The file needlemen.py includes a class called *NeedWunsch*(). This class includes helper methods to calculate optimal global alignment and output the best score for the pairwise alignment of two user-specified sequences. The methods included are documented below

- __init__(seq1,seq2,gap,mismatch,match): Constructor which populates seq1, seq2, gap, mismatch, match. User can set the two sequences by passing in any pairwise nucleotide sequence. An example is included in the main() method of execution where seq1 is initialized to “actcg” and seq2 is initialized to “acagtag”. Similarly, the gap penalty, mismatch and match can be set for the computation by populating gap, mismatch, match.
- BuildTable() : Builds the initial empty table and populates row 1 and column 1 with multiples of the gap penalty
- BuildTraceTable(): Creates empty table to store traceback values
- FillTable(): Iterates from the bottom right corner of the table and calculates the positions of the max scores of its top, left and diagonal neighbors. The max of all three will be filled into the traceTable to compose an optimal path.
- BuildPath(): Build path uses the traceTable to compose a string of moves representing the path that is optimal. It selects arbitrarily from the set of all optimal moves in the case there is more than one.
- GetPairwiseAlignment(): Uses the path calculated in BuildPath() to compose the optimal sequences and calculate score

How To Use:

1. Open needleman.py in an editor of choice
2. Go into def main():
3. Change the gapPenalty variable on line 116 to the desired gap penalty.
4. Change the mismatchPenalty variable on line 117 to the mismatch value
5. Change the match variable on line 118 to the desired match score
6. Change the first and second argument to NeedWunsch() on line 119 to the sequences of choice as strings.
7. Save and close neeedleman.py
8. Use python3 to run needleman.py
9. Observe the output printed to the screen showing the optimal sequence and the score.