Informatik II

`http://www-pu.informatik.uni-tuebingen.de/info-ii-y2k/`

Blatt 9 („Testatblatt“)

Vorführung: ab 20.7.2000

Dieses — letzte — Blatt hat Testatfunktion. Es gibt keine Abgaben; stattdessen wird die Lösung bei einem persönlichen Termin mit der Tutorin bzw. dem Tutor einzeln vorgeführt und erläutert. Mache also rechtzeitig einen Termin aus. Bei *allen* auftretenden Fragen oder Problemen, wende Dich an Deine Tutorin, Deinen Tutor oder Michael Sperber, und zwar sofort. Hinweise werden außerdem ggf. auf der Homepage gepostet. Für erfolgreiche Absolvierung dieses Testats gelten vergleichbare Kriterien wie für ein normales Übungsblatt.

Mike Sperber hat nach seinem Abgang von der Universität bei SAP eine hochbezahlte Stelle als Core-Entwickler angenommen und arbeitet jetzt nur noch montags vormittags. In seiner Freizeit regt er sich über Probleme des Alltags auf und sucht unermüdlich nach Lösungen und Wegen für eine bessere Welt. Beim täglichen Gang zum Fitness-Studio stößt ihm immer wieder der veraltete Aufzug in seinem Gebäude unangenehm auf: Die Programmierung aus den 70er Jahren ist (so Mike) hoffnungslos ineffizient und dämlich. Er beschließt, das Problem anzugreifen, und zwar total objektorientiert. Du bist Mike!

Die Aufzugtechniker (ebenfalls aus den 70er Jahren) sind zunächst nicht kooperativ, als Du ihnen lauthals unfaßbare Geschwindigkeitssteigerungen und Customer Satisfaction verspricht, wenn sie Dich die Software auswechseln lassen. (Daß Du sie am Ende „inkompetente Kabelaffen“ nennst, kann die Situation auch nicht verbessern.) Darum schreibst Du zunächst einen Aufzugsimulator, um es den Fatzkes von der Technik mal so richtig zu zeigen.

Arbeitsauftrag: Entwickle einen Simulator für einen Aufzug in einem Hochhaus mit beliebig vielen Stockwerken. Gehe davon aus, daß es nur einen Fahrstuhl im Haus gibt.

- Überlege Dir und dokumentiere eine einfache und effektive Fahrstrategie für einen Fahrstuhl. Wie entscheidet ein Fahrstuhl, abhängig von der eigenen Position und Fahrtrichtung und den Wünschen der Menschen in den Stockwerken, ob er nach oben oder unten weiterfährt? Der Fahrstuhl sollte dabei keinen potentiellen Fahrgast unendlich lang warten lassen!

- Schreibe mindestens Klassen für

  - Menschen, die einen Namen haben und die in ein bestimmtes Stockwerk gelangen wollen,
  - Stockwerke, die einen Namen haben, und in denen sich Menschen an der Aufzugstür anstellen, und die einen Aufzug, wenn er vorbeikommt, mit wartenden Menschen auffüllt,
  - Aufzüge, die Menschen transportieren und bestimmte Stockwerke anfahren können — letzteres auch gemäß der überlegten Strategie.

  Es könnte dabei hilfreich sein, sich zu überlegen, welche der obigen Entitäten in welchen anderen enthalten sind.

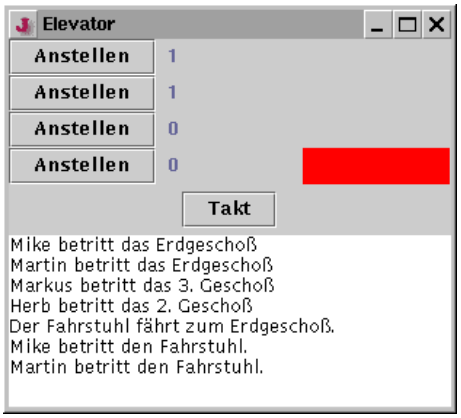
  Für die Menge, der wartenden bzw. fahrenden Menschen und die Menge der Stockwerke bietet sich die Verwendung der Klasse `java.util.Vector` an.

- Bringe in der Klasse für den Aufzug eine Methode `void step()` unter, die einen Zeittakt simuliert: Bei jedem Taktschlag fährt der Aufzug in das gemäß der Strategie angesagte Stockwerk. Dann steigen alle Fahrgäste im Aufzug, die das Stockwerk als Wunsch angegeben haben, aus, alle wartenden Fahrgäste steigen ein. All diese Einzelaufgaben sollen durch separate Methoden der Klasse für den Aufzug oder anderer Klassen erledigt werden, die durch `step` (direkt oder indirekt) aufgerufen werden.

- Schreibe Methoden, die Text über die Aktionen der Simulation ausgeben, also darüber, wer ein- und aussteigt und wohin der Aufzug fährt. Schreibe mit ihrer Hilfe ein Testprogramm, in dem in im Programm festgelegter Reihenfolge Menschen erzeugt werden bzw. der Zeittakt ausgelöst wird. Ein Beispiel ist im Fenster weiter unten zu sehen.

  Anmerkung: Hier befindet sich in der Aufgabenstellung die 50%-Marke.

- Entwickle eine grafische Benutzeroberfläche für die Simulation.
- Das Fenster der Simulation könnte etwa so aussehen (kreativer Freiraum ist selbstverständlich gestattet.)



  Die Position des Aufzugs ist als dunkelgrauer/roter Balken zu erkennen. *(Eine Klasse mit einer GUI-Komponente, die wahlweise einen roten Balken oder nichts anzeigt, befindet sich auf der Homepage der Vorlesung.)*

  Das Label-Feld in den Stockwerken zeigt die Anzahl der Menschen an, die dort warten.

  Der Knopf `Takt` löst den Takt aus. Dabei sollen die Aktionen wie oben beschrieben ausgelöst werden; außerdem soll die Anzeige des Aufzugs und der wartenden Menschen in den Stockwerken aktualisiert werden. Die Einstiegs- und Ausstiegsvorgänge sollen im Textfenster unten mitprotokolliert werden.

- Bei Druck auf einen der Knöpfe in den Stockwerken erscheint ein Dialog, in dem der Name eines neuen Menschen eingegeben und ein Zielstockwerk ausgewählt werden kann, etwa so:

  Dort kann ein Stockwerk mit einem Popup-Menü ausgewählt werden:

  Für derartige Menüs ist die `JComboBox`-Klasse hilfreich.