

Determinismus eines Algorithmus

Das Wort "Determinismus" leitet sich vom lateinischen "determinare" ab, dies bedeutet so viel wie beeinflussen, bestimmen oder feststellen. Anhand dieser Übersetzung leiten sich schon sehr gut die Eigenschaften eines deterministischen Algorithmus ab.

Eine einleitende Frage zum Thema könnte sein:
Welche Schritte durchläuft ein Algorithmus nach einer Eingabe?

Es geht bei einem deterministischen Algorithmus also darum, welche Schritte er nach einer Eingabe durchläuft und welche Zwischenergebnisse produziert werden, bis es zu einer Ausgabe kommt.

Eine Definition könnte sein:

"Ein Algorithmus oder Programm wird durch Maschinen schrittweise abgearbeitet. Der Algorithmus heißt deterministisch, wenn es zu jeder Programmsituation eine nachfolgende Situation geben kann, wenn also zu jedem Zeitpunkt der Folgeschritt eindeutig bestimmt ist."

Ein deterministischer Algorithmus hat somit keine Wahl bei der Verarbeitung einer Eingabe und folgt immer dem gleichen Weg von der Eingabe bis hin zur Ausgabe. Alle nachfolgenden Berechnungsschritte liegen somit schon bei der Eingabe fest und können auch nicht umgangen werden. Somit sind alle Zwischenergebnisse und Werte, die durch diesen Algorithmus berechnet werden in jedem Durchlauf, für die gleiche Eingabe ebenfalls wieder die gleichen Werte.

Man darf dieses Prinzip nicht mit der "Determiniertheit" eines Algorithmus verwechseln. Dies bezieht sich auf das Ergebnis eines Algorithmus: Bei einer Eingabe steht also schon fest, was dabei heraus kommt. Bei jedem determinierten Algorithmus bildet der Eingabewert eindeutig auf das Ergebnis ab. Daraus lässt sich schlussfolgern, dass jeder deterministische Algorithmus determiniert sein muss, denn es wird immer die gleiche Verarbeitung bei einer Eingabe x durchgeführt.

Im Gegensatz dazu steht aber, dass nicht jeder determinierte Algorithmus deterministisch sein muss, ein Algorithmus kann auch unterschiedliche Schritte abarbeiten und doch kommt er jeweils zum gleichen Ergebnis.

Für die Erläuterung der Determiniertheit sollte folgendes Beispiel hilfreich sein. Der Sortieralgorithmus Quicksort, teilt die vorgegebene Liste immer in Teillisten ein, dafür wird ein Pivotelement benötigt. Dies kann zum Beispiel zufällig gewählt werden, ausgehend von diesem Element wird die geteilte Liste in weitere Listen geteilt und diese Teillisten werden dann sortiert. Die Schritte des Algorithmus und somit die Zwischenergebnisse sind somit noch nicht festgelegt, der Algorithmus ist nicht

deterministisch, jedoch steht am Ende eine sortierte Liste und somit ist der Algorithmus determiniert.

Zum Abschluss soll der Determinismus mit folgendem Beispiel aus dem Leben erläutert werden. Bei dem Bau und der Montage einer Windkraftanlage, sind alle Schritte vorbestimmt. Zunächst wird ein Fundament ausgehoben, um einen sicheren Stand zu gewährleisten. In den weiteren Schritten werden die konischen Teilsegmente, des Turmes zusammen gefügt, man beginnt mit dem größten (breitesten) und arbeitet sich zum schmalsten nach oben, um dann letztendlich den Generator auf den Turm zu setzen und daran die Flügel zu montieren. Bis zur Montage der Flügel arbeitet der Algorithmus deterministisch, da man keine Wahlfreiheit hat bei den Schritten der Montage. Die Flügel können jedoch beliebig angebracht werden, also es spielt keine Rolle welcher Flügel zu erst angebracht wird. Dieser Schritt ist somit nicht deterministisch.

Quelle

Definition: Informatik Duden 3.Auflage 2001

Matrikel 113113