

Übung 9 [24]: Vergleich von Webseiten

Schreiben Sie ein Programm, das N Webseiten gleichzeitig aus N Dateien lädt und sofort vergleicht. Die Dateinamen werden als Kommandozeilenparameter übergeben. Im Fehlerfall (Exception) ist eine Fehlermeldung auszugeben, wobei jedoch zu beachten ist, daß nur der Vergleichs-Thread (und das Hauptprogramm) Ausgaben auf dem Bildschirm machen dürfen!

Die Webseiten sind zeichenweise in einem Buffer abzulegen, aus dem der Vergleichs-Thread (siehe unten) sich diese abholt. Wo dieser Buffer sich befindet (Downloader, Vergleicher oder ein eigenes Objekt) müssen Sie sich überlegen.

Nach jeweils 128 verglichenen Bytes geben Sie ein Zeichen '#' als Fortschrittsanzeige aus (ähnlich FTP).

Struktur des Programms:

1. Hauptprogramm: Anzahl der Webseiten wird hier festgelegt, steuert alles. Darf erst beendet werden, wenn alle anderen Threads beendet sind.
2. Vergleicher-Thread: Wartet bis von allen Downloadern mindestens ein Byte vorhanden ist und vergleicht diese dann. Dies ist (außer dem Hauptprogramm) der einzige Thread, der Ausgaben macht! Dies bedeutet, daß Fehler in den Lade-Threads an diesen Thread weitergereicht werden müssen (z. B. in Form einer Fehlermeldung), der sie ausgibt und danach das Vergleichen beendet (Rest der Daten muß gelesen werden!).
3. N Lade-Threads: Laden eine bestimmte Datei und speichern die Daten in einen Buffer. Diese Threads müssen sich automatisch selbst eindeutig numerieren. Diese Nummer ist auch bei Fehlermeldungen zu integrieren.

Test-Dateien:

Die Dateien test1 und test2 sind gleich, test3 ist in der Mitte anders, test4 ist gleich wie test1, aber etwas laenger.

Beachten Sie:

- Die verglichenen Dateien müssen auch alle gleich lang sein
- Testen Sie Exceptions, die erst in den Lade-Threads auftreten (z. B. falscher Dateiname)
- Bauen Sie in die Lade-Threads kurze zufällige Verzögerungen (1-5 msec nach jedem Byte) ein, um realistische Ergebnisse zu erhalten
- Verwenden Sie die Klasse Semaphore
- Sind die Dateien ungleich lang, müssen die längeren bis zum Ende gelesen werden, da sonst bei einem vollen Buffer ein Deadlock eintritt.

Beispiel 1:

```
C:>Java Vergleicher test1.htm test2.htm test3.htm
```

```
3 Dateien werden verglichen:
```

```
#####
```

```
Dateien sind ungleich
```

Beispiel 2:

```
C:>Java Vergleicher test1.htm test2.htm
```

```
2 Dateien werden verglichen:
```

```
#####
```

```
Dateien sind gleich
```

Beispiel 3:

```
C:>Java Vergleicher test1.htm test5.html
```

```
2 Dateien werden verglichen:
```

```
Fehler in Thread 2: Could not find file test5.html:  
java.io.FileNotFoundException: test5.html
```

```
Dateien sind ungleich
```