Question 2: Minimum Edit Distance (MED)

The med.py program computes the minimum edit distance between two strings supplied by command line parameters at runtime [med.py target source]. The main function distance is then called with the two strings and the associated costs for inserts deletes and substitutions. The MED/Levenshtein algorithm used was taken from the assignment specification provided. The provided function initially did not provide visual edit alignments so the functionality was added in.

The block of code represents lines 45-52

As the dist matrix is being populated with the proper distances, the ptr matrix is keeping track from which cell that distance came from. DIAGM represents coming from the upper left corner by character match. DIAGS represents upper left corner as well but by way of substitution. LEFT and UP represent the corresponding direction.

The backtrace nested method lines 55-75

The backtrace is a local helper function in distance that is passed in the last indices of the dist matrix that contain the end Levenshtein distance. Using the ptr matrix created earlier, backtrace as the name implies, backtrace’s the dist matrix creating a list of performed edits by appending to list “path”.

A LEFT in the ptr matrix represents a movement from the right which correlates to a deletion or D. The ptr string UP appends an I indicating an insertion. S and M are for substitution and match respectively. As we are working from the last cell in the matrix backtrace is called recursively and the final path reversed.

The above gives an example output when program is run with the words gamble and gumbo. The path list is omitted in the final version of the program and merely just shown for debugging and accuracy.

Here are a few more examples to show proper expected output.

To print the visual alignment as shown above, lines 92-111 in the distance function is executed.

