

Übungen zum Programmierkurs I

Abgabe bis zum Freitag, 06.12.2002, 20:00 Uhr

Bitte beachten Sie die allgemeinen Hinweise am Ende des Blattes!

Aufgabe 6.1 **Fibonaccizahlen (leicht)** **3 Punkte**

Die *Fibonacci-Zahlen* $(F_n)_{n \geq 0}$ sind eine Folge natürlicher Zahlen, die folgender Gesetzmäßigkeit gehorchen:

$$F(0) = 0, \quad F(1) = 1, \quad F(n+2) = F(n) + F(n+1) \text{ für } n \geq 0$$

Schreiben Sie ein Programm, welches für ein eingegebenes n den Wert von $F(n)$ bestimmt.

Hinweis: Lösen Sie die Aufgabe mit Schleifen.

Aufgabe 6.2 **Verbunde (mittel)** **8 Punkte**

Nachfolgend finden Sie eine (recht unvollständige) Liste von Politikern, die in jüngerer Zeit wichtige Ämter in Deutschland innehatten.

Name	Amtszeit	Amt
Kohl	1982–1998	Bundeskanzler
Schröder	1998–2002	Bundeskanzler
von Weizsäcker	1984–1994	Bundespräsident
Herzog	1994–1999	Bundespräsident
Rau	1999–2002	Bundespräsident
Honecker	1976–1989	Staatsratsvorsitzender

Entwerfen Sie einen Verbundtyp (**record**), der geeignet ist, die in der Tabelle enthaltenen Daten über einen Politiker zu speichern. Die obige Liste ließe sich dann als ein Feld (**array**) darstellen, dessen Einträge je einen solchen Verbund aufnehmen.

Auf der Vorlesungs-Webseite (siehe allgemeine Hinweise am Ende des Blatts) finden Sie eine Datei namens `politiker`, die eine längere Liste von Politikern enthält. Diese ist wie folgt aufgebaut: In der ersten Zeile steht die Anzahl der Politiker, die in der Datei stehen. Jeder Eintrag besteht dann aus vier Zeilen, die den Namen, den Beginn bzw. das Ende der Amtszeit und die Bezeichnung des Amtes beinhalten.

Schreiben Sie ein Programm, welches diese Datei einliest, ihren Inhalt in einem Feld abspeichert (wie oben dargelegt) und am Ende die Benutzer zur Eingabe einer Jahreszahl auffordert. Anschließend sollen alle in der Datei enthaltenen Politiker ausgegeben werden, die in dem eingegebenen Jahr ein Amt innehatten, zusammen mit der Bezeichnung ihres Amtes.

Hinweis: Dateien können Sie wie folgt einlesen: Benutzen Sie das Paket `Text_IO` und legen Sie mit `datei:File_Type`; eine 'Dateivariablen' an. Anschließend können Sie mit

`Open(datei, In_File, "politiker");` die besagte Datei öffnen. Wenn Sie jetzt eine Zeile einlesen wollen, in der eine Zeichenkette steht, machen Sie das mit

```
u:=Get_Line(datei);
```

falls `u` ein `Unbounded_String` ist (siehe Aufgabe 3.3). Wenn Sie eine Zeile mit einer Zahl einlesen möchten, tun Sie das am einfachsten mit den beiden Befehlen

```
Get(datei, zahl); Skip_Line(datei);
```

Daraufhin steht in `zahl` (welches vom Typ `Integer` sein sollte) die eingelesene Zahl. Mit `Close(datei);` können Sie das Einlesen der Datei beenden und diese 'schließen'.

Aufgabe 6.3 Primzahlen (mittel)

4+5 Punkte

Primzahlen sind die natürlichen Zahlen größer als 1, die nur durch 1 und sich selbst teilbar sind. Die erste Primzahl ist 2, die zweite ist 3, die 25. Primzahl ist 97 usw.

- Schreiben Sie ein Programm, das für die Eingabe n die n -te Primzahl ausgibt. Lassen Sie sich hiermit die Primzahlen für $n = 100$, $n = 1000$, $n = 10\,000$ und $n = 100\,000$ berechnen.
- Es sei p_i die i -te Primzahl und $d_i := p_{i+1} - p_i$ die Differenz zwischen der i -ten und der $(i+1)$ -ten Primzahl. Diese Differenzen wachsen im Mittel langsam an. Schreiben Sie ein Programm, welches die kleinsten Indizes i und die zugehörigen p_i ermittelt, für die $d_i \geq 10$, $d_i \geq 20$, $d_i \geq 50$, $d_i \geq 100$, $d_i \geq 150$ ist.

Hinweis: Um die Aufgabe zu lösen, dürfen Sie ein Array mit fester Größe von maximal 2^{16} Einträgen verwenden, um bereits gefundene Primzahlen zu speichern.

Allgemeine Hinweise

- **Geänderter Abgabetermin:** Jetzt bis Freitags 20:00 Uhr (zuvor: 15:00 Uhr).
- Aufgabenblätter stehen jeweils freitags auf der Webseite und werden Montag in der Vorlesung in Papierform verteilt; sie sind jeweils bis zum nächsten Freitag zu lösen. Die Webseite zum Programmierkurs ist wie folgt:
<http://www.informatik.uni-stuttgart.de/ifi/fk/lehre/ws02-03/ada95/>
Auf der Webseite finden Sie auch Musterlösungen zu vergangenen Aufgabenblättern und Folien (soweit elektronisch verfügbar). Die kompletten Folien finden Sie im Semesterapparat in der Bibliothek.
- Der Besuch der Übungen ist grundsätzlich Pflicht, da Sie in der Lage sein müssen, Ihre Programme auf Aufforderung in der Übung zu erklären. Sind Sie einmal aus gutem Grunde verhindert, sagen Sie (sofern möglich) zuvor Ihrem Tutor Bescheid. **Sie riskieren andernfalls, dass Ihnen die Punkte für das jeweils besprochene Aufgabenblatt aberkannt werden!**
- Pro Aufgabenblatt werden maximal 20 Punkte auf den Übungsschein angerechnet.
- Falls Sie Fragen irgendwelcher Art haben, sei es zu den Aufgaben, zum Ablauf, oder falls Sie mit dem Stoff Probleme haben, wenden Sie sich bitte an Ihren Tutor oder an die Übungsleitung:
Stefan.Schwoon@informatik.uni-stuttgart.de, Raum 0.019, oder Tel. 7816-427