

1	2	3	Σ

Blatt 7

(Abgabe am 19. October 2015)

Exercise 1 - *Smith Waterman today*

The 30 years old Smith-Waterman (SW) algorithm is one of the fundamental algorithms in bioinformatics. This dynamic programming (DP) algorithm allows in a biological sense for example the detection of Domains in a protein or a Protein in a larger sequence. But the algorithm has to face the problem of complexity of $O(n \times m)$. This complexity is getting a problem when starting to compare large sequences or even whole genomes. Still today has this algorithm a high importance in bioinformatics, which can be seen in the following examples. The classical example for today's usage of the Smith Waterman algorithm is the BLAST algorithm. Which uses algorithm for seed refinement. Thereby the algorithm is accelerated with vectorization process. Basically this is an improvement on the hardware level of a computer, which can be addressed by the software for parallelizing the alignment. As a result, the Smith-Waterman algorithm is accelerated by a factor of 10 in comparison to the standard algorithm Novoalign (<http://novocraft.com>). Also an advantage in complexity matters is constraining the algorithm around seeds from the seeding step. [Li, H., Homer, N.] Parallelizing is general a very modern topic for Smith-Waterman Algorithms, like in the next paper, were [Chen, B. and Xu, Y. et al]

Exercise 2 - *DP Algorithm for finding motifs*

```

for i = 0 to length(X)
  if X[i] == Y[0]
    Score = 0
    for j = 0 to length(Y)
      if (X[i+j] == Y[j]) Score += 1
    if (Score > T)      print(motif found at position i)      i += length(Y)

```

Exercise 3 - *Programming assignment: Needleman-Wunsch alignment*

One task was to handle command line options. We wrote a new class to have a modular and easy way to provide this functionality. The java package *args4j* has everything we need. So we used that package to write our own option handler class. The code can be found in *CommandLineParser.java*. We did the same with gap penalties. A gap penalty class gives one the advantage of modular gap handling. If a gap penalties are needed, just reuse this class.

A new member function of our NeedlemanWunsch class takes care of file writing. If a user wants the resulting alignment in a FastA file, this function takes care of it.

References

- [Li, H., Homer, N.] Li, H., Homer, N. (2010). A survey of sequence alignment algorithms for next-generation sequencing. *Briefings in Bioinformatics*, 11(5), 473–483. <http://doi.org/10.1093/bib/bbq015>
- [Chen, B. and Xu, Y. et al] Chen, B. and Xu, Y. and Yang, J. and Jiang, H. (2010). A New Parallel Method of Smith-Waterman Algorithm on a Heterogeneous Platform. Springer. *Algorithms and Architectures for Parallel Processing*. 79–90