



Aufgabe A4 CRC berechnen

Aufgaben-Idee:

Sie sollen den CRC von vielen(!) Dateien (z.B. allen Fotos) berechnen.

Für jede Datei soll die CRC-Berechnung für jedes der 4 folgenden 33 Bit-Polynome (geeignet für 32 Bit) erfolgen:

0x1000000af	(Polynom-ID: 0)
0x100400007	(Polynom-ID: 1)
0x104c11db7	(Polynom-ID: 2)
0x127673637	(Polynom-ID: 3)

Es soll das gegebene Interface `SignatureProcessor_I` unterstützt werden und in der Konsequenz muss die Methode "`Collection<Item_I> computeSignatures()`" eine Collection über Objekte, die das gegebene Interface `Item_I` unterstützen, abliefern.

Wie in der Vorlesung abgesprochen:

- Dürfen die physikalischen Dateien selbst nicht verändert werden.
- Sind für die Berechnung u.U. nötige Nullen nach dem Dateiende zu ergänzen.
(Siehe hierzu die Vorlesung)

Die CRC-Werte sind eindeutig. Mit dem `DemoAndReferenceProcessor` können Sie die Korrektheit ihrer berechneten Werte überprüfen. Bzgl. der Ergebnisse ist der `DemoAndReferenceProcessor` die **Referenz**.

Eine grundsätzliche Lösung ist u.a. mit `DemoAndReferenceProcessor` gegeben. Diese Lösung arbeitet jedoch streng sequentiell. Entwickeln Sie eine **performantere** Lösung. **Parallelisieren** Sie wo sinnvoll und möglich.

Sie sollten auch mit Datenmengen oberhalb von 100GiByte testen.

(Stand Mai 2016 – Fotoenthusiasten verlassen ohne eine leere 64GByte SDCard gar nicht erst das Haus ;-)

Sie haben alle Freiheitsgrade.

Sie müssen den gegebenen Code nicht verwenden und teils ist dies auch gar nicht sinnvoll.

Die Lösung muss in Java implementiert sein und den obigen Anforderungen genügen.

Tipp: Java bietet verschiedene IO-Packages: Z.B. :

`java.io` , `java.nio`

(bzw. `IO`, `NIO`, `NIO.2`)

Halten Sie Ihre Ideen, Überlegungen Erkenntnisse und Messungen als Plain-Ascii-Text (en.wikipedia.org/wiki/Plain_text) in einer Datei "**ThoughtsAndExperiences.txt**" fest und unterwerfen Sie diese Datei der GIT-Kontrolle.

Begründen Sie insbesondere die Verwendung aller verwendeten Java-Dinge, die

- (noch) nicht in der Vorlesung besprochen wurden und
- für die parallele Programmierung relevant sind.

Gehen Sie dabei auch auf das Folgende ein:

- Was sind deren Vorteile?
- Warum reichen die Mittel aus der Vorlesung nicht?

Mögliche Formatierung lässt sich z.B. mit den UNIX-Tools: `roff` / `nroff` / `troff` / `groff` / `cawf` vornehmen. Vermutlich haben Sie von der GIT-Bash aus Zugriff auf diese Tools.