Programmierung und Modellierung mit Haskell

Induktion & Korrektheit

Martin Hofmann Steffen Jost

LFE Theoretische Informatik, Institut für Informatik,
Ludwig-Maximilians Universität, München

24. April 2014

Induktionsprinzip: Beispiel 1

- Wir möchten zeigen, dass die Zahl der möglichen Kachelungen eines $n \times 2$ Korridors mit 2×1 Kacheln tatsächlich gleich F_n ist.
- . . . wobei $F_0 = F_1 = 1$ und $F_n = F_{n-1} + F_{n-2}$, falls $n > 1$.
- Für $n = 0, 1$ stimmt die Behauptung.
- Wenn $n > 1$, so gibt es zwei Möglichkeiten, die Kachelung zu beginnen: Man beginnt mit einer quergestellten Kacheln und hat dann noch eine $(n-1) \times 2$ Fläche zu kacheln.
- Oder man beginnt mit zwei längsgestellten Kacheln und hat dann noch eine $(n-2) \times 2$ Fläche zu kacheln.
- "per Induktion" oder "rekursiv" können wir annehmen, dass die Zahl der Möglichkeiten hierfür F_{n-1} bzw. F_{n-2} ist,
- Also ergeben sich $F_{n-1} + F_{n-2} = F_n$ Möglichkeiten, w.z.b.w.

Induktionsprinzip: Beispiel 2

- Wir möchten zeigen, dass der angegebene Algorithmus tatsächlich das Hanoi-Problem löst.
- Für $n = 1$ stimmt die Behauptung.
- Wenn $n > 1$, so schafft der erste Aufruf "per Induktion" die obersten $n - 1$ Scheiben auf den Hilfsstapel; der nächste Befehl schafft die unterste Scheibe ans Ziel; der zweite Aufruf schafft wiederum "per Induktion" die $n - 1$ Scheiben vom Hilfsstapel ans Ziel.

Induktionsprinzip: Beispiel 3

- Wir möchten zeigen, dass jede Befehlsfolge, die n Scheiben von einem Stapel auf einen anderen schafft mindestens $2^n - 1$ Befehle enthält.
- Für $n = 1$ stimmt das.
- Wenn $n > 1$, so muss irgendwann die unterste Scheibe bewegt werden und zwar auf einen freien Stapel. Das bedeutet, dass vorher die $n - 1$ darüberliegenden Scheiben auf den jeweils anderen verschafft wurden, damit dort Platz ist.
- "Per Induktion" erfordert das mindestens $2^{n-1} - 1$ Befehle.
- Anschließend müssen diese $n - 1$ Scheiben auch ans Ziel, macht noch einmal mindestens $2^{n-1} - 1$ Befehle, insgesamt also mindestens $2(2^{n-1} - 1) + 1 = 2^n - 1$ Befehle, w.z.b.w.

Das Induktionsprinzip

Ein Induktionsbeweis ist eigentlich ein rekursiver Beweis. Man darf die zu beweisende Aussage selbst benutzen. Allerdings muss die Kette solcher Rückgriffe irgendwann zum Ende kommen. Dafür bietet sich eine Abstiegsfunktion an.

Induktionsprinzip

Sei also A eine Menge, P eine Eigenschaft auf A (formal $P \subseteq A$) und $m : A \to \mathbb{N}$ eine Funktion, sodass für alle $a \in A$ gilt

"a ist in P unter der Annahme, dass alle $y \in A$ mit $m(y) < m(a)$ in P sind"

Dann ist $P = A$, also alle Elemente von A haben die Eigenschaft P.

Insbesondere muss natürlich für alle a mit $m(a) = 0$ direkt $a \in P$ gezeigt werden, da es ja in diesem Fall keine y mit $m(y) < m(a)$ gibt.

Beweis des Induktionsprinzips

Es seien A eine Menge, P eine Eigenschaft auf A (formal $P \subseteq A$) und $m : A \to \mathbb{N}$ eine Funktion, sodass für alle $a \in A$ gilt

"a ist in P unter der Annahme, dass alle $y \in A$ mit $m(y) < m(a)$ in P sind"

Wenn jetzt $P \neq A$, so sei $a \in A \setminus P$ ein Gegenbeispiel mit minimalem m-Wert. Für alle y mit $m(y) < m(a)$ gelte also $y \in P$. Nach Annahme gilt dann aber auch $a \in P$. Ein Widerspruch.

Anmerkungen

- Oft ist $A = \mathbb{N}$ und $m(a) = a$. Dann darf man also verwenden, dass die Behauptung $y \in P$ für alle $y < a$ schon gezeigt ist und hat dann $a \in P$ daraus herzuleiten.
- Bisweilen erlaubt man hier nur den Rückgriff auf $a - 1$ und nicht auf beliebiges $y < a$.
- Man kann auch Induktionsprinzipien für andere Maß-Werte als $\mathbb{N}$ betrachten. Es kommt nur darauf an, dass es jede echt absteigende Kette von Maß-Werten irgendwann zum Ende kommt. Beispiel: Paare von natürlichen Zahlen mit der lexikographischen Ordnung. I
- In den meisten Fällen kommt man aber mit $\mathbb{N}$-wertigen Abstiegsfunktionen aus.

Beispiel: Endständige Rekursion

```
fun f1(i,n,a) = if i=n then a else f1(i+1,n,a+i)
```

Man zeige durch Induktion, dass $\texttt{f1}(i, n, a) = a + \sum_{j=i}^{n-1} j$.

- Als Abstiegsfunktion nimmt man $m(i, n, a) = \max(n - i, 0)$.
- Falls $m(i, n, a) = 0$ so gilt $i = n$ und $\texttt{f1}(n, n, a) = a = a + \sum_{j=n}^{n-1} j$
- Falls $m(i, n, a) > 0$ so gilt $\texttt{f1}(i, n, a) = \texttt{f1}(i+1, n, a+i) = a+i+\sum_{j=i+1}^{n-1} j = a+\sum_{j=i}^{n-1} j$

Partielle Korrektheit

Oft hat man eine rekursive Funktion $f(a) = E(a, f)$ gegeben und möchte eine Aussage der Form $\forall a \in D.Q(a, f(a))$ zeigen, wobei D der Definitionsbereich von f ist.

Es genügt dann, für jedes a zu zeigen, dass $Q(a, f(a))$ erfüllt ist unter der Annahme, dass für alle rekursiven Aufrufe $f(a_1), \ldots, f(a_n)$, die in $E(a, f)$ vorkommen, bereits $Q(a_i, f(a_i))$ erfüllt ist.

Man kann nämlich dann als Abstiegsfunktion die Zahl der Auswertungsschritte nehmen. Auf dem Definitionsbereich ist diese erklärt und für alle rekursiven Aufrufe strikt kleiner. Slogan: Will

man eine rekursive Funktion zu verifizieren, so darf man die rekursiven Aufrufe bereits als korrekt annehmen.

Zusammenfassung

- Induktion als “rekursiver Beweis” mit überall definierter Abstiegsfunktion
- Beispiele: Kacheln, Hanoi, Endrekursion
- Partielle Korrektheit als Spezialfall der Induktion