

ФИТ НГУ, курс Haskell, осенний семестр 2018

Задача 0. Развлечения со списками

Общие сведения:

Требуется реализовать набор аналогов стандартных библиотечных операций со списками, для глубокого понимания их работы и механизма рекурсии.

Необходимые функции:

1. `get xs n` — получим n -й произвольный элемент списка `xs`, начиная с нулевого;
2. `head xs` — вернет первый элемент списка `xs`;
3. `last xs` — вернет последний элемент списка `xs`;
4. `tail xs` — вернет список `xs` без первого элемента;
5. `init xs` — вернет список `xs` без последнего элемента;
6. `reverse xs` — вернет обратный список;
7. `length xs` — вернет длину списка `xs`;
8. `append xs x` — добавит `x` в качестве последнего элемента к списку `xs`;
9. `concat x sys` — конкатенация двух списков `list1` и `list2`;
10. `drop n xs` — вернет список, где удалены первых n элементов из списка `xs`;
11. `take n xs` — вернет список из первых n элементов из списка `xs`;
12. `splitAt n xs` — вернет пару списков, полученных из списка `xs` разбиением с n -й позиции (причем n -й элемент войдет во второй список);
13. `null xs` — проверяет, пуст ли список `xs` (возвращает `True` или `False`);
14. `elem xs x` — проверит, лежит ли элемент `x` в списке `xs` (возвращает `True` или `False`);
15. `filter test xs` — вернет только те элементы списка, которые удовлетворяют условию `test` (с аналогичными выше требованиями);
16. `map f xs` — применит функцию `f` ко всем элементам списка `xs` и вернет новый список;
17. `zip xs ys` — вернет список пар вида (x,y) , в каждой из которых первый элемент `x` является очередным элементом первого списка `xs`, а второй элемент `y` — второго списка `ys`;

Технические детали:

1. Все функции должны быть реализованы в файле `lists.hs`, проверяться будут загрузкой в GHCi.
2. Никакие другие функции не должны «торчать» из Вашего файла, только

обозначенные в списке выше.

3. Для почти всех функций ожидается реализация через pattern matching, рекурсию и особый синтаксис «отщепления» головы списка.
4. Все функции должны укладываться в сложность $O(n)$, некоторые – в $O(1)$, где n – длина списка.
5. **Важным** условием является самостоятельный вывод и указание правильного типа перед каждой функцией, неправильно написанный тип влечет к его правке.
6. Запрещено пользоваться любыми библиотечными функциями для работы со списками кроме синтаксиса расщепления, но разрешается для более сложных функций использовать собственноручно реализованные простые функции.

Сроки, дедлайны и штрафы:

1. При отсутствии 70% функций на гитхабе по истечению срока, Вы получаете пачку дополнительных функций посложнее для реализации.
2. За честно скомунизженные функции у коллег\гугла – см. пункт 1.
3. Сдача будет проходить лично. Невозможность нормально объяснить, как работает та или иная функция, приравнивается по степени греха к пункту 1.