Übungsblatt 06

zu bearbeiten bis 17.06.2010

Wirtschaftsinformatik und Finanzmathematik

Um Euch auch in dieser Woche ein Beispiel dafür zu geben, dass Euch die Programmierung an Stellen begegnen kann, an denen Ihr sie vielleicht gar nicht erwartet, beschäftigen wir uns auf diesem Übungsblatt schwerpunktmäßig mit einem Beispiel aus der BWL: der Kapitalwertmethode.

Hinter der Kapitalwertmethode verbirgt sich „nur“ eine Formel, die – sofern sie mit geeigneten Eingaben „gefüttert“ wird – auf ein sinnvolles Ergebnis führt. Formeln wie diese lassen sich in Programmiersprachen sehr einfach umsetzen, und im Berufsalltag eines Wirtschaftswissenschaftlers wird es sicherlich die ein oder andere Begegnung mit derartigen Formel geben. Wie immer gilt auch hier: es ist unerheblich, in welcher Sprache sie eines Tages umgesetzt werden muß – das Prinzip ist wichtig!

Neben den üblichen Verständnisfragen zu Begriffen aus Vorlesung und Übung der letzten 14 Tage enthält dieses Übungsblatt noch eine Automatenaufgabe, die fast genau in dieser Form (sogar noch ein wenig reichhaltiger) in einer Klausur in der Vergangenheit gestellt worden ist.

Aufgabe 1 – Die Kapitalwertmethode (16 Punkte)

Als Wirtschaftswissenschaftler/innen werdet Ihr früher oder später mit der Kapitalwertmethode konfrontiert. Sie ist ein Verfahren aus der dynamischen Investitionsrechnung und stellt die Vorteilhaftigkeit (oder auch die Nachteilhaftigkeit) eines beabsichtigten Investitionsprojekts dar. Aber wem erzählen wir das...? Sicherlich kennt Ihr diese Methode schon längst.

Heute wollen wir ein Java-Programm schreiben, das einen konkreten Anwendungsfall der Kapitalwertmethode realisiert: die Berechnung bei unterschiedlich hohen jährlichen Rückflüssen. Die diesem Anwendungsfall zugrundeliegende Formel haben wir Euch unten auf dieser Seite notiert. Die Bedeutungen der einzelnen Parameter findet Ihr auf der nächsten Seite.

$$C_0 = -I_0 + \sum_{t=1}^{T} R_t \cdot q^{-t} + L_T \cdot q^{-T}$$

Schreibt ein Java-Programm, das von der Standardeingabe die benötigten Informationen liest und danach tabellarisch ausgibt, wie die jährlichen Werte gemäß der Kapitalwertmethode sind. Zur „hübschen“ Ausgabe sollte Euer Programm Gebrauch von der in der Vorlesung vorgestellten Formatter-Klasse machen. Selbstverständlich möchten wir die Ausgabe – wie im kaufmännsischen Leben üblich – auf zwei Nachkommastellen gerundet.

Wie immer eignet sich diese Aufgabenstellung nicht dazu, sich einfach an den Rechner zu setzen und „mal d'rauf los“ zu programmieren. Sinnvoll ist es, sich zunächst Gedanken über die benötigte Struktur des Programms zu machen:

- Welche Variablen werden benötigt?
- Von welchem Datentyp müssen sie sein?
- Wie soll der Ablauf meines Programms sein?
- Mit Hilfe welcher Kontrollstruktur(en) kann ich diesen Ablauf am besten realisieren?
- Wie lässt sich die angegebene Formel in meinem Programm realisieren?

C_0	Kapitalwert (Barwert aller Zahlungen zum Zeitpunkt 0)	t	einzelne Zeitpunkte von 0 bis T
I_0	Investitionsauszahlungen zum Zeitpunkt 0	q^{-t}	Diskontierungsfaktor (Abzinsungsfaktor) mit q=1+i
R_t	Rückfluss („Cash-Flow“) in Periode t	i	Kalkulationszinssatz
L_T	Liquidationserlös / -auszahlung zum Zeitpunkt T (Ende der Laufzeit)		

- Welche Informationen benötige ich vom Benutzer?
- Welche Informationen kann / soll mein Programm selber errechnen?
- Wie kann ich mit Hilfe der Formatter-Klasse eine „schöne“ Ausgabe erzeugen?

Auf unserer Homepage findet Ihr das Programm „`Kapitalwert.class`“, das die Musterlösung enthält. Wenn Ihr dieses Programm ausführt (geeignete Beispieldaten findet Ihr ebenfalls auf der Homepage), bekommt Ihr einen Eindruck davon, was Euer Programm leisten soll. Eine diesbezügliche Erklärung gab/gibt es selbstverständlich auch in den Übungen.

Aufgabe 2 – Endliche Automaten (12 Punkte)

Diese Aufgabe wurde fast genauso (und sogar etwas ausführlicher) in der Klausur des letzten Jahres gestellt. Sie riecht also extrem nach Klausur!

Ein endlicher, deterministischer Automat definiert eine Sprache wie folgt:

„Beginnend im Startzustand werden die einzelnen Symbole des Vokabulars gelesen und die definierten Zustandsübergänge durchgeführt. Ist ein Zustandsübergang nicht definiert, so geht der Automat implizit in einen Fehlerzustand. Sind alle Symbole gelesen und ist der Automat in den Endzustand gelangt, so gehört die Symbolfolge zur Sprache, ansonsten gehört sie nicht zur Sprache.“

Der Startzustand ist `S`, der Endzustand ist `E`, und die Symbole des Vokabulars der Sprache sind `{'0','1'}`.

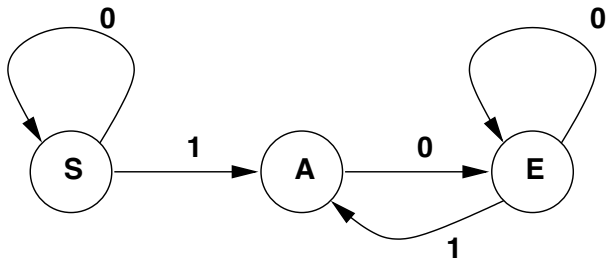
Gegeben sei nun der unten gezeigte Automat. Schreibt ein Java-Programm, das einen String von der Standardeingabe einliest und entscheidet (d.h. ausgibt), ob der gelesene String zur Sprache gehört oder nicht.

Aufgabe 3 – Fragen (2 Punkte)

1. Was ist eine Wrapper-Klasse und wozu wird sie benötigt?
2. Was unterscheidet eine Variable, die als `int` deklariert ist von einer als `Integer` deklarierten Variable?

Aufgabe 4 – Anmeldung zu den Vorleistungen

Bestätigt Eurem Tutor per Mail oder im persönlichen Gespräch, dass Ihr Euch zu den Vorleistungen im Hochschulportal angemeldet habt.

Ohne eine derartige ***fristgerechte*** Anmeldung ist eine Teilnahme an der Klausur (= Prüfung!) nicht möglich!

Weitere Informationen dazu – insbesondere zu den Fristen – haben wir Euch in der Vorlesung und Übung gegeben sowie auf der Homepage bereit gestellt.