HELP MANUAL

1.0 What does this Programme do?

This programme was scripted to to select a family of protein sequences from a user-defined subset of the taxonomic tree, which would subsequently be processed, and used to plot and determine the level of protein sequence conservation, across species within the user-defined taxonomic group. Following which, these plotted sequences are then scanned with motifs from the PROSITE database, to identify if any known motifs (domains) are associated with these sequences.

2.0 How does this Programme achieve this?

2.1 Performs an esearch (NCBI) based on user input.

- Taxon ID + Protein Name
- Taxon ID + Protein Name + Gene Name

2.2 Displays statistical information on the protein lengths of the proteins found in the aforementioned esearch.

- Option to filter protein sequences >X std deviation above the mean
- Option to filter protein sequences >X std deviation below the mean

2.3 Downloads the FASTA files of the proteins, from the updated dataset (following removal of sequences >x std deviation from the mean).

2.4 Filters for redundancy within species.

- Option to filter protein sequences with a 95% threshold
- Option to filter protein sequences with a 100% threshold

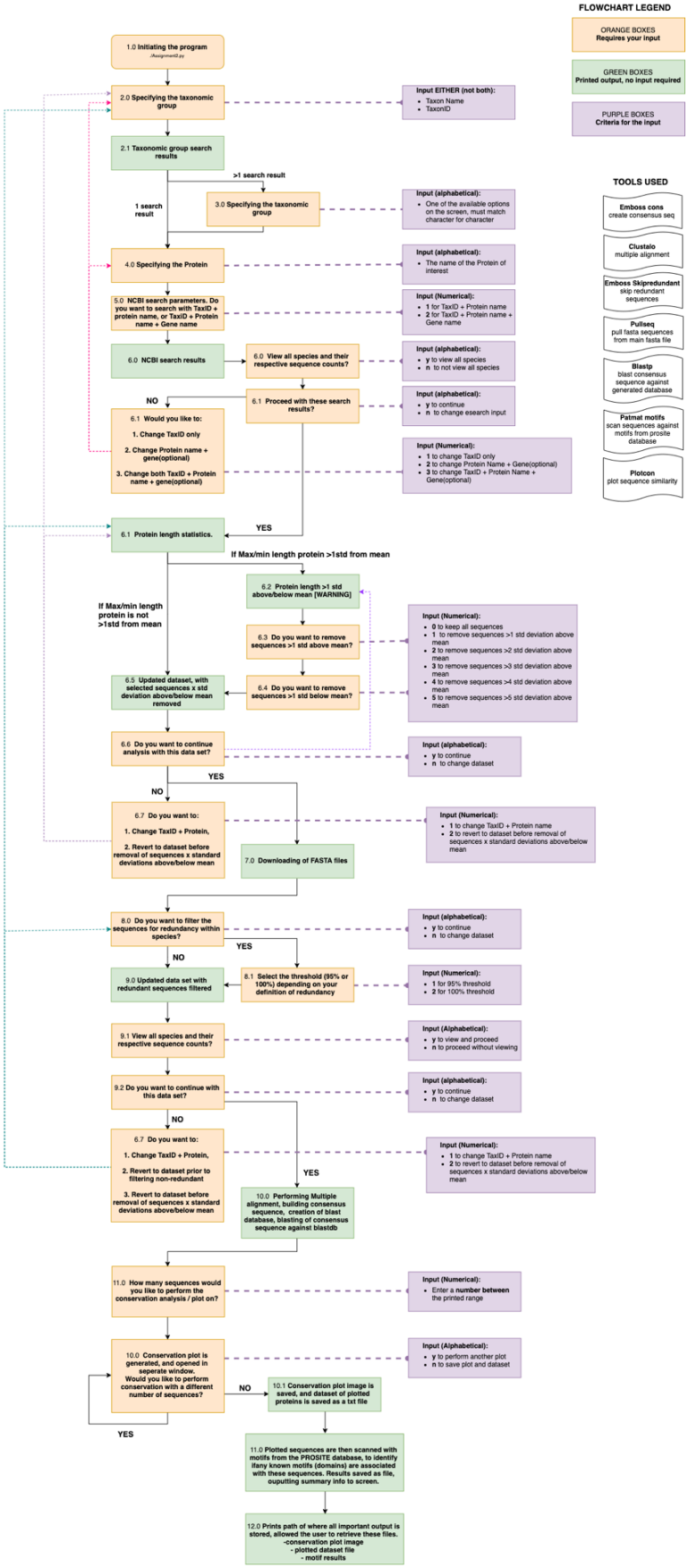
2.5 Performs multiple alignment, builds consensus sequence, creates a blast database from the filtered Fasta file of protein sequences, blasts the consensus sequence against the blast database that was made, to generate a list of proteins from most to least similar.

2.6 Performs conservation analysis, by plotting x most similar sequences on a conservation plot. Outputting the image, showing the protein sequence conservation across species. Image is saved, and dataset of plotted sequences are saved as a .txt file, so the user can see which protein sequences were plotted.

- Option to select between a minimum of 10 sequences, and maximum of 250 sequences, or maximum number of sequences in the dataset, whichever is lower.

2.7 Identifies if any motifs are associated with any of the plotted sequences, using the PROSITE database. Saves the results as a .txt file.

Biologist Flowchart:

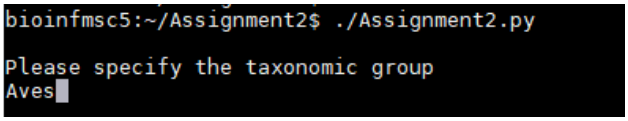
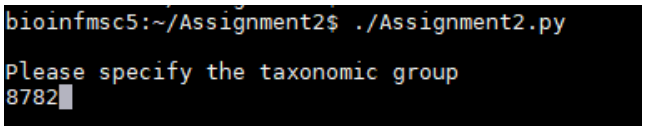


3.0 Example of Program input/output:

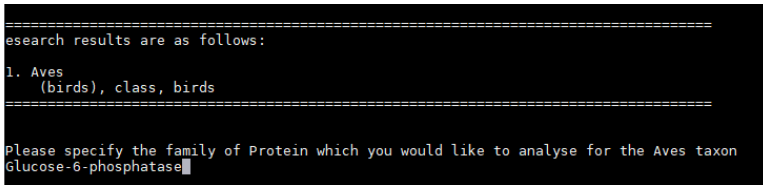
Below is a series of screenshots of both the inputs required from the user, and outputs that the user will be expected to see.

- This example was performed with
  - Taxon : Aves
  - Protein : Glucose-6-phosphatase

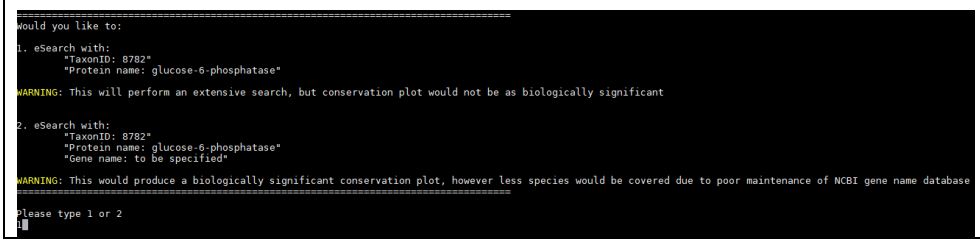
3.1 Running the script, user input required to specify either the Taxon Name / ID

Inputting Taxon Name	Input required
	Taxon name (alphabetical)
Inputting Taxon ID	**Input required**
	TaxonID (Numerical)

3.2 Taxon search output and user input required to specify Protein of interest

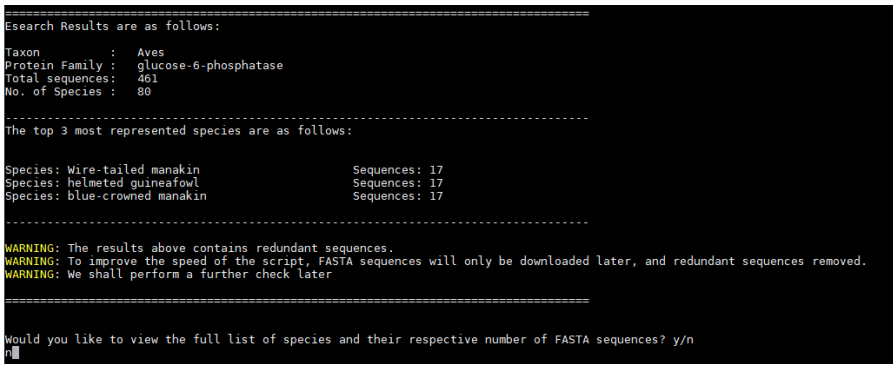
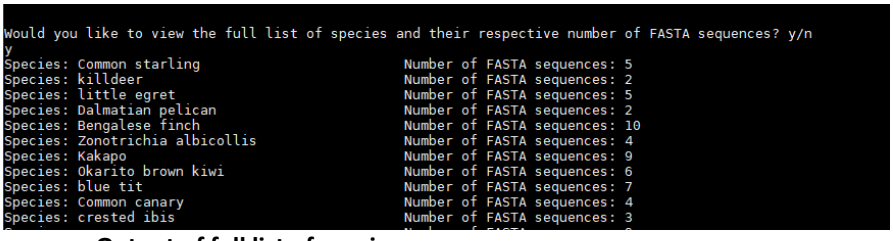
Taxon search output + Protein of interest input	Input required
 ○ **Output** of taxon search: Displays the esearch result of the taxonomy database, displaying the results in GenBank format.	Protein name (alphanumerical)

3.3 User input required to specify esearch parameters

Printed instructions and inputting choice of esearch parameters	Input required
	**1 or 2** (numerical) **1:** esearch w/ Taxon and Protein name [program continues to step 3.4]

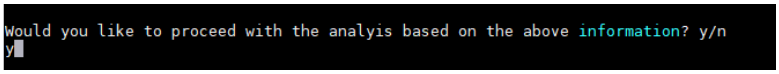
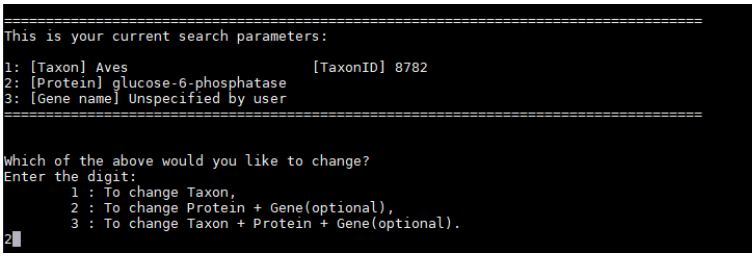
○ **Printed statement:** Informs the user of selection criteria and esearch parameters of each choice, enabling the user to make a sensible decision	**2:** esearch w/ Taxon + Protein name + Gene Name
If input was **2,** program will output the following:	**Input required**
	**Gene name** (alphanumerical) [program continues to step 3.4]

3.4 Esearch (NCBI) results and option for user to view full list of species and their respective sequence counts.

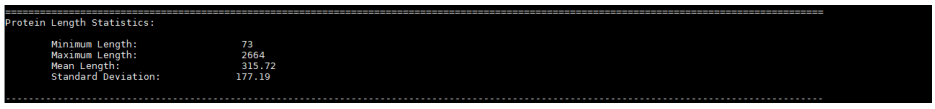
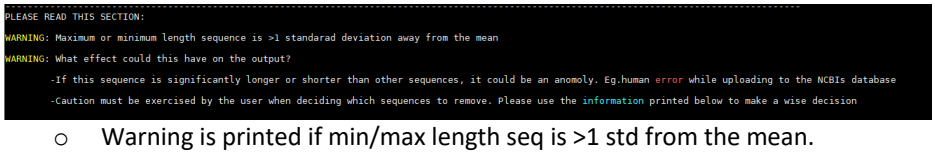
Printed instructions and inputting choice of esearch parameters	Input required
 ○ **Output of esearch w/ taxon + protein name + gene name(optional)** First section shows taxon, protein family, and total sequences found, and unique species that these sequences belong to. Second section shows the top 3 most represented sequences Third section is a warning message, informing the user that the above results contain redundant sequences	**y or n** **(alphabetical)** **y:** displays list of all species and their respective sequence counts. **n:** nothing displayed [program continues to step 3.5]
if input was **y,** program will output the following:	**Input required**
 ○ **Output of full list of species:** Full list of all species and their respective sequence counts shown.	**-NIL-** [Program continues to step 3.5]

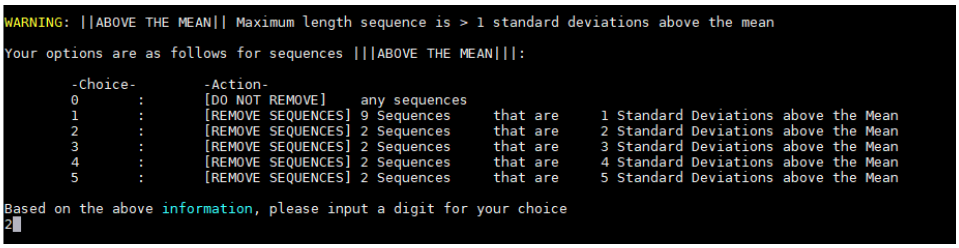
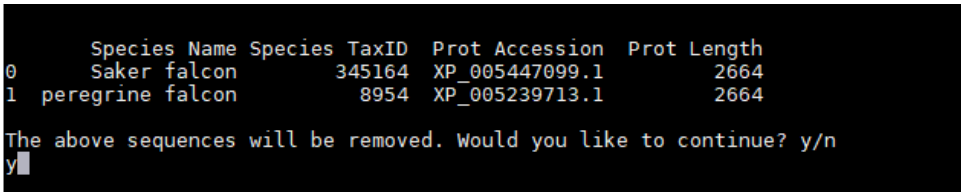
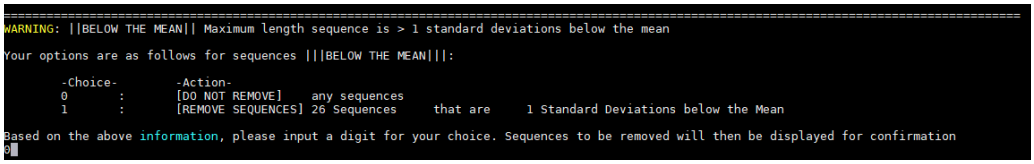
3.5 Option for the user to continue with the displayed dataset, or change the esearch input

Continue with the above displayed dataset?	Input required
	y or n **(alphabetical)**

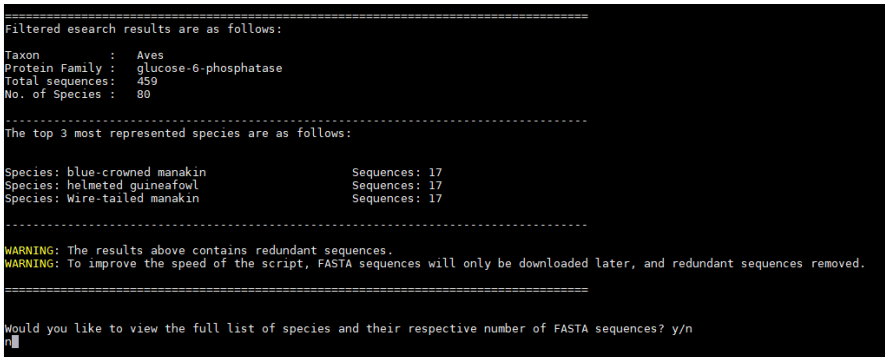
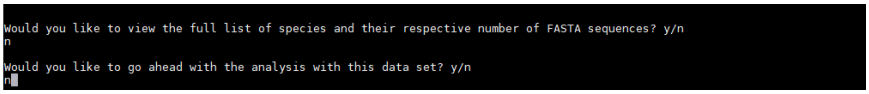
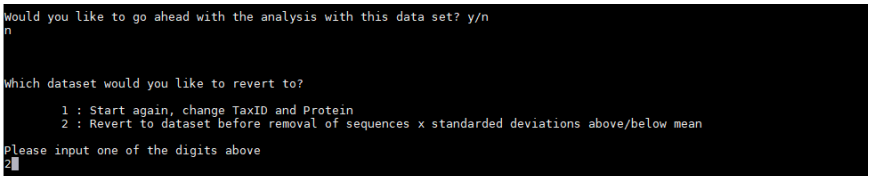
	**y:** [program continues to step 3.6] **n:** options are printed to the screen
if input was **n,** program will output the following:	**Input required**
 ○ **Printed statement:** First section shows current esearch input: Taxon + Protein + Gene Second section informs the user of options, to change Taxon, Protein, Gene.	**1, 2 or 3** **(numerical)** **1:** repeat esearch w/ new Taxon + previous specified Protein [program returns to step 3.1] **2:** repeat esearch w/ previously specified Taxon + New Protein + Gene(optional) [program returns to step 3.2] **3:** repeat esearch w/ new Taxon + new protein + gene(optional) [program returns to step 3.1]

3.6 Output containing protein length statistics, requiring user input to filter sequences >1 standard deviations above the mean

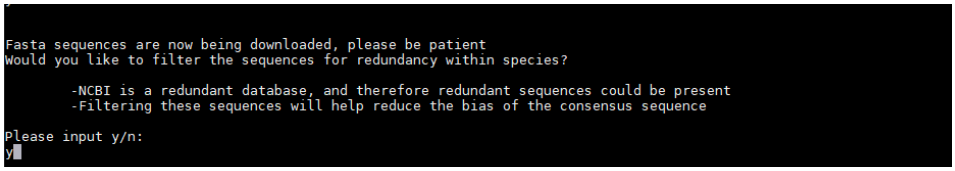
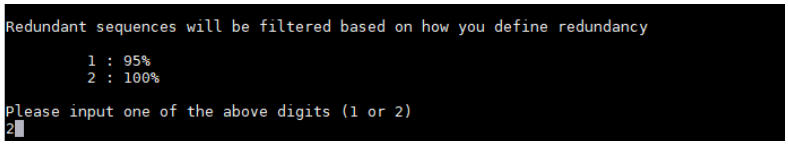
Protein length statistics and user input to decide whether or not to filter sequences	**Input Required**
 ○ **Output of the dataset's Protein statistics** minimum protein length maximum protein length mean protein length standard deviation of the dataset	**-NIL-** if no sequences were >1 std from the mean [program continues to step 3.7]
If min/max protein length is >1 std from the mean	**Input Required**
 ○ Warning is printed if min/max length seq is >1 std from the mean.	**-NIL-**

If maximum protein length >1 std above the mean	Input Required
 ○ **Output of maximum protein length >x std above the mean:** Options are printed onto the screen for the user, at increments of 1 std above the mean, and the respective number of sequences found. Capping at 5 standard deviations above the mean, or when 0 sequences are found.	**0, 1, 2, 3, 4 or 5 (numerical)** **0:** no sequences to be removed **1:** sequences >1 std deviation above the mean **2:** sequences >2 std deviation above the mean etc:
If input was **1, 2, 3, 4 or 5,** the program will output the following:	**Input Required**
 ○ **Output based on user selection of sequences >1 std above the mean to be removed:** Based on the user input of previous step, the selected proteins are printed on the screen, for the user to make an educated decision to remove them from the dataset or keep them.	**y or n (alphabetical)** **y:** the displayed/selected sequences will be removed from the dataset **n:** program returns to previous step, asking user to input his choice
If maximum protein length >1 std above the mean	**Input Required**
 ○ **Output of minimum protein length >x std below the mean:** Options are printed onto the screen for the user, at increments of 1 std below the mean, and the respective numver of sequences found. ○ Capping at 5 standard deviations above the mean, or when 0 sequences are found	**0, 1, 2, 3, 4 or 5 (numerical)** **0:** no sequences to be removed **1:** sequences >1 std deviation above the mean **2:** sequences >2 std deviation above the mean etc:
If input was **1, 2, 3, 4 or 5,** the program will output the following:	**Input Required**
same as above step, no differences!	

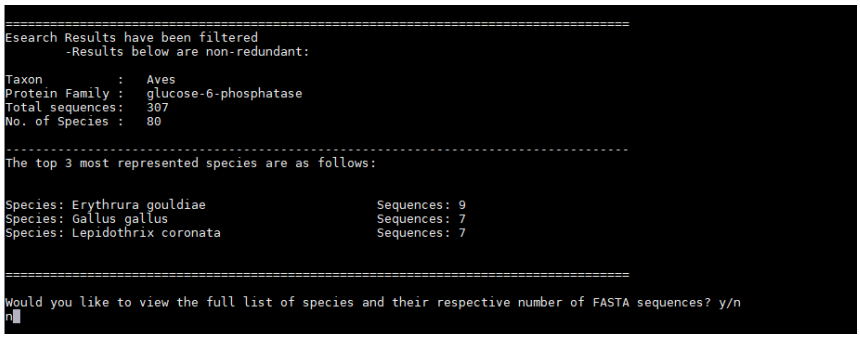
3.7 Outputting the updated dataset, following removal of sequences >1 std from the mean.

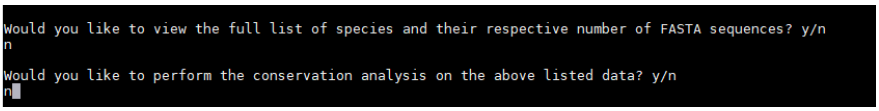
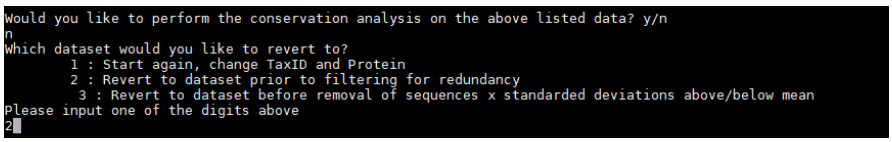
Displaying updated dataset, and user input to show all species and there respective sequence counts	**Input required**
 ○ **Output of updated dataset following removal of sequences >1 std from mean** First section shows taxon, protein family, and total sequences found, and unique species that these sequences belong to. Second section shows the top 3 most represented sequences Third section is a warning message, informing the user that the above results contain redundant sequences User input (y/n) required to decide whether or not to view full list of species and their respective sequence counts. Similar to that in step 3.4.	**y or n** **(alphabetical)** **y:** displays list of all species and their respective sequence counts. **n:** nothing displayed
Checking if user would like to continue analysis with current dataset	**Input required**
 ○ **Printed statement:** Asks the user if he/she wants to continue analysis with current dataset	**y or n** **(alphabetical)** **y:** [program continues to step 3.8] **n:** user decides not to continue analysis with dataset
If input was **n,** program prints the following:	**Input required**
 ○ **Printed statement** Options are printed on the screen for the user, giving him 2 options for changing his dataset.	**1 or 2** **(numerical)** **1:** restart esearch w/ new input [program returns to step 3.1] **2:** revert to dataset prior to removal of sequences >1 std from mean [program returns to step 3.6]

3.8 Input required from the user, to skip redundancies within the species, or not filter for redundancies and continue with the dataset

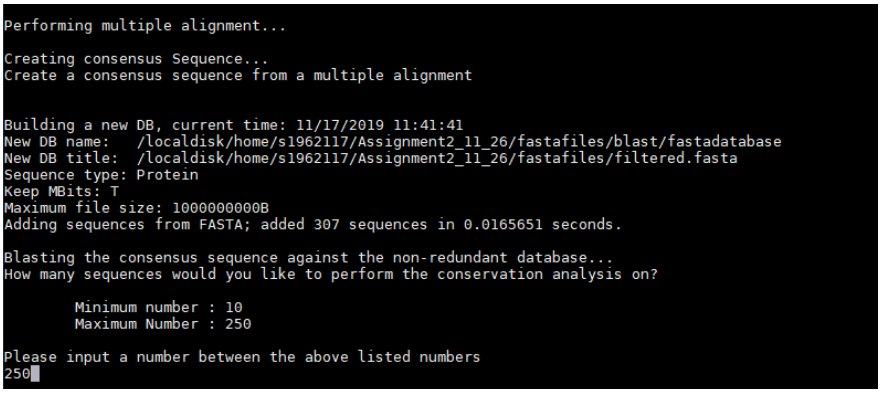
Asking user if he would like to skip redundancies within the species	Input required
 ○ **Printed statement:** Information and instructions are printed on the screen, to help the user make a decision on whether he/she wants to filter for redundancies	**y or n** **(alphabetical)** **y:** user decides to skip redundant sequences **n:** user decides not to skip redundant sequences [program continues to step 3.9]
If previous input was **y,** program outputs the following:	**Input required**
 ○ If user input for previous step was **y**, then the above statement would be printed to the screen. ○ User input (1/2) required, to decide the threshold for the skip redundancy, at either 95% or 100%, depending on the user's definition of redundancy.	**1 or 2** **(numerical)** **1:** skip redundancy within species at 95% threshold **2:** skip redundancy within species at 100% threshold

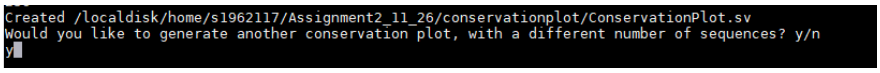
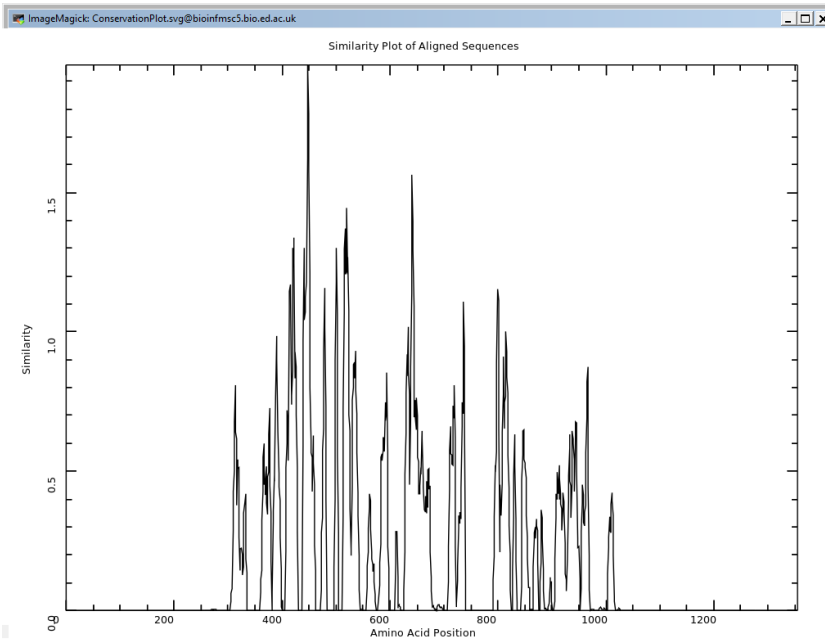
3.9 Printing updated dataset with redundant sequences removed, and checking if user

Printing filtered dataset summary	Input Required
 ○ Updated dataset summary information is printed to the screen ○ User input (y/n) required to decide whether or not to show full list of species in the filtered dataset, and respective number of sequences. 100% threshold skip redundancy	**y or n** **(alphabetical)** **y:** displays list of all species and their respective sequence counts. **n:** nothing displayed

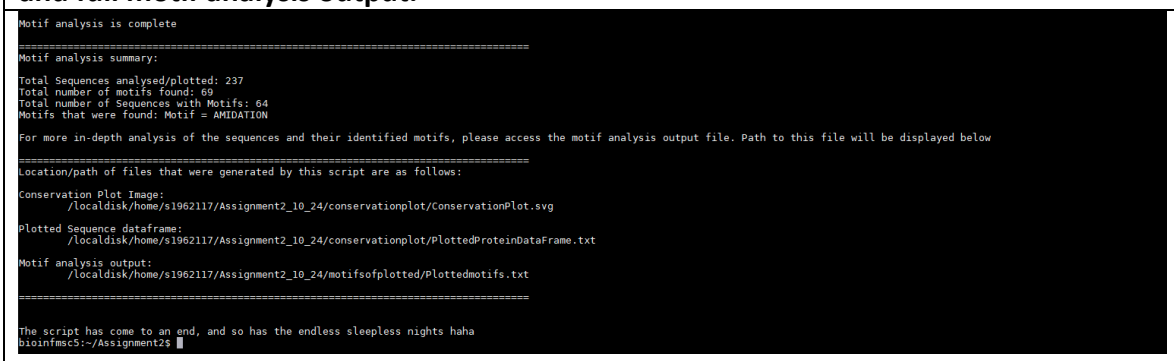
Asking user if he would like to continue with current dataset	Input required
 ○ **Printed statement:** Asking the user if he would like to continue the analysis and perform conservation analysis on the current dataset.	**y or n** **(alphabetical)** **y:** user decides to continue with dataset [program continues to step 4.0] **n:** user decides not to continue with dataset
if previous input was **n**, program prints the following:	**Input required**
 ○ **Printed statements:** Information and instructions are printed on the screen, to help the user make a decision on which dataset to revert to or if he/she wants to start fresh with new esearch input.	**1, 2 or 3** **(numerical)** **1:** **start with new esearch input** [program returns to step 3.1] **2:** revert dataset to the one prior to filtering for redundancy [program returns to step 3.8] **3:** revert dataset to the one prior to removal sequences >1std from the mean [program returns to step 3.6]

3.10 Multiple sequence alignment, generation of consensus sequence, creation of blast database, and blasting consensus sequence against the newly made blast database, followed by plotting of conservation plot based on user input

Multiple sequence alignment, generation of consensus seq, creation of blastdb, and blasting consensus against blastdb	Input required
	**Minimum: 10** **Maximum: 250 or total no. of sequences, depending which one is smaller.** **(numerical)**

○ **Printed statements:** Statements will be printed to the screen to inform the user that: multiple alignment is being performed, consensus sequence is being built, blast database is being built, and lastly, the consensus sequence is being blasted against the newly generated blast database. ○ **Printed statement for input:** Prints the range that the user is able to choose from, how many sequences are to be plotted. The x (user defined) sequences are then selected, these sequences are the most similar sequences to the Consensus sequence	
output of conservation plot and asking user if he wants to repeat conservation plot on different number of seq.	**Input required**
 ○ **Printed statements:** ○ Graph is generated, and path of the graph is printed on the screen. Graph is also displayed in a separate window ○ User is prompted if he wants to repeat the conservation plot	**y or n** **(alphabetical)** **y:** user decides to repeat conservation plot on different no. of seq. [program returns to step 3.10] **n:** user decides not to repeat conservation plot [program continues to step 3.11]
Conservation plot is displayed in a separate window	**Input required**
 ○ **Example of a conservation plot being displayed in a separate window.** This example in particular was for: aves, glucose-6-phosphatase, with 2 sequences >2 std deviations above mean removed, and filtered for redundancy at 100% within species. 250 of the most similar protein sequences were plotted.	**-nil-**

3.11 Performing motif analysis on the plotted sequences, Identifies if any motifs are associated with these sequences, using the PROSITE database. Prints the summary of the Motif analysis, and prints path/location of Conservation plot image, plotted sequence dataframe, and motif analysis result file.

Motif analysis summary and paths to conservation plot, plotted sequence dataframe, and full motif analysis output.	**Input required**
 ○ **Output of motif analysis for plotted sequences** The first section shows the summary of the motif analysis; Total sequences analysed/plotted, number of motifs found, and total number of motifs found, as well as the names of the motifs that were found. For further in-depth analysis, user is provided the path of the motif analysis output, which he can delve further into. ○ **Printed statemen:** Prints the file locations/path of conservation plot, plotted sequence dataframe, and motif analysis output	**-NIL-**

4.0 Important tips so as not to crash the program/script:

- Patience is key, **please do not input anything into the terminal while the script is running tasks in the background.** Prompts for input are extremely self-explanatory and obvious, as long as the user reads the prompts.
- If **esearch result returns with results >1, do not proceed with the analysis.** The script prompts the user to review the dataset/dataframe nearly every time it is altered, giving the user multiple options to revert the dataframe to a prior one, or even perform esearch with different input/parameters.
- If **esearch generates >10,000 results,** the script is designed in such a way for the user to curate and filter sequences, and if the user is not happy with the dataset, to revert the dataset or perform esearch again with different input. Only late into the script is fasta files downloaded. This helps speed up the script, by not having to continuously redownload fasta files and full sequences. If the esearch generates >10,000 sequences, please do not go ahead with the script as I can assure you that you'll be celebrating 5 birthdays before clustalo is done with the multiple alignment.
- **Although error traps are present at every input, please do read the input criteria, as prompts have been made as detailed as they can be.**
- **Output folder is generated in the home space with the hour and minute**, if the user decides to terminate the script immediately after the output folder is generated and runs the script within the same minute, there is a possibility that the file will fail to create as a minute may not have passed yet, thus same folder name (could be bypassed if I had named the folder with the seconds, sorry about that)

Maintenance Manual (competent Python3 code-writer)

5.0 Function Flowchart: