

Thomas Holland
String Manipulation program

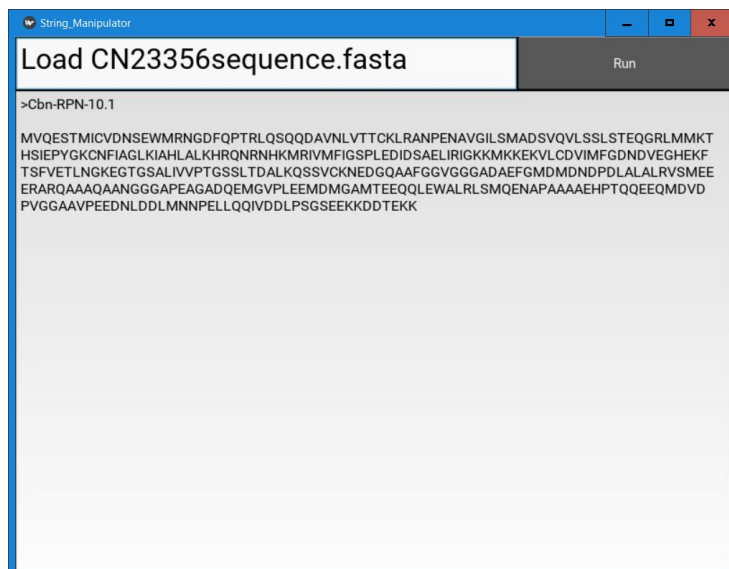
Fixed Issues since review meeting:

Fixed: fixed trim so that it can take only -front or only -back as well as both
Fixed so that file type is correctly saved
Added box shade
Added error handling and logging
Added comments to all major functions

Goal:

Create a Biology background user friendly program that allows for editing of the fasta files and other file types.

Visual of Frontend



Notable features:

All commands and flags are not case sensitive as they are all converted to lowercase before checking them for operations. Queries and other inputs are not converted to lowercase. Another notable feature is that a LOG.txt file logs all errors the user encounters. To generate this file create an error type an incorrect command such as Search ERA -mismatch r .

Commands:

- Load (filename) :This command is used to load in a file. The file can be a fasta or clustal file (extensions .fasta or .aln) and it must be placed in the same folder as this programs executable is located. Must first load in a file to complete certain other commands. Any new file loaded in will replace the initial file present if there is one already loaded in. This command will automatically determine whether the

file is a clustal or fasta file and then import it correctly based on the file type and format.

- Example: Load CN.fa
- Search (query): This command will search the loaded in file (must first run the load command in order for this to work). It will by default search the forward and not the reverse of the file for the entered query. The entered query must be shorter than the file length or it will just say that it is not found. The two flags available are -reverse which will search the reverse as well as the forward versions of the file and -mismatch n which will take any number n (as long as n is lower than the length of the query entered). Mismatch also allows for 1 substitution, insertion or deletion not just a substitution
 - Example: Search AER -reverse -mismatch 2
- Rename addition: Is used to quickly change all of the entries in a fasta file to be in the correct naming format. It will add the addition to the front of all headers in the file. It also has the flag -clear which will remove all info in the line other then the > and the first uninterrupted string. This is useful for poorly made graphing software that keeps unnecessary data for the graphs it produces.
 - Example: Rename Cel_ -clear
 - Result:
 - >NP_001041187.1
 - Converted to
 - >Cel_NP_001041187.1
- Save : Saves the current file to the filename provided. Will determine the file save format based on the extension provided.
 - Examples: Save
 - Save file.fasta
- Trim front n back t: will provide different trim outputs based on trim number provided. If flag -show_all is used, it will show all versions ranging from trimming by 1 to whatever n is.
 - Example: Trim -front 10 -back 2
- Boxshade: Takes in an aligned fasta and returns a boxshade with the correct settings.
- View: allows you to see what file is currently loaded in and what it consists of when seperated.
- Help: prints out all commands, explanations for all of them as well as examples of them in use

Future work:

If this is actually useful then we could look at improving it in ways that would make it more useful for researchers at TCNJ or students. We could also consider shifting the frontend to a web page and having it as a usable portal. Kivy is best for quick prototyping so that would be more work but could be more usable depending on its purpose.

- Concat files (in future could combine fasta and non fasta together into either a clustal or fasta output)
- Format check (check for common formatting errors with a flag to fix automatically common ones)
- Add more buttons so that it is more user friendly and less command line like
- Search for the file via an explorer instead of having the file in the same directory
- A degap feature
- Exporting/converting to other file types via seqret

Citations

Kivy: <https://kivy.org/#home>

Python: <https://www.python.org/>

Selenium: <https://selenium-python.readthedocs.io/>

BeautifulSoup: <https://www.crummy.com/software/BeautifulSoup/bs4/doc/>

BOXSHADE 3.21 written by K. Hofmann and M. Baron.

https://embnet.vital-it.ch/software/BOX_form.html Thompson, J. D., Higgins, D. G., & Gibson, T. J. (1994).

CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. Nucleic acids research, 22(22), 4673–4680. doi:10.1093/nar/22.22.4673

Help commands output for easy testing:

Commands:

help : Displays this message with commands as well as their uses and examples of them

Ex: help

load : Loads the file in for use by this program. file must be located in the same folder as this program. accepts .aln , .fa, and .fasta files

Ex: load test.fasta

save : Saves the file currently in the program to a file. can save as a fasta or aln file.

Ex: save backup.fasta

trim : Trims from the front and back of the sequences in the loaded in file. you can enter in a front value back value or both

Ex: trim -front 10 -back 2

Ex: trim -front 10

Ex: trim -back 2

rename : Renames the headers by adding to the front of the sequences. If clear is set to true it clears everything in the header minus the first word (words defined as character sets split by spaces)

Ex: rename id_

Ex: rename id_ -clear

search : Searches for the included query. Has a reverse flag which also searches in reverse for matches. Has mismatch that allows for a certain number of substitutions insertions or deletions.

Ex: search AER

Ex: Ex: search AER -mismatch 2

Ex: search AER -mismatch 2 -reverse

boxshade : Takes the prepared fasta file (might need to be aligned) that has been loaded in and downloads a boxshade in the same folder as this program generated from the loaded file .

Ex: boxshade

view : Shows the currently loaded file separating the headers and sequences in order to make them easy to look over.

Ex: view