

# Programmieren 2 – abschließende semesterbegleitende Arbeit (30.6.2020)

## Ablauf:

Es gelten die Hinweise in Ihrem Moodle Kurs über den Ablauf dieser prüfungsrelevanten Leistung.

## Aufgabe:

Sie sollen eine kleine Variante von ftp bzw. scp implementieren:

Ihr Programm besteht wenigstens aus zwei Komponenten, der *remote* und der *local* Komponente. Es wird davon ausgegangen dass die *remote* Komponenten in der Regel nicht auf dem gleichen Rechner, in jedem Fall aber nicht im gleichen Verzeichnis arbeitet. Die *remote* Komponente bietet folgende Eigenschaften, Funktionen:

- Eine Funktion erlaubt das Anlegen eines Files. Als Parameter mit ein Filename und ein InputStream. Über den InputStream wird der Inhalt des Files zur Verfügung gestellt. Diese Funktion soll ein neues (!) File anlegen und es mit den Daten aus dem InputStream füllen. Ein existierendes File soll nicht überschrieben werden.
- Eine zweite Funktion erlaubt das Löschen eines Files. Als Parameter wird ein Filename erwartet.

Die *local* Komponente bietet die gleichen Funktionen wie die entfernte, allerdings doppelt - die Funktionen zum Anlegen und Löschen eines Files existiert zweimal:

- Es ist erlaubt, Files lokal, d.h. im lokalen Verzeichnis der *local* Komponente anzulegen.
- Die *local* Komponenten kann über TCP eine Verbindung zur *remote* Komponente herstellen. Definieren Sie die Parameter zum Verbindungsaufbau (Host, Port) gern als Konstante oder als Eingabeparameter einer `main()`-Methode.
- Über eine aufgebaute TCP-Verbindung ist die *local* Komponente in der Lage, die Methoden der *remote* Komponenten aufzurufen. Damit kann die *local* Komponente Files anlegen und löschen allerdings im Verzeichnis (und auf dem Rechner) der *remote* Komponente.
- Implementieren Sie eine Methode, die ein lokales Files auf nach *remote* kopiert und umgekehrt.

Vermutlich klingt das alles schwieriger als es tatsächlich ist. Beide Komponenten können Files anlegen und über einen Stream füllen. Die lokale Variante kann zusätzlich eine TCP-Verbindung herstellen und damit die entfernte Komponenten aufrufen und damit Files auf der Gegenstelle erzeugen und löschen.

Gehen Sie wie folgt vor:

1. Geben Sie den beiden Komponenten sinnvolle Namen und definieren Sie Interfaces.
2. Arbeiten Sie testgetrieben und implementieren Sie erste Tests.
3. Übernehmen Sie den Code aus Ihrem Projekt zum Aufbau einer TCP-Verbindung. **Das ist kein Tipp, dass ist ein Muss!** Nutzen Sie dazu Ihren Code, den Sie in Github publizierten. **Schreiben Sie unbedingt einen Link auf das Github Projekt in Ihren Code an passender Stelle.** Die Herkunft dieses Codes werde ich definitiv prüfen.
4. Implementieren Sie das kleine Protokoll zwischen *local* und *remote*.
5. Beweisen Sie die Funktionsfähigkeit des gesamten Systems durch wenigstens einen Test.

**Wichtig:** Ein Programm ohne die TCP-Verbindung wird als **ungenügend** bewertet. Fehlt der Hinweis auf Ihr Github-Projekt wird die Abgabe als **ungenügend** bewertet. Es empfiehlt sich ebenfalls, einen „technischen Durchstich“ zu machen: Definieren Sie die Interfaces vollständig aber implementieren Sie zuerst die komplexeste Methode: das ist das Anlegen eines entfernten Files von der lokalen Komponenten aus (remote file copy). Wenn Sie das haben, ist der Rest ganz einfach. Es ist ein Spiel mit Streams. Wenn Sie damit nicht durcheinander kommen, ist alles ganz einfach. Wie immer im Leben und dem Rest gilt daher: Don't panic!