

Using GREP to Find and Count Specific DNA Sequences at the Command Line

Introduction: Grep is a command-line tool that is used to search for a particular string of characters. It gives you the line in a file containing the string that you are looking for. It can print the results to the screen or save them in a new file.

-Finding a primer sequence and saving it to a new file:

```
grep -s 'TAACTTCAGGGTGACCAAAAATCA' query_file.fasta > output1.fasta
```

This command looks for the sequence in question in the file query_file.fasta and saves it to output1.fasta.

The **-s** option in grep is used to suppress error messages about nonexistent or unreadable files. When you use **-s** with grep, it will silently ignore these errors instead of displaying them.

-Bringing the previous line with the query line into a new file: Adding “**-B 1**” lets you bring the previous line with the line containing the string in question. This is useful for getting both the DNA sequence and the header line for FASTA files.

```
grep -B 1 -s 'TAACTTCAGGGTGACCAAAAATCA' query_file.fasta > output1.fasta
```

-Counting with grep: Grep can also be used to count. For example:

```
grep -c 'TAACTTCAGGGTGACCAAAAATCA' infile.fasta
```

Counts how many of these sequence strings appear in infile.fasta.

-Searching for more than one pattern: You can also use grep to find a set of patterns in the same command. grep will print the line containing any one of those patterns you specify. For this, run it as follows:

Any one pattern of the three (OR):

```
grep 'pattern1|pattern2|pattern3' FILENAME
```

All three patterns (AND)

```
grep 'pattern1' FILENAME | grep 'pattern2' | grep 'pattern3'
```

-Note that in the OR example, | stands for or while in the AND example, it pipes the output from one command to another.