Dr. Andreas Borchert Institut für Angewandte Informationsverarbeitung 26. Oktober 2009
Wolfgang Kaifler / Michael Mattes Blatt 3

Systemnahe Software I
im Wintersemester 2009/2010

Abgabetermin: bis 03. November 2009

9 Zahlen sortieren (16 Punkte)

Schreiben Sie ein Programm `bsort.c`, welches von der Standardeingabe `int`-Ganzzahlen einliest (bis zu einer über das Makro `MAXNUMBERS` von Ihnen definierten Maximalanzahl), sie mit Hilfe des *Bubblesort*-Algorithmus sortiert und dann auf der Standardausgabe in aufsteigend sortierter Reihenfolge ausgibt.

Ihr Programm soll in der Lage sein, zwei Fehlerzustände zu erkennen: Erstens kann es sein, dass ein Benutzer statt einer Zahl etwas anderes eingibt und das Einlesen somit fehlschlägt. Zweitens soll erkannt werden, wenn mehr als `MAXNUMBERS` Zahlen eingeben wurden. In beiden Fällen soll das Programm eine Fehlermeldung auf der *Standardfehlerausgabe* ausgeben und sich beenden.

Werten Sie das Ergebnis der zum Einlesen verwendeten Funktion `scanf` mit Hilfe einer `switch`-Anweisung aus.

Im Erfolgsfall soll das Programm auf der Standardfehlerausgabe die Anzahl der durchgeführten Vertauschungen ausgeben.

Beispiele hierzu (`MAXNUMBERS` hat da den Wert 10):

```
$ ./bsort
4 3 1 2
1 2 3 4

Fertig. Es wurden 5 Vertauschungen gemacht.
$ ./bsort
1 2 3 4 5 6 7 8 9 10 11
Es gibt mehr als 10 Zahlen in der Eingabe!
$ ./bsort
1 2 3 a 5
Es wurde etwas anderes als eine Zahl gelesen!
```

Hinweis 1: Speichern Sie die Zahlen in einem Array der Größe `MAXNUMBERS` (ähnlich zu Java; Tipp: `int zahlen[MAXNUMBERS];`) und merken Sie sich, wie viele Zahlen eingelesen wurden.

Hinweis 2: Ob die Eingabe zu Ende oder ein Fehler aufgetreten ist, kann man anhand des Rückgabewerts der Funktion `scanf` erkennen. Sie können diesen beispielsweise mit `int gelesen = scanf(...);` speichern. Bei Erfolg liefert `scanf` die Anzahl der Variablenzuweisungen zurück, bei Misserfolg ist die Zahl also kleiner als erwartet. Wenn die Eingabe zu Ende ist, hat der Rückgabewert den Wert des Makros `EOF` ("end of file").

Hinweis 3: Sie können die Zahlen jedes Mal von Hand eingeben (Strg-D auf einer neuen Zeile beendet die Eingabe) oder Sie können sich die *Eingabeumlenkung* der Shell mit Hilfe des Operators `<` zu nutze machen. Legen Sie hierzu eine neue Datei (mit der gewünschten Eingabe als Inhalt) an und führen Sie das Programm wie folgt aus: `./programmname <eingabedateiname`

Hinweis 4: Eine Beschreibung von Bubblesort finden Sie beispielsweise in der Wikipedia (hier ausschnittsweise kopiert):

Die Reihe wird in aufsteigender Richtung durchlaufen. Dabei werden immer zwei benachbarte Elemente betrachtet. Wenn sie in falscher Ordnung stehen, werden sie vertauscht. Am Ende steht das größte oder das kleinste (je nach Sortierung) Element am Ende der Reihe.

Der obige Schritt wird solange wiederholt, bis die Reihe komplett sortiert ist. Dabei muss das letzte Element des vorherigen Durchlaufs aber nicht mehr betrachtet werden, da es seine endgültige Position schon gefunden hat.

Je nachdem, ob auf- oder absteigend sortiert wird, steigen die größeren oder kleineren Elemente wie Blasen im Wasser (daher der Name) immer weiter nach oben, das heißt, an das Ende der Reihe. Auch werden immer zwei Zahlen miteinander in "Bubbles" vertauscht.

Viel Erfolg!