**Technische Universität Berlin** **SS99**
**Fachbereich Mathematik**
Sekr. MA 6-1 / Frau Marcus

Dr. Michael Joswig

Olaf Jahn
Matthias Müller, André Opitz

# 7. Programmieraufgabe

**Abnahme: spätestens 20.07.99 zur Rechnervorrangzeit**

Jetzt sollt Ihr endlich eine „richtige" Suchmaschine programmieren, natürlich mit einer schönen Fensteroberfläche (keine Ausgaben in ein `xterm`!). Sie soll in zwei aufeinanderfolgenden Phasen arbeiten:

**Phase I** Das Programm baut seinen Datenbestand auf. Dazu sucht es ausgehend von einer Startseite bis zu einer vorgegebenen Suchtiefe Webseiten ab und merkt sich, wo welches Wort wie oft vorkommt, d. h. es erstellt einen Index. Extrem häufige Wörter wie „ist", „ein", „der", „the", „is", „a", etc. sollten nicht mit abgespeichert werden. Wie schon in der 5. Programmieraufgabe darf keine Seite mehrfach angefordert werden. Dem Benutzer soll es möglich sein, Startseite und Suchtiefe einzugeben.

Eine sichtbare Fehlermeldung braucht Ihr nur dann auszugeben, wenn es schon mit der URL, die der Benutzer eingegeben hat, Probleme gibt. Fehler, die bei Folgeseiten auftreten, ignoriert Ihr einfach.

Für jeden Webserver (erkennbar am Host-Teil der URL), von dem das Programm eine Seite anfordern möchte, holt es sich zunächst dessen `/robots.txt` und speichert deren Inhalt gemäß der letzten Programmieraufgabe. Von diesem Webserver dürfen keine Seiten angefordert werden, die er in seiner `/robots.txt` verboten hat.

Natürlich darf von jedem Webserver die `/robots.txt` nur ein einziges Mal geholt werden. Auch ist es möglich, daß ein Webserver gar keine `/robots.txt` vorsieht.

**Phase II** Nachdem der Datenbestand gespeichert ist, kann der Benutzer Anfragen nach Wörtern stellen. Euer Programm gibt ihm die Titel und URLs der Seiten aus, auf denen in Phase I dieses Wort gefunden wurde, absteigend sortiert nach Häufigkeit des Wortes auf der jeweiligen Seite. Gebt die jeweiligen Häufigkeiten mit aus.

In dieser Phase dürfen keinerlei Seiten mehr von Webservern geholt werden. Die Häufigkeiten der Wörter sollen bereits in Phase I ermittelt und abgespeichert worden sein.

Überlegt Euch vor dem Tippen insbesondere

- wie Ihr die zwei Phasen von der Oberfläche her trennen wollt, und
- wie Ihr in Phase I die Daten abspeichert (und welche Daten überhaupt anfallen), so daß Ihr sie in Phase II bequem abrufen könnt.

Wir wünschen Euch viel Vergnügen mit dieser letzten Programmieraufgabe des Semesters.