

Далее подразумевается, что элементы типа, от которого зависит ваш АДТ, являются элементами требуемых классов.

- ❶ (Вариант 1) Сделать `Multiset` экземпляром классов `Ord`, `Show`, `Eq`, `Read`, а также моноидом относительно объединения. Проверить свойства моноида. Согласовать `join` в моноиде и `bind`.
- ❷ (Вариант 2) Сделать `DoubleList` экземпляром классов `Ord`, `Show`, `Eq`, `Read`, а также моноидом относительно конкатенации. Проверить свойства моноида. Согласовать `join` в моноиде и `bind`.
- ❸ (Вариант 3,4) Сделать `EvalTree` (`EvalATree`) экземпляром классов `Ord`, `Show`, `Eq`, `Read`, и `Foldable`. Проверить, что `foldr` — естественное преобразование.
- ❹ (Вариант 5) Сделать `SuperList` экземпляром классов `Ord`, `Show`, `Eq`, `Read` и `Foldable`. Проверить, что `foldr` — естественное преобразование.
- ❺ (Вариант 6) Сделать `ErrorTree` экземпляром классов `Ord`, `Show`, `Eq`, `Read` и `Foldable`. Проверить, что `foldr` — естественное преобразование.
- ❻ (Вариант 7) Определить моноид на `ContR`. Проверить согласованность `join` и `bind`.

Далее подразумевается, что элементы типа, от которого зависит ваш АДТ, являются элементами требуемых классов. В вариантах 3 и 4 используем сужение типов  $a = b$  и считаем, что при отображении в строку в нелистовых узлах можно писать результаты вычислений (т.е. результат вычисления поддерева с корнем в рассматриваемом узле), а при чтении строки — читать имена функций.

Проверку свойств моноидов или `foldr` осуществить в любом модуле фазз-тестирования (на ваш выбор), например, `QuickCheck`.

Часть функций классов `Eq` и `Ord` реализовать через рекурсию, часть — через отложенные вычисления. Проверить выполнение алгебраических законов `Eq` и `Ord` над ними.

Функции `Read` желательно реализовать через механизм аппликативных парсеров (лекция 7).



- Свой вариант лабораторной номер 2 или лабораторной номер 1 (на выбор) реализовать на языке Haskell с обработкой ошибок. Если входные данные некорректны, выдать сообщение о том, на каком входном символе произошла ошибка.
- Если используются варианты 2 ЛР номер 2, 4, 5, 7 — дополнительно добавить REPL-среду. При этом уже просчитанные варианты вычислений желательно хранить в монаде State, чтобы доставать их оттуда по повторному запросу.