Warmup

CS181, Fall 2016

Out: Sept. 9
Due: Sept. 20, 11:59 PM

The first two exercises are selected or adapted from Rosalind (http://rosalind.info), an online collection of bioinformatics problems for beginners.

For each problem, write a program that outputs the solution. This is a homework to help you get your feet wet in your programming language of choice for this class.

To hand in, log into a department linux machine. Put all of the files that you want the TAs to see into a directory. Then navigate to that directory and type `cs181_handin warmup`. This will recursively hand in your entire directory. You should receive an email confirmation of your handin.

Specifications: To facilitate anonymized & automated grading, each of your solutions must be accompanied by a shell script. Make sure each problem is able to output the correct result, using the shell script provided. Each problem has a specification section that describes how it should be run. Also make sure your code works on the department machines. Please come to TA hours if any part of this paragraph is unclear to you.

To grab the support code, run the command `cs181_setup warmup`. For this project, we will provide you with stencil shell scripts.

Problem 1: Hamming Distance

(Rosalind: HAMM) Given two strings s and t of equal length, the *Hamming distance* between s and t, denoted $d_H(s,t)$, is the number of positions where s and t differ.

For example, the Hamming distance between the pair of strings `GAGCCTACTAACGGGAT` `CATCGTAATGACGGCCT` is 7.

Given: Two DNA strings s and t of equal length.

Return: The Hamming distance $d_H(s,t)$, printed to standard output.

Specification: `sh hamming.sh STRING1 STRING2`

```
> sh hamming.sh GAGCCTACTAACGGGAT CATCGTAATGACGGCCT
7
```

Problem 2: Reverse Complement

(Rosalind: REVC) In DNA, nucleotides `A` and `T` on opposite strands form a Watson-Crick base pair, as do nucleotides `C` and `G`. We say that `A` and `T` are *complementary* nucleotides, and `C` and `G` are complementary nucleotides. By convention, the string of nucleotides representing one strand of a DNA molecule is written in the $5' \to 3'$ direction, the direction of DNA synthesis. Thus, the string representing the other (complementary strand) is the *reverse complement* string.

The reverse complement of a DNA string s is the string s^C formed by reversing the symbols of s, then taking the complement of each symbol.

For example, the reverse complement of `GTCA` is `TGAC`.

Given: A DNA string s.

Return: The reverse complement s^C of s, printed to standard output.

Specification: `sh reverse.sh STRING`

```
> sh reverse.sh GTCA
TGAC
```

***Note**: Your implementation must not contain any usage of a built-in `reverse()` function on strings.*

Problem 3: Counting *k*-mers

When analyzing DNA sequences, it is often important to ask ourselves what properties we expect a "random" DNA sequence to have. If we assume, for example, that DNA strings are generated by independently drawing nucleotides from the set $\{A, C, G, T\}$ uniformly at random in succession, what do we expect the resulting string to look like? We'll refrain from doing any statistical analysis in this problem. Instead, we will simply count the number of distinct k-mers in a gene for a range of k values. (A "k-mer" is simply a string of length k.)

For example, `AAATC` has three distinct 2-mers: `AA`, `AT`, and `TC`.

Given: A file, `Tthermophilus.txt`, containing the sequence of a real gene.

Return: The number of distinct k-mers in the gene for each of $k = 1, \dots, 10$, printed to standard output. Print each count on its own line.

Specification: `sh kmers.sh Tthermophilus.txt`

```
> sh kmers.sh Tthermophilus.txt
4
16
...
```

As an optional exercise, think about the following. You won't be graded on these questions, but feel free to post your thoughts on Piazza!

- How many possible distinct k-mers are there for each value of k?
- For which values of k do you fail to observe all possible k-mers in the gene?
- Generate a large number of random DNA strings of length 1098, which is the length of the given gene. (Use the definition of "random" given above.) Use these strings to compute empirical distributions on the counts of distinct k-mers for each value of k. How do the counts of distinct k-mers from the real gene sequence compare to the empirical distributions?
- What does this tell you about our model of random DNA sequences?
- The file we gave you happens to contain the sequence of a gene from *Thermus thermophilus*, an extreme thermophile. What does this tell you about the nature of the sequence?

Optional reading

Both of the following readings are available on the course page in the *Resources* section.

1. "Molecular Structure of Nucleic Acids: A Structure for Deoxyribose Nucleic Acid" (Watson & Crick, *Nature*, 1953).
2. "The Unusual Origin of the Polymerase Chain Reaction" (Mullis, *Scientific American*, 1990).