Advanced Algorithms in Bioinformatics (P4)

Sequence and Structure Analysis

Freie Universität Berlin, Institut für Informatik
Prof. Dr. Knut Reinert, Sandro Andreotti, Markus Bauer, Andreas Döring
Sommersemester 2008
1. Review, 2008-05-21

Name, Vorname	Matrikelnummer

Zur Bearbeitung des Reviews stehen Ihnen 50 Minuten zur Verfügung. Jeder Punkt entspricht in etwa einer Minute.

Geben Sie auf diesem Titelblatt und auf allen eventuell zusätzlich abgegebenen Blättern ihren Namen und ihre Immatrikulationsnummer an.

Schreiben Sie ihre Lösungen direkt auf die entsprechenden Aufgabenbögen. Sollte dort der Platz nicht ausreichen und Sie weitere Blätter benötigen, vermerken Sie dies bitte, damit wir auch den Rest ihrer Antwort finden und bei der Bewertung berücksichtigen können. Am Ende des Reviews sind sämtliche Aufgabenblätter wieder abzugeben.

Ergebnis:

Aufgabe	maximal	erreicht
1	12	
2	12	
3	13	
4	13	
$\sum$	50	

Exercise 1. 4+2+3+3=12 Punkte

1. Explain the idea of the Horspool algorithm for exact pattern matching.

2. How does the performance depend on the alphabet size? Explain why?

3. Give an example of pattern P and text T where the Horspool algorithm has runtime $O(|P||T|)$ and argue why.

4. For the chosen P and T, will the application of the Wu-Manber algorithm with a block size of 2 reduce the number of character comparisons?

Exercise 2. 12 Punkte

In the following, C is a dynamic programming matrix computed for two strings (say, P and T) using the edit distance. Assume that you already know that the following assertions hold for all indices i and j such that $i + j \leq k$, for some given number k.

$$\begin{array}{rrcccl}
\text{horizontal adjacency property} & \Delta h_{i,j} & = & C_{i,j} - C_{i,j-1} & \in & \{-1, 0, +1\} \\
\text{vertical adjacency property} & \Delta v_{i,j} & = & C_{i,j} - C_{i-1,j} & \in & \{-1, 0, +1\} \\
\text{diagonal property} & \Delta d_{i,j} & = & C_{i,j} - C_{i-1,j-1} & \in & \{0, +1\}
\end{array}$$

Now prove that the *horizontal property* also holds for i, j such that $i + j = k + 1$.

Exercise 3.   3+3+4+3=13 Punkte

1. Complete the pigeonhole Lemma:

   Lemma 1. *Let Occ match P with k errors, $P = p^1, \ldots, p^j$ be a concatenation of subpatterns, and $a_1, \ldots, a_j$ be nonnegative integers such that $A = \sum_{i=1}^{j} a_i$. Then* ...

2. Explain how the pigeonhole principle is used for filtering.

3. You want to search the pattern `searchme` with 3 errors. Build the verification tree as it is used by the PEX algorithm. For each node give the parameters (*from, to, error*).

4. Given two strings of length 33 that match with at most 4 errors. How many common 5-grams do they at least share.

Exercise 4. 5+3+5=13 Punkte

1. The Manber Myers algorithm uses five arrays for the construction of the suffix array for a text T in time $O(|T| \log |T|)$. Explain the role of the arrays suftab, sufinv, count, bh, and b2h during the construction algorithm.

2. Remember the algorithm of Kasai et al. Below you find a correct suffix array (`suftab`) and a would-be height array (`lcptab`) with exactly *one* error. Where is the error in the height array, and why cannot the height array be correct as it is written below? You do not need to know the underlying string in order to answer this question.

i	suftab[i]	lcptab[i]
0	5	-
1	3	2
2	0	2
3	6	1
4	4	0
5	2	2
6	1	1
7	7	0

3. Amortized analysis: Given a Stack with three possible operations push, pop and multipop. The operations push and pop add and remove exactly one element from the stack. The operation multipop is called with a parameter k and returns the last $\min(k, S)$ elements of the stack where S is the actual number of elements in the stack. Show that the amortized cost of any operation is $O(1)$.