

TPFI 2023/24

Hw 4: Effectful Haskell

assegnato: 14 maggio 2024, consegna 26 maggio 2024

1. Input/Output

Definite un'azione `charCount :: IO ()` che legge un numero n da tastiera, poi n stringhe e alla fine stampa il numero di stringhe in cui appare ciascuna lettera.

2. Nodi Equilibrati con Applicativi e Monadi

Risolvere l'esercizio 3 dell'Homework 2 (Nodi Equilibrati) usando applicativi e monadi (in analogia con le funzioni che creano un albero o che rietichettano i nodi dei un albero visti nella Lezione), in modo da evitare di dover usare (esplicitamente nel codice) parametri e risultati di ritorno ausiliari.

3. Monadi/Eccezioni

Definire il tipo `NatBin` che rappresenta i numeri naturali come sequenze binarie. Potete definirlo come liste (di lunghezza fissata) di 0 e 1, oppure potete dare una definizione con `data` (ad esempio usando 3 costruttori, di cui uno sia la costante 0 e gli altri due... in ogni caso, immaginare di definire una “parola di memoria”, quindi prevedete una lunghezza massima costante).

Definire un valutatore di espressioni aritmetiche su `NatBin`, analoghi a quelli visti a lezione, ma considerare tutte le operazioni aritmetiche (+, ×, `div`, `mod` e -). Estendere il tipo `Maybe` in modo che il risultato di un'espressione possa essere eventualmente un'eccezione diversa a seconda dell'eventuale situazione anomala che si è verificata: divisione per zero, numero negativo oppure overflow.

Potete completare l'esercizio facendo in modo che il tipo `NatBin` sia un'istanza delle usuali classi `Eq`, `Ord`, `Num`, `Show`.