of conditional statements that other member's used.