



## Verteilte Systeme, Übungsblatt 1, Sommer 2024

### Aufgabe 1

In dieser Aufgabe geht es um die tatsächliche Genauigkeit der Uhrensynchronisation. Recherchieren Sie im Internet nach Möglichkeiten, die Genauigkeit der lokalen Uhr eines Rechners über einen Zeitraum von mindestens einer Stunde kontinuierlich mit einem oder mehreren Zeitservern aus dem Internet zu vergleichen. Sammeln Sie die gemessenen Daten als CSV-Datei und generieren Sie daraus ein einfaches Diagramm, das die Ergebnisse graphisch wiedergibt. Abzugeben sind die CSV-Datei und ein kurzes Protokoll (vorzugsweise PDF), das den verwendeten experimentellen Ansatz beschreibt und das Ergebnis anhand des Diagramms diskutiert.

### Aufgabe 2 - Problemstellung

Entwerfen Sie ein einfaches Würfelspiel in einer Programmiersprache Ihrer Wahl. Dieses Spiel besteht aus einem Programm, das den Spielleiter verkörpert, und einem zweiten Programm, das, mehrfach instanziiert, mehrere Spieler simuliert. Das Würfelspiel soll in Runden ablaufen. Zu Beginn jeder Runde versendet der Spielleiter eine Nachricht **START** an alle angemeldeten Spieler. Mit dem Empfang einer **START**-Nachricht wartet der Spieler eine zufällige Zeit zwischen 0 und **SPIELER\_LATENZ** Sekunden, anschließend würfelt der Spieler eine Zahl **WURF** zwischen 1 und 100 und sendet diesen **WURF** zusammen mit seinem **NAMEN** zurück an den Spielleiter. Nachdem der Spielleiter die **START**-Nachricht an alle versendet hat, wartet er **DAUER\_DER\_RUNDE** Sekunden und versendet eine **STOP**-Nachricht an alle Teilnehmer. Anschließend ermittelt der Spielleiter den Gewinner mit dem höchsten **WURF** in der Runde und protokolliert dieses Ergebnis in einer Datei. Die Werte für **SPIELER\_LATENZ** und **DAUER\_DER\_RUNDE** sollen einfach veränderbare Parameter sein und z.B. als Kommandozeilenargumente festlegbar sein. Semantisch soll jeder von einem Teilnehmer zurückgesendete Wurf jeweils in der Runde nach der kausal zugehörigen **START**-Nachricht berücksichtigt werden.

Es ist klar, dass durch ergänzende Informationen wie z.B. dem Durchnummerieren der Runden und der Angabe der Rundenummer in den Wurfantworten alle Kausalitätsprobleme einfach gelöst werden können. Da es um allgemeine Lösungsansätze rein auf der Basis von Zeit oder logischen Uhren geht, soll deshalb auf diese Zusatzinformation verzichtet werden (würde diese Würfelspiel in einem Raum stattfinden und alle Teilnehmer einschließlich dem Spielleiter sitzen an einem Tisch, dann würden wir ja auch keinen Ausruf „Es beginnt Runde 7322“ erwarten, sondern nur eine Meldung „Runde beginnt“). Sie können davon ausgehen, dass keine Teilnehmer das Spiel vorzeitig verlassen oder ausfallen, auch gehen keine Nachrichten verloren (vorzugsweise verwenden Sie TCP bei der Implementierung). Entscheiden Sie sich für eine Vorgehensweise bei zu spät eintreffenden Würfeln, sofern Sie diese überhaupt erkennen können.

### Aufgabe 2a

*DOCKER is day ☺*

Ihre erste Implementierung soll nur die konkreten Uhren auf den beteiligten Rechnern nutzen, ohne diese Zeiten aber in den Nachrichten mitzusenden. Starten Sie das Programm auf mindestens 3 verschiedenen Rechnern (nicht **localhost**), um zumindest ansatzweise verschiedene Uhren zu verwenden. Wer Zugriff auf geographisch weit entfernte Rechner hat, sollte diese einbeziehen.

### Aufgabe 2b

Realisieren Sie eine Implementierung ausschließlich auf der Basis logischer Uhren. Die logischen Zeitstempel sollen zusammen mit den Nachrichten versendet werden. Starten Sie auch dieses Programm auf mindestens 3 verschiedenen Rechnern (nicht **localhost**).

### Aufgabe 2c

Führen Sie die Implementierung aus Aufgabe 2b ausschließlich auf dem lokalen Rechner (**localhost**) aus. Instanzieren Sie diesmal möglichst viele Teilnehmer.

### Abgabe

Die Abgabe von Aufgabe 1 besteht aus einem kurzen Protokoll und der CSV-Datei. Die Abgabe von Aufgabe 2 ist ein Link auf ein Github-Repository, das die beiden Implementierungen zu 2a und 2b sowie einen kurzen, maximal 2-3 Seiten umfassenden, Ergebnisbericht beinhaltet. Gehen Sie in diesem Bericht auch auf den Aufwand ein, der sich aus der experimentellen Umsatz ergab, und diskutieren Sie insbesondere den potentiellen Aufwand für Teilaufgabe 2c, wenn alle Spieler auf physisch verschiedenen Rechnern ausgeführt werden sollen.

Beide Abgaben haben eine Frist bis zum 2.6.2024.