The goal of this project is to identify peptides from mass spec data. Peptides are short amino acids that represent segments of proteins. A mass spec machine measures the mass of peptides by measuring how long it takes them to fall. It starts with the unmodified peptide, then shears off one amino acid and measures again. It continues to measure the mass of smaller and smaller segments until it is out of material. The experiment's output is a list of floats corresponding to its observed masses. In a perfect world, the first amino acid's mass would be the first value. The second would be the mass of the first and second peptides together, and so on. Mass spec data is noisy and is mostly false positives. There are also false negatives.

Our strategy for identifying the source protein for a peptide is to find all possible masses from a database of known proteins. We will then create a lookup table that maps masses to peptides in our database. These will then be clustered.

I would like you to focus on the method used to create the database from a protein fasta file:

max_mass takes an amino acid sequence, and an ion that is either b or y, and a charge which is 0 if it is singly charged or 1 if it is doubly charged. The mass of a sequence depends on the amino acids it contains and some basic properties of that molecule including what kind of ion it is and its charge. At the we return the mass normalized by its charge.
https://github.com/ryanlayerlab/hypedsearch/blob/main/src/gen_spectra.py#L78

get_data takes a kmer sequence, its position in its parental protein as start and end values, and id of its parental protein id. The start and end are a zero-based close interval. The function returns its mass considering the different ion and charge possibilities.
https://github.com/ryanlayerlab/hypedsearch/blob/main/src/preprocessing/merge_search.py#L47