Exercise 9

10.02.2003 **Due date: 23.01.2003**

Assignment 9.1.

Betrachte eine Warteschlange vor einem Ticketschalter (z.B. Kino). Die Ankunft von Kunden und das Verlassen des Schalters ist das Gesamtereignis individueller Entscheidungen einer spezifischer Anzahl von Personen.

Mit Zufallszahlen werden Ereignisse (events) erzeugt, wie zum Beispiel die Ankunft von Kunden, oder die Beendigung eines Vorganges. Der Zustand des Modells ert sich nur zu diskreten Zeitpunkten. Zwischen diesen Ereignissen passiert nichts. Bei der Simulation einer Warteschlange im Computer springt die Zeit von Ereignis zu Ereignis. Diese Art der Simulation diskreter Ereignisse heisst deshalb auch zeitdiskret.

??????????? Zur Simulation der Warteschlange werden zwei Folgen von Zufallszahlen ben?t: eine, um die Zeit zwischen den Ankunftsereignissen zu bestimmen, die andere, um die Abfertigungsdauer eines Kunden zu bestimmen.

Quelle **Warteschlange** **Station/Bedienung**

??????????? Eine grosse Anzahl von Anwendungen kann in die Einheiten Kunden, Quelle und Stationen zerlegt werden. Oben stehendes Bild zeigt diese Elemente im Fall einer einzelnen Warteschlange. Es liegt folgende Situation vor:

??????????? Kunden durchlaufen das System. Quellen erzeugen gemaess einer Strategie Kunden. Stationen sind Einheiten, bestehend aus einer Bedieneinheit und einer Waretschlange. Die Station ordnet Kunden in ihre Warteschlange ein und sorgt daf?ss sie bedient werden.

Grundidee:

1. Der Kunde besucht die Station und fordert einen Dienst von dieser Station ein.
2. Die Station benachrichtigt den Kunden, wenn der gew?e Dienst zur Verf?steht. Der Kunde legt fest, wann die Bedienzeit beendet ist.
3. Nach Ablauf der Bedienzeit gibt der Kunde die Station frei, bevor er das System verl t.

Die Elemente:

1. **Kunden** sind Paare von Aktion und Ereignis und wissen stets, wie sie zu reagieren haben, wenn sie eine bestimmte Nachricht erhalten. Kunden besitzen eine Instanzvariable **Zeit**, in der Eintrittszeitpunkt des Ereignisses gespeichert wird, sowie eine Variable **Dauer,** die die Dauer der Aktion beschreibt. Die Methode commit() ist die einzige Methode, die die Funktionalitaet des Kunden beinhaltet. Die Simulation ruft die Methode auf, wenn der Kunde an der Bedienung ist.
2. Kunden werden zuf ig mittels eines **Kundengenerator** erzeugt. Kunde und der Kundengenerator sind Simulationskomponenten, die zu einem bestimmten Zeitpunkt benachrichtigt werden m? also Unterklasse derselben Klasse sind. Betrachten wir unter diesem Gesichtspunkt den Kundengenerator genauer. Er erzeugt Kunden zuf ig. Man spezifziert die Methode ***commit*** der Klasse Kundengenerator so, dass zun st ein neuer Kunde erzeugt wird und dann bereits das n ste ankommende Ereignis, d.h. der n ste Kundengenerator, erzeugt wird, und in die Warteschleife der Simulation eingereiht wird.
3. **Warteschlangen** sind eine wichtiges Simulationselement. So bilden Kunden eine Warteschlange vor einer Servicestation. Eine Warteschlange kann Elemente u.a. nach dem FIFO (First-In-First-Out) oder nach dem LIFO (Last-In-First-Out) Prinzip verwalten. Das Zeit/Dauer-Attribut des Kunden erlaubt dar?naus einen weiteren Warteschlangentyp: die geordnete, sortierte Warteschlange.
4. **Die Simulation** realisiert ein diskretes Simulationssystem, in dem Ereignisse (Kunden, Kundengeneration) verarbeitet werden. Entsprechend besteht die Hauptaktivitaet der Simulation darin, ueber Kunden und Kundengeneratoren zu iterieren, wobei der Eintrittszeitpunkt des zugeh?en Ereignisses als Sortierkriterium dient. Die Reihenfolge der Ereignisse wird mittels einer sortierten Warteschlange realisert. Die Simulation ist durch eine Dauer begrenzt. Die Dauer wird als Parameter zu Beginn der Simulation festgelegt.
5. **Stationen** bearbeiten hereinkommende Anforderungen der Kunden. Falls die Station belegt ist, werden die Kunden vor der Station in eine Warteschlange eingereiht.

Dynamik:

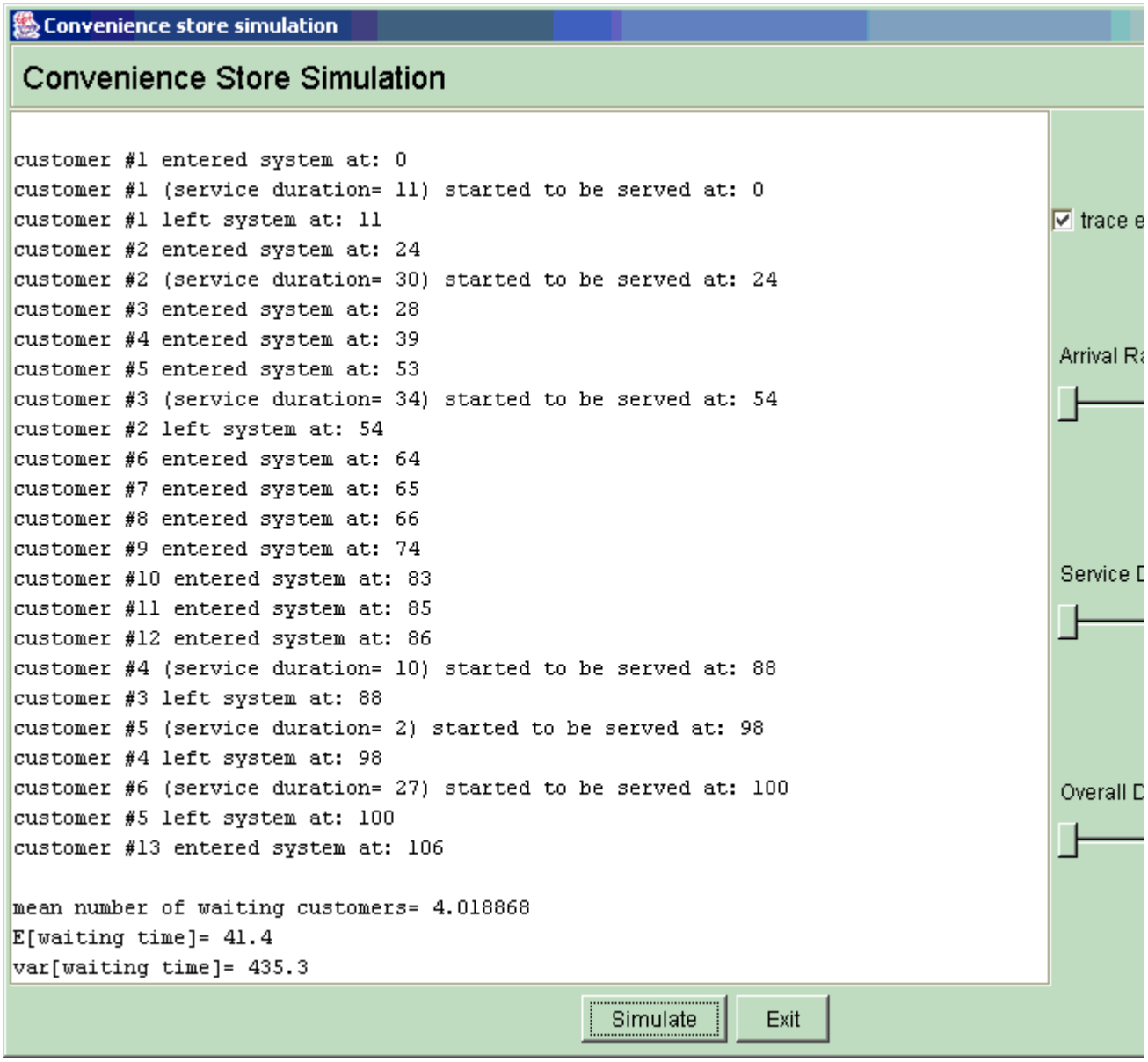
Im folgenden betrachten wir den einfachsten Fall einer Service-Station. Sie beschr t die Warteschlange nicht, bietet nur eine Service-Station an und behandelt die ankommenden Kunden nach dem FIFO Prinzip.

Der Kundengenerator erzeugt Kunden, die ihrerseits mit der Service-Station interagieren. Konkret ben?t ein neu erzeugter Kunde eine Service-Station. Wenn die Service-Station frei ist, wird der Kunde sofort bedient, andernfalls wird er in die Warteschlange vor der Service-Station eingef?obald die Service-Station frei ist, wird die Nachricht ***activate*** an den Kunden geschickt, der als n stes an der Reihe ist. Dabei legt der Kunde die Dauer der Bearbeitung fest. Nachdem der Kunde bearbeitet wurde, erh er die ***commit*** Nachricht, gibt die Service-Station frei und verl t das Simulationssystem.

Der Kunde fordert einen Dienst von der Service-Station mittels der ***requestService*** Nachricht an. Falls die Service-Station besetzt ist, wird der Kunde in die Warteschlange eingeordnet, ist sie aber frei, so wird sie durch den Kunden besetzt (siehe oben). Durch die Simulation wird der Service-Station mitgeteilt, dass der Kunde fertig ist (***commit***). Die Service-Station wird freigegeben. Falls die Warteschlange nicht leer ist, kommt der am l sten wartende Kunde an die Reihe.

Ausgabe:

Die Ausgabe der Simulation sollte Auskunft geben ueber den Verlauf der Simlation und wiefolgt aussehen:



Pr ntation:

Bei der Demonstration des Programmes sollte ein Sequenzdiagramm zur Visualisierung des Programmablaufs pr ntiert werden k?n !!!