Celine Al-Noubani

Programming for Bioinformatics 7200:

Exercise 4

**Instructions**  
Modify the find\_homologs.sh script you wrote last week to carry out the following steps:

1. Find putative homologs of query amino acid sequences in FASTA format (same as last week's instruction - i.e., what your current script does)
2. Use Bash loops and Bash conditional statements (no awk) to identify which genes in a BED file contain the identified homologous histidine kinase domains. (Hint: if a gene contains a domain, then the location of the domain will be entirely within the boundaries of the gene, inclusive of the gene start and stop position)
3. Write an output file containing the unique gene names which your script identified as containing predicted HK domains.

Usage of your script should be: homolog\_identify.sh <query.faa> <subject.fna> <bedfile>  
<outfile>  
Your submission must contain the following:

1. Upload your script as a .sh file named "gtusername.sh" where gtusername is your Georgia Tech username.
2. Upload a submission sheet showing the following:
   1. A screenshot showing the execution of your script in a loop on the 4 genomes provided. Your loop should run your script on each genome and its associated BED file and output the result to a file with a distinct and informative name.
   2. A screenshot of your terminal showing the number of genes identified for each organism. It must be clear in the screenshot which organism each file refers to (i.e., name them informatively in your loop

A black screen with white text

Description automatically generated

Contents of script:

#!/usr/bin/env bash

query=$1

subject=$2

bedfile=$3

outfile=$4

# Make tmp files

blast\_tmp=$(mktemp)

gene\_tmp=$(mktemp)

# Run BLAST (only keep hits with >30% id & 90% length)

tblastn -query $query -subject $subject -outfmt '6 std qlen' \

| awk '$3>30 && $4>0.9\*$13' > $blast\_tmp

# Loop over BLAST hits

while read \_ blast\_seqid \_ \_ \_ \_ \_ \_ blast\_start blast\_stop \_

do

# Loop over BED file

while read bed\_seqid bed\_start bed\_stop gene orientation

do

# Check if we're on the same contig and the BLAST hit overlaps w/ the BED feature

if [[ $blast\_seqid == $bed\_seqid && $blast\_start -le $bed\_stop && $blast\_stop -ge $bed\_start ]]

then

echo $gene >> $gene\_tmp

fi

done < $bedfile

done < $blast\_tmp

sort -u $gene\_tmp > $outfile

rm $blast\_tmp

rm $gene\_tmp

echo "Number of homolog matches in $bedfile: "$(wc -l < $outfile)""

A screenshot of a computer screen

Description automatically generatedA screen shot of a computer

Description automatically generated