Sommersemester 2006 Abgabe am 02. Juni 2006

Prof. Dr. Stefan Posch, Dipl.-Bioinform. André Gohr, Dipl.-Bioinform. Jan Grau

8. Übung „Algorithmen der Bioinformatik I“

1. In der Vorlesung wurden die Begriffe *maximales Paar* (i, j, k) und *maximaler Repeat (wiederholter Teilstring)* eingeführt und Eigenschaften dazu bewiesen.

   Formulieren Sie eine Hypothese über die maximale Anzahl von maximalen Repeats in einem String der Länge n und beweisen Sie sie (2 Punkte).

2. a) Überlegen Sie sich einen in der Grösse eines Suffixbaums linearen Algorithmus, der für jeden inneren Knoten entscheidet, ob dieser links-divergent ist oder nicht und diesen entsprechend dieser Eigenschaft markiert. (3 Punkte)

   b) Beschreiben Sie eine Methode, mit der alle maximalen Paare eines Strings S der Länge n gefunden werden können. Diskutieren Sie die Korrektheit und Laufzeit dieser Methode. Welchen Speicherplatzbedarf hat Ihr Algorithmus (abgesehen vom Bedarf für den üblichen Suffixbaum)? Modifizieren Sie den Algorithmus so, dass der Speicherplatzbedarf $\mathcal{O}(n)$ nicht übersteigt. (5 Punkte)

   Hinweis: Für jeden inneren Knoten v des Suffixbaums von S sollen maximal soviele Listen erzeugt werden, wie es Buchstaben im Alphabet gibt. Die Liste im Knoten v, die zum Buchstaben x gehört, soll alle Anfangspositionen von Teilstrings des Strings S enthalten, die mit der Pfadbeschriftung von v im Suffixbaum übereinstimmen und als linkes Zeichen ein x haben. Aus diesen Listen können nun die maximalen Paare konstruiert werden.

   c) Was müsste an Ihrem Algorithmus zum Finden aller maximalen Paare geändert werden, wenn nur maximale Paare einer minimalen Länge m gesucht sind? Welchen Einfluss hat das auf die Laufzeitabschätzung? (2 Punkte)

3. Für die Implementierung eines Suffixbaums für einen String muß man sich eine geeignete Datenstruktur für die inneren Knoten überlegen. Ein Kriterium dafür ist, wie die vom Knoten ausgehenden Kanten im Knoten repräsentiert werden. Dabei spielen sowohl der benötigte Speicherbedarf als auch die Zugriffszeit auf eine gesuchte Kante (um sie zu verfolgen) eine wichtige Rolle.

   Wenn Σ das Alphabet aller möglichen Zeichen im String ist, dann gibt es maximal $|\Sigma|$ ausgehende Kanten pro inneren Knoten. Damit können alle Kanten mit einem Feld der Länge $|\Sigma|$ repräsentiert werden (der erste Buchstabe der Kantenbeschriftung wird als Index verwendet). Alternativ können ausgehende Kanten über eine verkettete Liste oder einen balancierten binären Suchbaum im Knoten repräsentiert werden.

   Vergleichen Sie die drei Möglichkeiten hinsichtlich des Zeitbedarfs zum Finden einer Kante und des Speicherbedarfs (ein Verweis auf ein Listenelement bzw. einen Kindknoten benötige genausoviel Speicher wie ein Verweis auf eine Kante). Überlegen Sie sich, in welchen Ebenen des Suffixbaums (obere, mittlere, untere) Sie welche Art der Repräsentation anwenden würden und begründen Sie Ihre Entscheidung. (3 Punkte)

Die theoretischen Lösungen bitte ausgedruckt oder handschriftlich in der Übung abgeben.