Adrian Flannery

Bioinformatics II

Assignment No. 2

Forms:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name: | Main |  |  |  |
| Inputs | Outputs | Calls | Called by | Files |
| String Sequence 1 | ‘Similarity score’ | similarity(String sequence1, String sequence 2) | none | none |
| String Sequence 2 | ‘aligned sequence 1’ | getAlignment(long[][] alignMatrix, String sequence 1, String sequence 2) |  |  |
|  | ‘aligned sequence 2’ |  |  |  |
| Processing: | Drives the program |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name: | p |  |  |  |
| Inputs | Outputs | Calls | Called by | Files |
| String nucleotide1 | ‘match/mismatch score' | none | similarity | none |
| String nucleotide2 |  |  | getAlignment |  |
| Processing: | Compares nucleotides and returns match/mismatch score | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name: | max |  |  |  |
| Inputs | Outputs | Calls | Called by | Files |
| int xLeft | 'max score' | none | similarity | none |
| int xAbove |  |  |  |  |
| int xDiagonal |  |  |  |  |
| Processing: | Compares the three paths and returns the max score | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name: | similarity |  |  |  |
| Inputs | Outputs | Calls | Called by | Files |
| String sequence1 | Long[][] alignmentMatrix | p | main | none |
| String sequence2 |  | max |  |  |
| Processing: | Computes the values of each cell | | |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name: | getAlignment |  |  |  |
| Inputs | Outputs | Calls | Called by | Files |
| Long[][] alignMatrix | String[] sequences | p | main | none |
| String sequence1 |  |  |  |  |
| String sequence2 |  |  |  |  |
| int maxCellX |  |  |  |  |
| int maxCelly |  |  |  |  |
| Processing: | Does the traceback on the matrix and returns the local alignment | | | |

Variables:

Function p:

String nucleotide1: substring that is used to compare nucleotides for match/mismatch score

String nucleotide2: substring that is used to compare nucleotides for match/mismatch score

Function max:

Int xLeft: holds the score from the cell from the left plus the gap score for comparing which path yields the highest score

Int xAbove: holds the score from the cell from above plus the gap score for comparing which path yields the highest score

Int xDiagonal: holds the score from the diagonal plus the match/mismatch score

Function similarity:

String sequence1: holds sequence to be aligned

String sequence2: holds other sequence

Int matchScore: holds a match/mismatch score for filling out matrix

Long[][] alignmentMatrix: holds scores for the alignmentMatrix

Function getAlignement:

Long[][] alignMatrix: same as alignmentMatrix

String sequence1: same as before

String sequence2: same as before

Int maxCellX: holds the x-value of the max score cell

Int maxCellY: holds the y-value of the max score cell

Int currentX: holds an x-value for keeping track of what cell the loop is on

Int currentY: holds an y-value for keeping track of what cell the loop is on

String[] sequences: holds the aligned sequences to be returned

Main method:

String seq1: sequence from input

String seq2: sequence from input

Int maxCellX: holds the x-value of the max score cell

Int maxCellY: holds the y-value of the max score cell

Long maxScore: holds the score of the maximum cell

Long[][] alignMatrix: alignment matrix

String[] sequences: holds the aligned sequence