

# Primitiv-rekursiv

(Primitiv-rekursiv)

Stichwörter: Berechenbarkeit

## Primitiv-rekursiv

Lösungsvorschlag

Begründe, dass folgende Funktionen primitiv-rekursiv sind: 1.  $f(x) = x!$  (1 wenn  $x > 0$ )  
2.  $\text{sig}(x) = 0$  sonst

Lösungsvorschlag

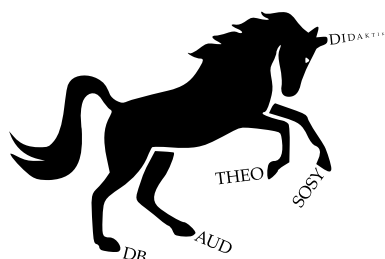
Begründung durch eine Angabe einer Funktion: 1.  $f(0) = 1, f(n+1) = \text{mult}(n+1, f(n))$  2.  $\text{sig}(0) = 0, \text{sig}(n+1) = 1$

Gebe eine konkrete primitiv-rekursive Implementierung für `if x1 then x2 else x3` an. Wo- bei wie bei Programmiersprachen `x1` als wahr gilt, wenn der Wert nicht Null ist.

Lösungsvorschlag

Zusätzlich werden die folgenden Funktionen festgelegt:

$\text{isZero}(0) = 1, \text{isZero}(n) = \text{isZero}(n+1) = 0, \text{not}(n) = 1 - n, \text{ite}(x1, x2, x3) = \text{isZero}(x1) * x3 + \text{not}(\text{isZero}(x1)) * x2$



### Die Bschlangaul-Sammlung

Hermine Bschlangauland Friends

Eine freie Aufgabensammlung mit Lösungen von Studierenden für Studierende zur Vorbereitung auf die 1. Staatsexamensprüfungen des Lehramts Informatik in Bayern.



Diese Materialsammlung unterliegt den Bestimmungen der Creative Commons Namensnennung-Nicht kommerziell-Share Alike 4.0 International-Lizenz.

Hilf mit! Die Hermine schafft das nicht allein! Das ist ein Community-Projekt! Verbesserungsvorschläge, Fehlerkorrekturen, weitere Lösungen sind herzlich willkommen - egal wie - per Pull-Request oder per E-Mail an [hermine.bschlangaul@gmx.net](mailto:hermine.bschlangaul@gmx.net). Der TeX-Quelltext dieses Dokuments kann unter folgender URL aufgerufen werden: [https://github.com/bschlangaul-sammlung/examens-aufgaben/blob/main/Module/70\\_THEO/20\\_Berechenbarkeit/Aufgabe\\_Primitiv-rekursiv.tex](https://github.com/bschlangaul-sammlung/examens-aufgaben/blob/main/Module/70_THEO/20_Berechenbarkeit/Aufgabe_Primitiv-rekursiv.tex)