Q2 & Q3: Sequence assembly

We're going to write a program in sequence\_assembly.py which will take in 3 different TextGrid objects that represent one gene, and output the correct sequence for that gene. Your job is to fill in the functions get\_best\_nucleotide(nucleotide1, nucleotide2, nucleotide3) and assemble\_sequence().

Genomic sequencing, which is a sophisticated way of saying "making expensive machines read DNA", has seen many revolutionary advances in modern times, but is still far from perfect. DNA sequences are read incorrectly in some cases -- once in a while, an A is read as a C, a nucleotide (a letter in the DNA sequence) isn't read at all, etc. One solution scientists have come up with, to get around this inaccuracy, is just to read a given piece of DNA over and over and compare these reads to each other in order to piece together the true sequence and lower the risk that we've misread something. "Piecing back together" the different reads of the same piece of DNA is called [sequence assembly](https://en.wikipedia.org/wiki/Sequence_assembly).

Your job is to write the code to do sequence assembly, for small genomes which we will represent using TextGrids. You will be given 3 sequences (TextGrids) at a time that correspond to the same gene. Each individual sequence will have random mistakes or missing letters, but you will be able to piece them together in order to generate the true sequence and return the solution TextGrid.

We've split this problem into two milestones -- in the first, you'll write get\_best\_nucleotide(nucleotide1, nucleotide2, nucleotide3) , and in the second you'll use your code from the first milestone to write assemble\_sequence().