Task 1: A simple dotplot program

Overall aim

To build a program that creates a simple **dotplot** using no graphics but simply printing a matrix out that indicates where two sequences show similarity.

Your program should contain the following functions:

A) A function (called `dotplot`) that takes two sequences and fills a 2D matrix with ones and zeroes, according to whether a position corresponds to two identical residues or two different ones.

Arguments: two strings, one for each sequences

Returns: a 2D matrix filled in with zeroes and ones

NOTE: You should write a separate function to create your own 2D matrix (call the function: `create_mat` that takes two integer arguments and creates a matrix with number of columns and number of rows defined by the two arguments i.e. `create_mat(nrows, ncols)`

Improvement:

It is generally accepted that the `numpy` package should be used for creating and manipulating matrices in python. Find out how to create a 2D matrix in `numpy` and change your code to replace the `create_mat` function with a `create_mat_numpy` function.

B) A function called `print_dotplot` that prints out the dotplot.

Arguments: two strings representing the sequences, one 2D list representing the matrix already filled with zeroes and ones and two characters to be used for printing out "ones" and "zeroes" (note: a common convention is to use an asterisk to indicates ones in the matrix and a white space to indicate zeroes).

Returns: nothing (simply prints out the dotplot matrix)

Note: Use the function `sys.stdout.write` from the package `sys` to print to the standard out stream (screen in this case).

***C)** Dotplots can be represented as heatmaps (especially if real similarity values are used instead of just 0 and 1 values). Write a simple function to plot a heatmap using the `matplotlib`'s `imshow` function to plot your dotplot matrix as a heatmap. If you are not familiar with matplotlib, look it up online. You will need to import the library at the beginning of your program and more specifically pyplot as follows:

```
import matplotlib.pyplot as plt
```

Arguments: a numpy 2D array containing the values of similarity between two sequences and two strings (one for each sequence).

Returns: Nothing

Improvement: Try to change the axes of the plot so that the ticks on each axis are labeled with the characters of each string.

Task 2: Consensus and profiles from ungapped aligned DNA sequences

1) Solve this Rosalind problem on creating consensus and profiles from an alignment of ungapped DNA sequences:

https://rosalind.info/problems/cons/

2) Alter your code to return log-odds scores relative to a background probability for each DNA letter (by default this should be 0.25 for all four letters; you should allow the function to take a different set of values and use those instead).

3) Use the biopython class: Bio.motifs to check your output is correct.

Task 3: Open Reading Frames

Write a program to find all Open Reading Frames in both strands of a DNA string given as input (this is equivalent to Rosalind's ORF task: https://rosalind.info/problems/orf/)

Task 4: The Needleman-Wunsch Algorithm

Write a program that will solve the global sequence alignment problem for two sequences (start with DNA for simplicity) with linear gap penalties using the Dynamic Programming approach suggested by Needleman and Wunsch.

Hint: You can use two matrices (2d arrays), one to keep information on the best way to reach each element and one to use for trace-back.