Wintersemester 2006/2007
Universität Ulm,
Institut für Angewandte
Informationsverarbeitung

Prof. Dr. Franz Schweiggert
Norbert Heidenbluth

Übungsblatt Nr. 13

Abgabe bis spätestens 31.01.2007

In diesem Übungsblatt beschäftigen wir uns mit Methoden, einer zentralen Technik in allen Programmiersprachen. Synonym (in nicht objekt-orientierten Programmiersprachen) verwendet man übrigens auch die Begriffe „Funktion“, „Prozedur“ oder „Funktionsprozedur“.

Wofür dienen nun aber Methoden? Anstatt die gesamte Programmlogik in ein einziges (Haupt-)Programm zu packen, unterteilt man das Programm in einzelne Unterprogramm (Methoden), die jeweils eine ganz spezielle Aufgabe lösen. Ein großer Vorteil dabei ist die Wiederverwendbarkeit, denn eine Methode muß nur einmal geschrieben („implementiert“) werden, und ist dann beliebig oft zu benutzen.

Verwaltungssoftware (10 Punkte)

Heute versetzen wir uns für dieses Übungsblatt mal in eine kleine Märchenwelt, denn selbstverständlich ist die ganze Story sowie die Satzung frei erfunden und Ähnlichkeiten mit real existierenden Unis sind rein zufällig. (Obwohl...? Wer weiß...?):

Es war einmal eine kleine Uni „im Schwäbischen“, deren Rekt..., äh, Präsident(in) sich mit folgenden Problemen herumschlagen musste:

1.) Es gibt Unmut über die Studiengebühren und deren Verwendung (Stichwort: Heizkosten)

2.) Die Software für die Verwaltung der Studiengebühren funktioniert noch nicht.

In unserer Märchen-Uni gibt es ferner eine Satzung, die genau vorschreibt, wie und wofür Studiengebühren zu verwenden sind. Diese Satzung findet Ihr im Rahmen auf der nächsten Seite. Hier ganz kurz eine kleine Zusammenfassung: die von den Studierenden gezahlten Gebühren werden vom Land nochmals um 10% aufgestockt - dies ergibt die zu Verfügung stehende Summe. Davon dürfen maximal 5% für Heizkosten aufgewendet werden, zwischen 7 und 11% müssen in die Rücklagen gestellt werden, und mindestens 10.000, höchstens aber 20.000 Euro sind für Geschäftsreisen des Präsidiums reserviert. Die Summe dieser drei Posten darf aber die Hälfte der zur Verfügung stehenden Gebühren nicht überschreiten.

Eine eigens dafür geschaffene Planstelle in der Verwaltung soll nun ein wenig mit den Eingabeparametern jonglieren und prüfen, ob eine gegebene Kombination von Eingabeparametern satzungskonform ist. Zur Unterstützung dieser Aufgabe soll eine neue Software zum Einsatz kommen, die, wenn sie denn richtig läuft, die notwendigen Parameter abfragen würde und daraus ermittelt, ob die gewählten Parameter zu einer satzungskonformen Verwendung der Studiengebühren führen.

Der smarte „In-House-Consultant“ eines großen Softwarehauses bekommt das System aber einfach nicht ans laufen und ist nun erstmal für einige Wochen in Urla.., also zur Fortbildung. Aber da man an dieser Uni weiß, dass Ihr in „Allgemeine Informatik 1“ zur kompetenten Beherrschung der schwarzen Magie des Programmierens gedrillt werdet, bietet man Euch einen (lausig bezahlten) Hiwi-Job an, um die Software nun fertig zu stellen.

Vorhanden ist momentan der Programmrahmen sowie ein vollständiges Hauptprogramm. Dieses findet Ihr auf unserer Vorlesungshomepage zum herunterladen. Übersetzen lässt sich das Programm jedoch nicht, da alle Methoden, die im Hauptprogramm aufgerufen werden sollen, noch nicht vorhanden („implementiert“) sind. Und damit sind wir gleich bei der Aufgabe für diese Woche:

Übrigens...

... welches Wetter beschreibt ein Informatiker mit „Caps-Lock Wetter“ ?

Dauerregen...
Es SHIFT ohne Ende!

Vervollständigt das gegebene Programm um alle fehlenden Methoden, sodass es lauffähig ist und gemäß der Vorgaben in der Satzung die ordnungsgemäße Verwendung der Studiengebühren bescheinigt oder verneint.

Wichtig: Im gegebenen Hauptprogramm dürfen keinerlei Veränderungen vorgenommen werden! Desweiteren sollte es keine (unnötigen) Redundanzen im Quellcode geben, d.h. wiederkehrende Aufgaben sollten in Methoden ausgelagert werden.

Tips zum Vorgehen

Auch für diese Aufgabe geben wir einen Vorschlag, wie man sie am besten angeht.

Schritt 1.) Zunächst sollte man sich anschauen, welche Methoden im Hauptprogramm erwartet werden, welche Parameter sie erhalten und welchen Datentyp sie zurückgeben. Dieses ist wieder eine eher theoretische Arbeit.

Schritt 2.) Nun sollte man alle in Schritt 1 identifizierten Methoden im Programm deklarieren und lediglich mit einer beliebigen (aber zum Rückgabetyp passenden) return-Anweisung füllen. Man spricht hier auch von einer „Dummy“-Implementierung.

Schritt 3.) Der jetzige Stand des Programms sollte sich nun übersetzen (compilieren) lassen - ohne Fehlermeldung. Zwar scheitert die Ausführung des Programms am Fehlen jeglicher Programmlogik, aber wenn der Compiler keine Fehler meldet, ist man schon auf dem richtigen Weg.

Schritt 4.) Nun werden nacheinander die einzelnen Methoden mit richtigem Inhalt gefüllt („implementiert“), also mit der notwendigen Programmlogik versehen. Weil jede Methode nur eine ganz bestimmte Aufgabe zu erledigen hat und die sogenannten Schnittstellen - also die Eingabe-Parameter sowie der Rückgabetyp - bekannt sind, ist dieser Teil der Aufgabe innerhalb einer Tutoriengruppe hervorragend aufzuteilen.

Das war es soweit. Wenn alle Methoden implementiert sind und sich das Programm fehlerfrei übersetzen lässt, kann es ausgeführt und getestet werden. Fehler, die nun während des Ablaufes auftreten, haben ihre Ursache in der Programmlogik - nicht mehr jedoch im prinzipiellen Aufbau des Programms.

Satzung zur Verwendung von Studiengebühren

1.) Alle Parameter sind grundsätzlich ganze Zahlen.

2.) Die Höhe der Einnahmen errechnet sich aus dem Produkt der Anzahl der Studierenden mal der Höhe der Gebühren pro Studierender/m.

3.) Auf dieses Produkt gewährt das Land einen Zuschuß von derzeit 10%. Die Summe aus Einnahmen und Zuschuß bezeichne man als „Verfügbare Gebühren“.

4.) Von den verfügbaren Gebühren dürfen maximal 5% für Heizkosten aufgewendet werden.

5.) Von den verfügbaren Gebühren müssen mindestens 7%, maximal aber 11% den Rücklagen zugeführt werden.

6.) Für Geschäftsreisen des Präsidiums müssen mindestens 10.000 Euro, maximal aber 20.000 Euro der verfügbaren Gebühren reserviert werden.

7.) Nach Abzug der Posten gemäß Ziffer 4 bis 6 muß noch mindestens die Hälfte der verfügbaren Gebühren vorhanden sein. Diese werden dann „zweckgebunden“, also zum Beispiel zur Verbesserung der Lehre verwendet.

UNIX TIP der Woche

Und? Habt Ihr Gefallen am „screen“-Kommando gefunden, das auf dem letzten Blatt vorgestellt wurde? Dann kommt jetzt die gute Nachricht: Das war noch nicht alles: „screen“ kann noch mehr...

Nehmen wir an, Ihr habt von zu Hause aus (z.B. via Putty) auf einem unserer Rechner hier ein screen-Kommando gestartet und mehrere virtuelle Fenster erzeugt. Wenn Ihr nun die Tastenkombination „CTRL-A“ + D drückt (D für „detach“), dann könnt Ihr Euren Rechner daheim ausschalten - der screen läuft aber (auf unserem Rechner hier) weiter. Wenn Ihr dann an der Uni im Pool seid, braucht Ihr Euch bloß auf demselben Rechner erneut anzumelden und screen mit den Optionen „r“ und „d“ starten, also so:

```
screen -r -d
```

Und schon habt Ihr die von daheim „hinterlassene“ Konfiguration hier an der Uni. In die andere Richtung geht das natürlich auch.

Übrigens hatte sich im letzten Blatt ein Tippfehler eingeschlichen: Die Zählung der screen-Sitzungen beginnt bei 0, nicht bei 1. Sorry!