Dr. Andreas Borchert
Michael Wiedemann
Abteilung Angewandte Informationsverarbeitung
17. Juni 2004
Blatt 6

Allgemeine Informatik II (SS 2004)

Abgabetermin: 24. Juni 2004

7 Zufalls-Zuteilung – 10 Punkte

Nehmen wir einmal an, Ihr seid die Organisatoren eines sportlichen Großereignisses, nennen wir es der Einfachheit halber Fußball-Europameisterschaft. Ihr sollt eine Anzahl von Ländern nach gewissen Vorgaben einteilen. Dabei wollen wir uns nicht an so antiquierte Methoden wie Losen halten, sondern das ganze mit den geschickteren Methoden bewerkstelligen. Nichts leichter als das, denken wir uns, wozu haben wir während meines Studiums Informatik gelernt. Außerdem wollt Ihr natürlich noch andere Ereignisse organisieren, deswegen wollen wir uns im Vorfeld nicht auf eine fest vorgegebene Anzahl an Teilnehmern festlegen.

Nun denn, schreiben wir also ein kleines Oberon-Programm, das mit linearen Listen und Pointern arbeitet. Der Programmaufruf soll folgendermaßen aussehen:

Einteilung [-i input] [-s select] [-u unselect] count

- Sämtliche Flags sind optional. Das nicht-optionale Argument *count* gibt die Anzahl der Einteilung an (Am Beispiel der Fußball-EM: 4 – jede Vorrunden-Gruppe der EM hat vier Teilnehmer).
- -i input: aus der Datei *input* sollen die Rohdaten zeilenweise gelesen werden (z.Bsp. alle Teilnehmer der Fußball-EM).
- -s select: in die Datei *select* werden die ausgewählten Teilnehmer geschrieben (im Beispiel eben genau vier).
- -u unselect: der Rest der Liste wird in die Datei *unselect* geschrieben. Wird die Option -u nicht angegeben, soll die Ausgabe des Restes unterdrückt werden.
- Bei einem Aufruf wird eine Einteilung vorgenommen. Das bedeutet, bei 16 Mannschaften stehen nach dem Aufruf Eures Programms mit count = 4 in *select* 4 Mannschaften, in *unselect* 12 Mannschaften (bei entsprechendem Aufruf mit allen Flags).

Generelle Tipps zum Programm:

- Wie immer wird ohne Angabe von den entsprechenden Flags (und somit von den Dateien) von der Standardeingabe gelesen bzw. an die Standardausgabe geschrieben (außer bei der Option -u). Zu Beginn solltet Ihr versuchen, die Einteilung einfach zu halten, das heißt, nehmt zunächst nacheinander einfach die Teilnehmer aus der vorhandenen Liste. Wer es fortgeschritten mag, kann versuchen, die Teilnehmer, die ausgewählt werden sollen, über einen Zufallsgenerator zu bestimmen. Dazu unten mehr.
- Ihr solltet überprüfen, ob die Angabe von *count* zulässig ist. Aus einer Liste von 8 Teilnehmern 10 auszuwählen, ist nicht erlaubt. Genauso wenig ist eine nicht-positive Angabe sinnvoll.
- Seid vorsichtig bei dem Umgang mit Pointern. Versucht Ihr innerhalb Eures Programms auf Daten hinter eines Zeigers zuzugreifen, der auf NIL zeigt, fängt dies kein Compiler ab. Stattdessen bekommt Ihr einen Fehler zur Laufzeit des Programmes (core dumped).
- Ihr könnt Euch an folgenden groben Ablauf halten:
  1. (leere) Liste initialisieren.
  2. Liste mit den eingelesenen Daten füllen bzw. aufbauen (in diesem Zusammenhang am besten zur Kontrolle diese Liste ausgeben).
  3. Liste bearbeiten, das heißt in diesem Fall, den geeigneten Teilnehmer aus der ursprünglichen Liste herauspicken und damit eine neue Liste füllen.
  4. Die veränderte Liste repräsentiert nun den Rest, die neue eure Auswahl. Diese beiden Listen gemäß Argumentverarbeitung gegebenenfalls ausgeben.
- Orientiert Euch an den Listenbeispielen im Skript.
- Zum Zufallsgenerator: angenommen, Eure Liste hat n=10 Teilnehmer. Am einfachsten holt Ihr Euch eine Zufallszahl mittels:
  SHORT(RandomGenerators.Val(0, n-1))
  Dies liefert Euch eine zufällige Integer-Zahl in den Grenzen [0-9] zurück. Dazu benötigt Ihr das Modul RandomGenerators.

Viel Erfolg!