Assignment 5: K-mer Index
Computational Genomics

Objectives: Become familiar with using k-mers and hash tables to align sequences to a reference.

Tasks:

1. Accept the assignment at https://classroom.github.com/a/Fx_NP1KI
2. Clone the repository
3. In `src/kmer_idx.py` implement a short-read aligner using a hash table of k-mers.
    a. Implement the index creation in `get_kmer_index`
    b. Implement read alignment in `align_reads` and `align_read`
        i. Read alignment should use the index to get a list of seed hits, and then for each seed hit extend the seed to see if that region of the reference contains a potential hit. Extend alignments until the maximum number of mismatches have been reached.
        ii. For any good extensions, use Smith-Waterman in (`sw.py`) to get the optimal alignment
    c. Print the unique alignments for every read with the reference offset location, the alignment score, and the alignment
4. Experiment with k-mer size and error rates to better understand the relationship between those parameters and the ability to find closely related sequences.
    a. HINTS:
        i. How does the median number of seeds per k-mer change? More specific means fewer seeds, more sensitive means correct seeds.
        ii. Does the forward/backward extension help with runtime? Does it hurt sensitivity?
    b. There are 3 references in the repository. Use the largest `chr22.fa.gz` for your final experiments.
5. Create figures that demonstrate the results of your experiments.
6. Update `README.md` and create a `doc/kmer_index.tex` (or similar) to include your new experiments.
7. Push your final code to GitHub.
8. Submit your final PDF to Canvas.