

WS 25/26 Verteilte Systeme

Praktikumsaufgabe 2

Ziel der Aufgabe ist es einen verteilten Algorithmus mit Hilfe von asynchronem Nachrichtenaustausch zu implementieren. Die Aufgabe soll in Teams mit zwei Personen bearbeitet werden. Bei Praktikumsgruppen mit ungerader Personenzahl gibt es ein dreier Team.

Der zu implementierende Algorithmus bestimmt den größten gemeinsamen Teiler (ggT) mehrerer Zahlen. Ausgeführt werden soll er von mehreren Prozessen, die über das Netzwerk kommunizieren. Für jede Ausgangszahl für die der ggT bestimmt werden soll, wird jeweils ein Prozess gestartet.

Die Prozesse sollen einen Ring bilden, in dem jeder Prozess mit seinem Vorgänger und Nachfolger kommunizieren kann. Dafür bekommt jeder Prozess beim Start die Referenz auf seinen Vorgänger, Nachfolger und eine der Ausgangszahlen übergeben.

Jeder Prozess soll folgenden Algorithmus ausführen, wobei M die übergebene Zahl des jeweiligen Prozesses ist:

```
# Eine Nachricht y ist eingetroffen
if y < M then
    M = mod(M - 1, y) + 1;
    send M to all neighbours;
end
```

- Nach dem Start aller Prozesse muss jeder Prozess beiden Nachbarn seine lokale Zahl M mitteilen. Dies startet die Ausführung des Algorithmus.
 - Die gesamte Kommunikation zwischen den Prozessen soll asynchron und über Nachrichten implementiert werden. Keiner der Prozesse darf blockieren.
 - Für die Implementierung dürfen Frameworks und Middlewares für die Kommunikation benutzt werden. Beispiele wären Aktor-Frameworks (z.B. Akka) oder Message Broker (z.B. ActiveMQ).
 - Die verwendete Programmiersprache kann frei gewählt werden. Eine Möglichkeit wäre die Implementierung in Java. Der Algorithmus muss entsprechend an die Programmiersprache angepasst werden.
 - Die Kommunikation soll über das Netzwerk laufen. Die Prozesse sollen gleichmäßig auf die zur Zeit der Ausführung zur Verfügung stehenden Rechner verteilt werden.
 - Die Kommunikation soll über ein standardisiertes menschenlesbares Nachrichtenformat, wie zum Beispiel JSON oder XML, laufen. Auch hier sind wieder Externe-Bibliotheken für die Serialisierung erlaubt.
 - Zum Starten der Prozesse soll ein eigenes Programm geschrieben werden, welches eine Methode implementiert, die es erlaubt den ggT für eine variable Anzahl von Zahlen zu bestimmen. Dieses Programm muss die entsprechenden Prozesse starten, eine vordefinierte Zeit warten und das Ergebnis abfragen.
Dieses Startprogramm soll auch die Verteilung der Prozesse auf den aktuell verfügbaren Rechner umsetzen.
- 1.) Definiert das Nachrichtenformat basierend auf dem gewählten Standard. Für jede Art von Nachricht ist der Aufbau zu beschreiben und ein Beispiel anzugeben.
 - 2.) Implementiert den Algorithmus in der Programmiersprache eurer Wahl

- 3.) Erstellt ein Sequenzdiagramm, das einen möglichen Ablauf des Algorithmus für die Zahlen 108, 76, 12, 60 und 36 darstellt. Der Ablauf soll vom Aufruf der Methode im Startprogramm bis zur Ausgabe des Ergebnisses darstellt werden.
- 4.) Wie kann festgestellt werden, dass der Algorithmus terminiert und das Ergebnis endgültig ist? Definiert alle hinreichenden Kriterien.
Dieser Schritt muss nicht implementiert werden, für das Praktikum reicht eine vorgegebene Wartezeit des Startprogramms.

Abzugeben ist ein Bericht in Form eines Text-Dokuments mit allen relevanten Design-Entscheidungen (z.B. Wahl der Programmiersprache) und Ergebnissen der Aufgaben. Der Source-Code muss während der Abgabe erklärt werden können. Die Implementierung muss auf den Pool Rechnern über das Netzwerk lauffähig sein.

Spätester Abgabepunkt ist am Anfang des dritten Praktikum Termins.