

Übungsblatt 1

Aufgabe 1.1 Fahrkartenautomat

[12 Punkte]

Sie haben die Aufgabe, einen endlichen Automaten zu konstruieren, der eine Fahrkarte für 3 Euro verkauft. Dabei sollen folgende Regeln gelten:

- Der Automat befindet sich zu Beginn im Grundzustand (“Idle”).
- Man kann mit Münzen von 1 Euro oder 2 Euro bezahlen. Die Reihenfolge des Einwerfens ist egal. Andere Münzen werden sofort wieder ausgeworfen.
- Wurde zuviel eingeworfen (zweimal 2 Euro), wird die letzte Münze sofort wieder ausgeworfen.
- Die Fahrkarte wird automatisch gedruckt, sobald 3 Euro exakt erreicht sind.
- Solange 3 Euro noch nicht erreicht sind, kann die Transaktion mit einem “Abbrechen”-Knopf abgebrochen werden. Das bereits gezahlte Geld wird dann wieder ausgeworfen.
- Nach dem Drucken einer Fahrkarte überprüft ein Sensor, ob in der Geldkassette noch genügend Platz für eine weitere Transaktion ist. Ist dies der Fall, kehrt der Automat in den Grundzustand zurück. Andernfalls geht er in den Zustand “Außer Betrieb” über. Wird in diesem Zustand Geld eingeworfen, wird es sofort wieder ausgeworfen und die Meldung “Außer Betrieb” angezeigt.

Listen Sie zuerst die notwendigen Zustände, Ereignisse und Aktionen auf. Zeichnen Sie dann den endlichen Automaten in *Graphendarstellung* (Zustände werden als Kreise dargestellt, Zustandsübergänge als Pfeile, die mit dem auslösenden Ereignis und der ausgeführten Aktion beschriftet sind). Sie können dafür ein Zeichenprogramm Ihrer Wahl benutzen oder eine (lesbare!) Papierzeichnung anfertigen, die Sie dann mit Ihrem Handy fotografieren. Die Lösung soll in jedem Fall als PDF-File eingereicht werden.

Aufgabe 1.2 Bibliothek

[16 Punkte]

Konstruieren Sie einen endlichen Automaten, der die Bibliotheksausleihe eines einzelnen Buches regelt.

- (a) Im ersten Aufgabenteil soll der Automat das folgende Verhalten realisieren:
- Der Automat befindet sich zu Beginn im Grundzustand.
 - Trifft das neu gekaufte Buch ein, wird es katalogisiert und ist danach “ausleihbar”.
 - Ein ausgeliehenes Buch muss nach vier Wochen zurückgegeben werden. Geschieht dies, ist die Ausleihe kostenlos, und das Buch ist danach wieder ausleihbar. Andernfalls erhält der Entleiher eine erste Mahnung. Der Ablauf der Leihfrist

soll durch einen `Timer` signalisiert werden (siehe Vorlesung). Da ein endlicher Automat nicht über zusätzlichen Speicher verfügt, kann er sich den Namen des Entleihers leider nicht merken, aber dieses klitzekleine Problem dürfen Sie hier ignorieren.

- Nach der ersten Mahnung muss das Buch innerhalb von zwei Wochen zurückgegeben werden. Geschieht dies, wird eine Verzugsgebühr von 5 Euro fällig, und das Buch ist danach wieder ausleihbar. Andernfalls wird eine zweite Mahnung verschickt. Der Ablauf der Mahnfrist soll ebenfalls durch einen `Timer` signalisiert werden.
- Nach der zweiten Mahnung muss das Buch innerhalb von einer Woche zurückgegeben werden. Geschieht dies, wird eine Verzugsgebühr von 20 Euro fällig, und das Buch ist danach wieder ausleihbar. Andernfalls wird das Buch aus dem Katalog gestrichen und die Angelegenheit an einen Rechtsanwalt übergeben. Der Automat kehrt dann in den Grundzustand zurück.
- Der Versuch, ein verliehenes Buch nochmals auszuleihen, soll die Fehlermeldung “Buch ist verliehen” auslösen.
- Ist das Buch “ausleihbar”, wird ab und zu sein Zustand überprüft. Wenn das Buch in schlechtem Zustand ist (inhaltlich veraltet oder beschädigt), wird es weggeworfen und aus dem Katalog ausgetragen.

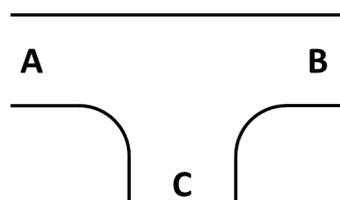
Definieren Sie den Automaten durch eine *Übergangstabelle*, deren Zeilen den aktuellen Zuständen und deren Spalten den Ereignissen entsprechen. Die Felder der Tabelle beschreiben die Aktion, die in der betreffenden Situation ausgeführt wird, und den Folgezustand, in den der Automat danach übergeht. Wenn ein Ereignis in einem bestimmten Zustand nicht auftreten kann, bleibt das betreffende Feld leer.

- (b) Erweitern Sie den Automaten um die Möglichkeit, ein verliehenes Buch vorzubestellen. Das Buch kann immer nur von einer Person vorbestellt werden (deren Namen sich der Automat ebenfalls nicht merken kann :-), eine Warteliste von Vorbestellungen soll nicht unterstützt werden. Wird ein vorbestelltes Buch zurückgegeben, wird der Vorbesteller darüber informiert, und das Buch geht sofort wieder in den Zustand “verliehen” über. Die Leihfrist startet dabei von vorn. Wegen des Fehlens von zusätzlichem Speicher müssen Sie diese Funktionalität durch Hinzufügen weiterer Zustände realisieren. Erweitern Sie die Übergangstabelle entsprechend. Beachten Sie dabei, dass Vorbestellungen auch für ein bereits gemahntes Buch funktionieren müssen. Warum kann ein endlicher Automat keine Warteliste von Vorbestellungen unterstützen?

Aufgabe 1.3 Verkehrsregeln

[12 Punkte]

Betrachten Sie eine Kreuzung von drei Straßen:



Es gibt hier sechs verschiedene Fahrmöglichkeiten: “kommt von A, will nach B”, “kommt von C, will nach A” usw.

- (a) Konstruieren Sie die Übergangstabelle für einen endlichen Automaten, der die Vorfahrtsregeln für die Situation “kommt von A, will nach B” implementiert, insbesondere die “rechts-vor-links”-Regel (ein von C kommendes Fahrzeug hat in dieser Situation Vorfahrt). Verwenden Sie dabei die Zustände “(an die Kreuzung) heranfahren”, “warten” und “weiterfahren” sowie die Ereignisse “Fahrzeug bei B”, “B ist frei”, “Fahrzeug bei C” und “C ist frei”. Ereignisse, die auf die Vorfahrt keinen Einfluss haben, können Sie natürlich ignorieren. Falls Sie die Vorfahrtsregeln nicht kennen sollten, wird es höchste Zeit, sie zu lernen: <https://de.wikipedia.org/wiki/Vorfahrt>.
- (b) Geben Sie weitere fünf Übergangstabellen für die Vorfahrtsregeln der fünf anderen Fahrmöglichkeiten an. Beachten Sie dabei auch die Regel, dass ein Rechtsabbieger warten muss, wenn ein Fahrradfahrer geradeaus durchfahren will.
- (c) Konstruieren Sie eine Situation mit mehreren beteiligten Fahrzeugen, bei der keiner jemals weiterfahren kann, d.h. alle Automaten bleiben im Zustand “warten” hängen. Man bezeichnet eine solche Situation in der Informatik als *deadlock* (dt. Blockierung). Beschreiben Sie ein realistisches Szenario, wie diese Situation im Straßenverkehr tatsächlich entstehen kann. Im wahren Leben müssen die Fahrer sich jetzt irgendwie einigen, bei Computern hilft nur noch abschalten. Deshalb sollte deadlock unbedingt vermieden werden. Die theoretische Informatik untersucht, wie man dies garantieren kann.

Bitte laden Sie Ihre Lösung spätestens bis 31. Oktober 2015, 11:00 Uhr in Moodle hoch.