

Parallelprogrammierung Assignments

Bei allen **Assignments** handelt es sich um **Prüfungsleistungen**!

Die Lösungen sind als in Eclipse bzw. IntelliJ importierbares (Java-)Projekt spätestens bis zum jeweils angegebenen Abgabedatum, 24.00 Uhr auf den Git-Server des Labors für Multimedia hochzuladen und in der ersten Übung nach dem Abgabedatum kurz zu präsentieren! Verwenden Sie für jedes Assignment das entsprechende bereits auf dem Git-Server vorhandene, leere Repository und legen Ihre Abgaben dort jeweils in einem Branch mit dem Namen `assignment` ab! Zusätzlich ist dort eine Textdatei `members.txt` abzulegen, die die Namen und die Matrikelnummern der an der Abgabe beteiligten Personen enthält. Verwenden Sie für jedes Assignment höchstens einen Java-Source-Folder. Repositories, die eine Struktur enthalten, die sich nicht ohne Änderung in Eclipse/IntelliJ importieren lässt, werden nicht berücksichtigt!

Der Git-Server ist über die IP-Adresse `172.19.49.46` innerhalb des Hochschulnetzes direkt und von außerhalb über VPN erreichbar. Die Accounts haben die Form `PPWiSe24xx`; die Anmeldedaten werden den Gruppen auf Anforderung per E-Mail zugeschickt.

Das direkte Einloggen auf dem Server ist z.B. durch `ssh PPWiSe24xx@172.19.49.46` möglich; durch `git clone PPWiSe24xx@172.19.49.46:git/assignment01.git` kann z. B. das Git-Repository für das erste Assignment geklont werden.

Assignment 1

Abgabedatum: 18. Dezember 2024

Arbeitsgruppen: 3-er-Gruppen

1.1

Eine Achterbahn in einem Freizeitpark wird durch die drei Teilprozesse `Drehkreuz`, `Steuerung` und `Wagen` modelliert. Der Einstieg in den Wagen wird durch eine `Steuerung` organisiert: Wenn noch keine M Passagiere in dem Wagen Platz genommen haben, darf ein weiterer Passagier durch das `Drehkreuz` gehen und sich in den Wagen setzen. Sobald genau M Passagiere im Wagen sitzen, kann er abfahren. Nach Durchfahren der Bahn steigen alle Passagiere gleichzeitig aus, und die nächsten M Passagiere dürfen nacheinander in den Wagen einsteigen.

Die Prozesse `Drehkreuz` und `Steuerung` interagieren durch die gemeinsame Aktion `passagier`, die signalisiert, dass ein weiterer Passagier das `Drehkreuz` passiert und sofort in dem Wagen Platz nimmt. `Steuerung` und `Wagen` besitzen die gemeinsamen Aktionen `abfahrt` und `aussteigen`, die anzeigen, dass der Wagen abfahren kann bzw. dass alle Passagiere den Wagen wieder verlassen.

Zeichnen Sie ein Strukturdiagramm der Achterbahn und modellieren Sie die Achterbahn durch FSP.

1.2

Implementieren Sie Ihr Modell aus Aufgabe 1.1 in Java unter Verwendung des Monitorkonzepts entsprechend dem Vorgehen aus der Vorlesung. Welche der FSP-Prozesse sind dabei als Thread und

welche als Monitor umzusetzen?

Für jeden implementierten Prozess muss durch geeignete `System.out`-Ausgaben jederzeit nachvollziehbar sein, in welchem Zustand er sich gerade befindet. Eine GUI zur Visualisierung ist ausdrücklich nicht gefordert.

1.3

- a) Erweitern Sie Ihr Modell aus Aufgabe 1.1 so, dass drei Wagen im Einsatz sind. Die Steuerung muss dabei folgende Regeln beachten:
- zur Zeit darf sich nur maximal ein Wagen auf der Bahn befinden
 - zu Beginn stehen alle Wagen hintereinander am Einstieg
 - stehen mehrere Wagen am Einstieg, steigen die Passagiere nichtdeterministisch in einen der dort stehenden Wagen ein
 - der vordere der am Einstieg stehenden Wagen kann abfahren, sobald dort M Passagiere eingestiegen sind (ein dahinter stehender Wagen darf auch dann nicht zuerst abfahren, wenn er bereits mit M Passagieren besetzt sein sollte!)
 - der nächste Wagen darf frühestens dann abfahren, wenn der vor ihm abgefahrene Wagen nach Ende seiner Fahrt, und nachdem alle seine Passagiere ausgestiegen sind, wieder hinten am Einstieg steht
 - es kann davon ausgegangen werden, dass die am Einstieg stehenden Wagen automatisch vorrücken, sobald vor ihnen ein Stellplatz frei geworden ist.
- b) Implementieren Sie Ihr Modell aus Aufgabe 1.3a in Java unter Verwendung des Monitorkonzepts entsprechend dem Vorgehen aus der Vorlesung.