Universität Bielefeld
Technische Fakultät

AG Genominformatik
Prof. Dr. Jens Stoye
Dipl.-Inform. Klaus-Bernd Schürmann

Foundations of Sequence Analysis
Winter semester 2003/2004

Exercises

Exercise 3, Discussion: 24/11/2003 and following dates.

1. (a) For all sequences u and v, $lcp(u,v)$ denotes the longest common prefix of u and v. Given two sequences x and y of lengths m and n. Develop an $O(mn)$ time algorithm, computing the
$m \times n$-matrix M^{lcp} such that:
$$M^{lcp}_{i,j} = |lcp(x_i \dots x_m, y_j \dots y_n)|$$
for all $i \in [1,m]$ and $j \in [1,n]$. For the sequences $x = acaacacaacac$ and $y = aaacaccacaca$ the algorithm is supposed to return the following matrix:

```
1 1 3 0 2 0 0 3 0 3 0 1
0 0 0 2 0 1 2 0 2 0 2 0
2 5 1 0 1 0 0 1 0 1 0 1
1 1 4 0 2 0 0 5 0 3 0 1
0 0 0 3 0 1 4 0 4 0 2 0
1 1 3 0 2 0 0 3 0 3 0 1
0 0 0 2 0 1 2 0 2 0 2 0
2 5 1 0 1 0 0 1 0 1 0 1
1 1 4 0 2 0 0 4 0 3 0 1
0 0 0 3 0 1 3 0 3 0 2 0
1 1 2 0 2 0 0 2 0 2 0 1
0 0 0 1 0 1 1 0 1 0 1 0
```

(b) For all sequences u and v, $lcsuf(u,v)$ denotes the longest common suffix of u and v. How can the algorithm of 1(a) be used to compute the $m \times n$-matrix M^{lcsuf} such that:
$$M^{lcsuf}_{i,j} = |lcsuf(x_0 \dots x_i, y_0 \dots y_j)|$$
for all $i \in [1,m]$ and $j \in [1,n]$.

2. Implement an object-oriented program according to the Smith-Waterman algorithm for the computation of local optimal alignments. The program should be called *swalign* and have four parameters:

int indelscore - the score of an insertion resp. deletion
NNMatrix scorematrix - a symmetric $n \times n$-matrix holding the replacement scores
String u - first input sequence
String v - second input sequence.

The program should be divided in subroutines for

- building the matrix (*swBuildMatrix*),
- finding the maximal entry (i,j) in the matrix (*swFindMaxScore*),
- computing the optimal local alignments by backtracking on a maximizing path starting at (i,j) (*swCompOptAligns*).