Prof. Dr. Manfred Schmidt-Schauß
Künstliche Intelligenz/Softwaretechnologie

Fachbereich Informatik (20)
Johann Wolfgang Goethe-Universität Frankfurt am Main

Einführung in das funktionale Programmieren:

Sommersemester 1997

Übungsblatt Nr. 3

Aufgabe 1

Programmieren Sie nützliche Listenfunktionen wie sie in Haskell zur Verfügung stehen in **KFP**. Z.B. `tail`, `init`, `last`, `reverse`, `take`, `drop`, `splitAt`, `takeWhile`, `dropWhile`, `map`, `foldr`, `filter`.

Aufgabe 2

Geben Sie zu den Ausdrücken die Reduktionen bei den Auswertungsstrategien AO und NO an. Definieren Sie noch fehlende Funktionen.

1. `double` (`square` 4)
2. `take` 10 (`map double primes`)
3. `const Win` (`head []`)

Aufgabe 3

In der Vorlesung wurde gezeigt, wie man ohne Konstruktoren in **KFP** auskommt. Programmieren Sie in dieser Art die dort angegebenen Funktionen für positive und negative ganze Zahlen.

Aufgabe 4

Zeigen Sie, daß die Funktion `plus` in Aufgabe 3 korrekt ist.

Aufgabe 5

Lassen Sie die **KFP** Funktionen, die Sie bisher definiert haben in Gofer/Hugs auswerten. Sind Änderungen nötig? Was passiert bei der Auswertung der Funktionen aus Aufgabe 3?