Programming for Bioinformatics
BIOL 8803 B
October 19th, 2015

This week we begin the noble study of Perl. Perl concepts and commands for this week:

scalars

arrays

`shift`, `unshift`, `pop`, `push`

`split`, `join`

hashes

`strict`

`if`, `elsif`, and `else`

`while`, `for`, and `foreach`

streams

`open`, `close`

`chomp`

`@ARGV`

1.) Write a script that will get numbers from the user using STDIN and add them to an array. If the numbers are positive, add them to the back of the array. If the numbers are negative, add them to the front of the array. Stop if the user enters 0. Print the array in the end, with the values separated by dots. Also print the sum of the numbers entered.

2.) Download the RepeatMasker table for the hg19 version of the human genome using the table browser on the UCSC genome browser. The RepeatMasker table is under the 'Varation and Repeats' group. Name this file as **RepeatMasker**. There are three columns in the RepeatMasker table which classify the repeat: repClass, repFamily and repName. Using Perl, count the occurrences of every repName, repFamily and repClass in the STDOUT. Print the results in some sort of pretty table, *i.e.* format the results in some visually pleasing way.

3.) Write a script to summarize an input BED file (download the UCSC gene table from table browser in BED format. Name this file as **ucsc.bed**. Your BED file should contain at least the strand information along with the chr, start and stop), the name of which you should take from the command line, including:

a.) The total number of entries in the file

b.) The total length of the entries in the file

c.) The number of entries on either strand

d.) The longest entry

e.) The shortest entry

f.) The average and standard deviation of the length

g.) Put it somewhere on your PATH

4.) A real research example with hash
Navigate to ftp://hgdownload.cse.ucsc.edu/goldenPath/, go into hg19 -> database. This is the place where you'll see a bunch of flat files that make up the databases for the UCSC genome and table browser. We are going to use two of the files from here -> knownGene and kgXref to find coordinates of some genes of interest. Name the files as **knowGene** and **kgXref**

With the assignment you'll find a file named InfectiousDisease-GeneSets.txt

Your objective is to write a script that reads in these three files in a specific logical order and spit out the coordinates for the genes in InfectiousDisease-GeneSets.txt file.

The filenames will be taken as commandline arguments in the order -> knownGene kgXref InfectiousDisease-GeneSets.txt

NOTE: It can happen that certain genes are absent in the kgXref table which is ok, this inconsistency is due to discordance in the update date of the table and the GeneSets file. But there shouldn't be a lot of such cases.

5.) Finish up last week's pipeline if you haven't. Pipelines are ~58% of bioinformatics.

Deliverables
Code:

1. week8_1.pl for question 1)
2. week8_2.pl for question 2)
3. week8_3.pl for question 3)
4. week8_4.pl for question 4)