Übungen zur Vorlesung Sequenzanalyse

Universität Bielefeld, SS 2017
Prof. Dr. Jens Stoye · M.Sc. Tizian Schulz

https://gi.cebitec.uni-bielefeld.de/teaching/2017summer/sa

Übungsblatt 7 vom 20.06.2017
Abgabe 29.06.2017

Aufgabe 1 (Implementierung des *q*-gram Index) (5 Punkte)
Schreibe ein kleines Programm, das den q-gram Index eines Strings berechnet. Verwende bei der Berechnung die geschickte Variante, die im Skript auf Seite 59 unten beschrieben ist. Als Eingabe soll das Programm einen String und ein bestimmtes q entgegen nehmen und den q-gram Index ähnlich einer Tabelle mit den Spalten *q-gram*, *Positionen* und *Anzahl Vorkommen* ausgeben. Verwende eine Programmiersprache, die mit deinem Tutor abgesprochen ist und sende ihm dein Programm per Email zu.

Aufgabe 2 (Suffixbäume) (8 Punkte)

1. Beschreibe zwei Anwendungen aus der bioinformatischen Praxis, für die man einen Suffixbaum verwenden kann.
2. Gegeben ist der String $s\$ = \texttt{abbabcacbb\$}$, wobei $\$ < a < b < c$.
   (a) Zeichne den Suffixbaum für s, sortiere dabei die Blätter lexikographisch.
   (b) Beschrifte die Blätter mit dem Start-Index des zugehörigen Suffixes in s. Die Indizierung beginnt bei 1.
   (c) Beschrifte jeden Knoten mit der Anzahl der unter ihm liegenden Blätter.
3. Analysiere die *worst-case*- und *average-case*-Laufzeit des WOTD-Algorithmus in eigenen Worten.
4. Zeige, dass der Speicherverbrauch eines Suffixbaums linear bezüglich der Eingabe ist.

Aufgabe 3 (Verallgemeinerter Suffixbaum) (4 Punkte)
Gegeben seien die Strings s=`CTTC` und t=`TTCC`.

1. Zeichne den generalisierten Suffixbaum von s und t (mit $\# < \$ < C < T$).
2. Überlege dir, wie man den längsten gemeinsamen Substring von s und t finden kann und gib seine Vorkommen an.
3. Formuliere allgemein, wie man längste gemeinsame Substrings zweier Strings im generalisierten Suffixbaum finden kann.
4. Mache dir Gedanken darüber, wie man das längste palindromische Teilwort in einem Wort mit einem generalisierten Suffixbaum finden kann. Verwende dazu das Beispiel $x =$ `BANANAS`. Beschreibe deine Idee. Es ist nicht nötig, den Baum explizit zu zeichnen.

Aufgabe 4 (Maximale Repeats) (3+3* Punkte)
Lies den Abschnitt 7.7.3 im Skript über das effiziente Auffinden von maximalen Repeats in einem Text s mit Hilfe des Suffixbaums von s.

1. Stelle den Suffixbaum von $s =$ `GCATATGATACATG` auf.
2. **Satz:** In jedem String der Länge n gibt es höchstens n Teilworte, die maximale Repeats sind.
   Argumentiere unter Berücksichtigung des Suffixbaums, warum diese Aussage korrekt ist. Bedenke: Es stimmt nicht, dass an jeder Position nur ein maximales Repeat beginnen oder enden kann.
3. Für 3* Punkte: Finde alle maximalen Repeats in s unter Verwendung des im Skript geschilderten Algorithmus. Beschreibe dein Vorgehen beim Annotieren des Suffixbaums aus Aufgabenteil 1.