Prof. Dr. Knut Reinert
Thimo Wellner
Sascha Meiers

Institut für Informatik
AG Algorithmische Bioinformatik

Algorithmen und Datenstrukturen in der Bioinformatik
Vierte Programmieraufgabe WS 13

Abgabe Montag, 16.12., 15:00 Uhr per SVN

In dieser Programmieraufgabe werden Sie die Suche im Suffixarray parallelisieren und auf größere Texte anwenden.

Aufbau des Suffixarrays Das Aufbauen großer Suffixarrays mit dem naiven Sortieralgorithmus ist nicht sinnvoll. Laden Sie deshalb unter `https://svn.imp.fu-berlin.de/aldabi/WS13/material` den Text, das fertige Suffixarray (binär) und die vorbereitete Einlesefunktion herunter. Letztere können Sie als Ausgangspunkt für ihr Programm verwenden.

Texte einlesen Ihr Programm soll neben dem Suffixarray noch zwei Textdateien einlesen, die die Datenbank und die Liste der Suchwörter enthalten. Beim Einlesen der Datenbank sollen alle Zeilen **ohne Umbruchzeichen** konkateniert werden. Nur dann passen Text und Suffixarray zusammen. Beim Einlesen der Suchwörter soll jede Zeile als einzelner String interpretiert werden.

Suche Erweitern Sie Ihre Binärsuche dahingehend, dass Sie die linke und rechte Grenze (L_p und R_p) aller Vorkommen des Suchworts im Suffixarray finden. Achten Sie darauf, die *mlr*-Heuristik anzuwenden und vermeiden Sie teure copy-Operationen. Suchen Sie dann parallel alle Suchwörter aus der Query-Datei im Suffixarray und geben Sie die Anzahl der Treffer, die das Suchwort im Text hat, durch Komma getrennt aus. Achten Sie bei paralleler Ausführung darauf, dass die Ausgabe in der gleichen Reihenfolge wie die Eingabe geschieht.

Laufzeit messen Messen Sie die Laufzeit ihres Suchprogramms als *wall clock time* mit dem `time`-Befehl[1]. Die Zeit für das Einlesen von Dateien wie für die Ausgabe soll nicht mitgemessen werden (Suchergebnisse zwischenspeichern und erst am Ende ausgeben). Geben Sie die Laufzeit in Sekunden über den `cerr` aus.

[1] http://www.cplusplus.com/reference/ctime/time/

Beispiel

```
./aufgabe4 chr1_noN.sa chr1_noN.fa queries.fa
1,1,1,1,1,1,1,1,1,58,1,1,1,1,1,2,1,...
time:9s
```

Bereitgestellte Dateien

- `chr1_noN.fa.gz`: Größerer Text (Modifiziertes Chromosom 1, Länge 225.279.450). Entpacken mit *gunzip*
- `chr1_noN.sa`: Suffix Array zum großen Text. Einlesen mit der bereitgestellten Funktion. Größe ca. 850 MB.
- `queries.fa`: 300 DNA-Sequenzen aus dem mod. Chromosom 1. Beachten Sie, dass zur Laufzeitmessung deutlich größere Dateien ($>$ 1 Mio Zeilen) benutzt werden.
- `queries.results`: Suchergebnisse zu `queries.fa`
- `aufgabe4_basis.cpp`: C++ Funktion zum Einlesen des Suffixarrays

Praktikumshinweise Beachten Sie die Hinweise unter `https://www.mi.fu-berlin.de/w/ABI/AlDaBiWS13Praktikum`.