C4M Workshop (Level 2) Week 2:

1 Lists and Loops

1. We store patient data in two lists of the same length. For example:

   ```
   sex_data =             ["m", "f", "f", "m", "m", "f"]
   ward_data =            [  1,   3,   2,   2,   1,   2]
   length_of_stay_data =  [ 10,   5,   7,   2,   3,   4]
   ```

   Write a function that computes the average length of stay in the hospital for a given sex, in a given ward. The function signature is:

   ```
   def avg_stay(sex, ward, sex_data, ward_data, length_of_stay_data):
           '''Return the average length of stay for patients of sex sex in ward ward

           Arguments:
           sex_data -- a list containing N strings that correspond to patients' sex ("m" or "f")
           ward_data -- a list of N ints that correspond to patients' ward
           length_of_stay_data -- a list of N ints that correspond to patients' legnths of
                                                        stay
          '''
   ```

2. Write a function that returns `True` iff string `s1` starts with string `s2`. For example,

   `starts_with("abc", "ab")` should return `True`

   `starts_with("ad", "ab")` should return `False`

3. Recall that you can estimate the extent to which the data is spread by computing the *standard deviation* of the data.

   The standard deviation of $x_1, x_2, ..., x_n$ can be estimated as $\hat{\sigma}(x) = \sqrt{\frac{\sum_i^n (x_i - \bar{x})^2}{n-1}}$. Write a function to compute the standard deviation of the length of stay in a given ward.

4. Write a function that returns `True` iff a DNA subsequence matches a DNA sequence somewhere. Both the subsequence and the sequence are represented as strings. For example, if

   ```
   seq = "CGGGGAATAGCCCCC"
   subseq = "AATA"
   ```

   then `match_subseq(seq, subseq)` should return `True` since you can match `subseq` to `seq`, but if

   ```
   seq = "CGGGTCGGGCGC"
   subseq = "AAA"
   ```

   then `match_subseq(seq, subseq)` should return `False`.

   Hint: think of what a useful helper function would be that's similar to what we already wrote.