

Programmieren C: Arrays: Sortiert Einfügen und Löschen

Klaus Kusche

Gesucht ist ein Programm, das ein aufsteigend sortiertes Array von **int**'s > 0 verwaltet.

In einer Schleife soll das Programm immer wieder

- alle gespeicherten Zahlen der Reihe nach ausgeben
- und den Benutzer nach einer Zahl fragen (**scanf** oder **cin** $>>$... verwenden!):
 - Wird 0 eingegeben, soll das Programm enden.
 - Wird eine Zahl größer 0 eingegeben, wird diese Zahl im Array an der sortierungsmäßig richtigen Stelle eingefügt. Dieselbe Zahl darf auch mehrfach eingefügt werden.
 - Wird eine Zahl kleiner 0 eingetippt, wird die entsprechende Zahl > 0 gesucht. Wenn sie nicht vorhanden ist, soll eine Fehlermeldung kommen (aber das Programm soll weitermachen und nochmal fragen!). Wird die Zahl gefunden, soll sie aus dem Array gelöscht werden (bei mehrfach vorkommenden Zahlen wird jedes Mal nur eine davon gelöscht).

Hinweise:

- Das Programm wird ohne zusätzliche Angaben auf der Befehlszeile aufgerufen.
- Du darfst für dein Array eine fixe Maximalgröße (z.B. 100 Elemente) vorgeben (bitte verwende eine benannte Konstante!).
Wenn der Benutzer versucht, mehr Zahlen zu speichern, als das Array groß ist, soll eine Fehlermeldung ausgegeben und die neue Zahl nicht gespeichert werden. Das Programm soll aber trotzdem weitermachen.
- Du wirst dir in einer zusätzlichen Variable merken müssen, wie viele Zahlen gerade im Array gespeichert sind. Beim Einfügen kommt eine dazu, beim Löschen eine weg.
- Das Array enthält beim Start des Programmes noch keine Zahlen.
- Nach dem Löschen einer Zahl wirst du alle Zahlen hinter der gelöschten Zahl einzeln der Reihe nach um eine Stelle nach vorne kopieren müssen.
- Vor dem Einfügen einer Zahl wirst du alle Zahlen ab der Einfüge-Position einzeln der Reihe nach um eine Stelle nach hinten kopieren müssen.

Achte dabei auf die Richtung:

Beim nach vor Kopieren musst du von vorne nach hinten arbeiten,

beim nach hinten Kopieren von hinten nach vorne!

(Sonst überschreibst du dir die nächste Zahl, die du noch nicht kopiert hast!)

Mit anderen Worten:

- Beim Löschen lassen wir nach dem Entfernen des betreffenden Elementes die entstandene Lücke nach hinten wandern, bis sie "hinten hinausfällt".

- Beim Einfügen lassen wir vor dem Speichern des neuen Elementes eine Lücke von ganz hinten bis zu jener Stelle nach vorne wandern, an der wir das neue Element einfügen müssen.

Beim Einfügen sucht man daher am besten von hinten beginnend die richtige Stelle und verschiebt dabei gleich die Zahlen nach hinten.

- Beim Suchen für's Löschen solltest du nur so viele Zahlen wie nötig durchsuchen: Im Schnitt nur die Hälfte aller Zahlen, nicht alle! (Wann kannst du aufhören?)

Anmerkung:

Bei größeren Datenmengen ist es **nicht klug**, die Daten sortiert in einem Array zu speichern, wenn immer wieder einzelne Elemente eingefügt oder gelöscht werden sollen: Das Verschieben beim Einfügen und Löschen dauert viel zu lange!

Es gibt für diesen Zweck viel besser geeignete Methoden (z.B. binäre Bäume)!