



SA4E, Übungsblatt 1, Winter 2024

Aufgabe 1 „Zaghafte erste Glühwürmchen“ (Monolith - Multi-Threaded)

In dieser Übung implementieren Sie eine Simulation von synchronisierten Glühwürmchen in einer Programmiersprache Ihrer Wahl. Ihre Aufgabe besteht darin, ein einzelnes Programm zu erstellen, das mehrere als Threads implementierte Glühwürmchen in einer Torus-Struktur anordnet und simuliert. Der Zustand jedes Glühwürmchen soll durch ein farbiges Rechteck in einem Fenster dargestellt werden. Die Simulation soll den Prozess erfassen, bei dem benachbarte Glühwürmchen sich gegenseitig beeinflussen und ihre Aufleuchtzyklen gemäß dem Kuramoto-Modell allmählich synchronisieren. Dieses Modell verwendet einfache Regeln zur Synchronisation und Nachbarschaftsinteraktion, bei denen jedes Glühwürmchen vom Zustand seiner unmittelbaren Nachbarn im Torus beeinflusst wird. Ziel ist es, eine einfache und intuitive Benutzeroberfläche zu erstellen, die die Torus-Anordnung und den Synchronisationsprozess visualisiert, wobei Farbe oder Helligkeit verwendet werden, um den Aufleuchtzyklus jedes Glühwürmchens darzustellen.

Aufgabe 2 „Kommunizierende Glühwürmchen“ (Distributed – Apache Thrift/gRPC/Java RMI)

In dieser Übung erweitern Sie Ihre Glühwürmchen-Simulation aus Aufgabe 1, indem Sie jedes Glühwürmchen als eigenständiges Programm realisieren, das über ein Netzwerk mit den anderen Glühwürmchen kommuniziert. Ziel ist es, die notwendigen Zustandsinformationen zwischen den einzelnen Glühwürmchen mithilfe eines Kommunikationsprotokolls wie Apache Thrift oder gRPC auszutauschen, um den Synchronisationsprozess zu steuern. Jedes Glühwürmchen fungiert als unabhängiger Prozess und tauscht regelmäßig Statusinformationen mit seinen Nachbarn in der Torus-Anordnung aus, um eine Synchronisation zu erreichen. Wenn Sie die Programmiersprache Java verwenden, ist alternativ auch Java RMI als Kommunikationsmechanismus zulässig. Die Aufgabe besteht darin, die dezentrale Natur der Simulation zu verdeutlichen und sicherzustellen, dass die Glühwürmchen auf Basis der empfangenen Informationen den Synchronisationsprozess wie zuvor initiieren können.

Abgabe

Abzugeben ist ein Ergebnisbericht im PDF-Format, der Ihre Lösungen zu den Aufgaben 1 und 2 erläutert, und ein oder zwei kurze Videos mit den Simulationen. Die von Ihnen entworfenen Programme müssen vollständig im Sourcecode vorliegen; dies geschieht vorzugsweise in der Form eines Links auf ein Github-Repository. Deadline für die Abgabe ist der 1.12.2024.