

## 4.6 Kind Inference

本節では、種の推論、つまり、プログラム中の型構築子、および、クラスそれぞれに適切な種を計算するための規則について述べる。

種の推論プロセスにおける初めのステップは、データ型、シノニム、および、クラス定義の集合を依存グループに分けることである。これは、4.5 節で述べた変数宣言の依存解析とほぼ同様の方法で達成できる。例えば、以下のプログラム片はデータ構築子  $D$ 、シノニム  $S$  とクラス  $C$  を含むが、これらは全て一つの依存グループに入る。

```
data C a => D a = Foo (S a)
type S a = [D a]
class C a where
    bar :: a -> D a -> Bool
```

各グループにおける変数、構築子、および、クラスの型は、型推論と種保存単一化の標準的なテクニックを用いて決定することができる<sup>8\*1</sup>。例えば、上の定義においてパラメータ  $a$  は  $bar$  型において関数構築子 ( $\rightarrow$ ) の引数として出現しており、したがってその種は  $*$  に違いない。さらに  $D$  と  $S$  はいずれも  $* \rightarrow *$  の種を持ち、クラス  $C$  のインスタンス全ての種は  $*$  となる。

対応する定義から、全ての種推論が一意には定まらない可能性もあり、そのような場合には、デフォルトの種として  $*$  が仮定される。例えば、以下の例におけるパラメータ  $a$  は、いずれも任意の種  $\kappa$  を仮定することができる。

```
data App f a = A (f a)
data Tree a = Leaf | Fork (Tree a) (Tree a)
```

そうすると、 $App$  と  $Tree$  の種は、 $\kappa$  を任意の種として、それぞれ  $(\kappa \rightarrow *) \rightarrow \kappa \rightarrow *$  と  $\kappa \rightarrow *$  となり、かつ、種に関する多相を許すような拡張が必要となる。そうする代わりに、デフォルトの束縛  $\kappa = *$  を用いることで、これらの2つの構築子の種はそれぞれ  $(* \rightarrow *) \rightarrow * \rightarrow *$  と  $* \rightarrow *$  となる。

---

<sup>\*1</sup> <https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.120.6646&rep=rep1&type=pdf>