

Übungen zur „Deskriptiven Programmierung“ Blatt 1

Aufgabe 1. Implementiere den Datentyp „Wörterbuch“ unter Verwendung geordneter Listen.

type *FMap* *key* *val*

Statt von „Wörterbüchern“ spricht man auch von *endlichen Abbildungen*. Im einzelnen sollen die folgenden Funktionen realisiert werden.

Konstruktion:

empty :: *FMap* *k* *v*
add :: (*Ord* *k*) ⇒ (*k*, *v*) → *FMap* *k* *v* → *FMap* *k* *v*
addWith :: (*Ord* *k*) ⇒ (*v* → *v* → *v*) → (*k*, *v*) → *FMap* *k* *v* → *FMap* *k* *v*
union :: (*Ord* *k*) ⇒ *FMap* *k* *v* → *FMap* *k* *v* → *FMap* *k* *v*
unionWith :: (*Ord* *k*) ⇒ (*v* → *v* → *v*) → *FMap* *k* *v* → *FMap* *k* *v* → *FMap* *k* *v*

Ein Wörterbuch assoziiert mit jedem Schlüssel höchstens einen Wert; wenn mit *add* ein Schlüssel hinzugefügt wird, der bereits im Wörterbuch enthalten ist, so wird der alte Werte überschrieben. Bei *addWith* kann man angeben, wie der alte und der neue Wert kombiniert werden. Entsprechendes gilt für *union* und *unionWith*.

Modifikation:

delete :: (*Ord* *k*) ⇒ *k* → *FMap* *k* *v* → *FMap* *k* *v*
deleteWith :: (*Ord* *k*) ⇒ (*v* → *Maybe* *v*) → *k* → *FMap* *k* *v* → *FMap* *k* *v*
intersect :: (*Ord* *k*) ⇒ *FMap* *k* *v* → *FMap* *k* *v* → *FMap* *k* *v*
intersectWith :: (*Ord* *k*) ⇒ (*v* → *v* → *v*) → *FMap* *k* *v* → *FMap* *k* *v* → *FMap* *k* *v*
amap :: (*k* → *v* → *v'*) → *FMap* *k* *v* → *FMap* *k* *v'*

Bei *deleteWith* *f* *a* wird die Funktion *f* auf den mit *a* assoziierten Wert angewendet: ist *f* *v* gleich *Nothing*, so wird der Eintrag gelöscht; ist das Ergebnis *Just* *v'*, so wird *v* durch *v'* ersetzt. Die Funktion *intersect* bildet den Durchschnitt zweier Wörterbücher; bei *intersectWith* kann zusätzlich angegeben werden, wie Werte kombiniert werden, die in beiden Wörterbüchern auftreten. Was macht *amap*?

Extraktion:

lookup' :: (*Ord* *k*) ⇒ *k* → *FMap* *k* *v* → *Maybe* *v*
lookupWithDefault :: (*Ord* *k*) ⇒ *FMap* *k* *v* → *v* → *k* → *v*
fromList :: (*Ord* *k*) ⇒ [(*k*, *v*)] → *FMap* *k* *v*
toList :: (*Ord* *k*) ⇒ *FMap* *k* *v* → [(*k*, *v*)]

Schlägt man mit *lookup'* einen Schlüssel nach, der nicht im Wörterbuch enthalten ist, so wird *Nothing* zurückgegeben, anderenfalls der assoziierte Wert in der Form *Just* *v*. Bei *lookupWithDefault* wird ein Default-Wert angegeben, der zurückgegeben wird, wenn der Schlüssel nicht im Wörterbuch enthalten ist.