

Лабораторная работа №2

Алгебраические типы данных

Задание

Написать скрипт, реализующий задание (см. ниже, по вариантам) и демонстрирующий работоспособность функций. Тестовые (демонстрационные) данные могут находиться в самом скрипте.

Можно использовать как `Either` и `Option`, так и `Sealed traits` + `case objects` по необходимости.

Напоминаю, что для отладки в режиме REPL можно подгрузить созданные вами функции командой `:load` либо напрямую вставить код с помощью команды `:paste`

Варианты

1. Определите тип данных, представляющий информацию о карте в карточной игре. Каждая карта характеризуется одной из четырех мастей. Карта может быть младшей (от двойки до десятки), либо картинкой (валет, дама, король, туз).
Определите функции:
 - a. `isMinor` - проверяет, что её аргумент является младшей картой
 - b. `sameSuit` - проверяет, что переданные в неё карты - одной масти
 - c. `beats` - проверяет, что карта, переданная ей в качестве первого аргумента, бьёт карту, являющуюся вторым аргументом
 - d. `beats2` - аналогично `beats`, но принимает третьим аргументом козырную масть
 - e. `beatsList` - принимает в качестве аргумента список карт, карту и козырную масть; возвращает список карт из первого аргумента, которые бьют указанную карту с учетом козырной масти
 - f. функция, возвращающая по заданному списку карт список чисел, каждое из которых является возможной суммой очков указанных карт, рассчитанных по правилам игры “блэк джек”: младшие карты считаются по номиналу,

валет, дама и король - 10 очков, туз может рассматриваться как 11 или как 1 (возвращаются все возможные варианты)

2. Определите тип, представляющий геометрические фигуры на плоскости. Фигура может быть: окружностью (характеризуется координатами центра и радиусом), прямоугольником (характеризуется координатами левого верхнего и правого нижнего углов), треугольником (координаты вершин). Определите функции:

- a. `area` - возвращает площадь фигуры
- b. `getRectangles` - из списка фигур выбирает и возвращает только прямоугольники
- c. `getBound` - по заданной фигуре возвращает ограничивающий прямоугольник
- d. `getBounds` - по списку фигур возвращает ограничивающий прямоугольник
- e. `getFigure` - по списку фигур и точке возвращает первую фигуру, для которой заданная точка попадает в ограничивающий прямоугольник (может никуда не попасть)
- f. `move` - по заданной фигуре и вектору сдвига возвращает новую фигуру, сдвинутую на указанный вектор

3. В агентстве недвижимости продают квартиры, комнаты и частные дома. Квартира характеризуется этажом, площадью и этажностью дома. Комната характеризуется, помимо этого, площадью комнаты (в дополнение к площади всей квартиры). Частный дом характеризуется только площадью. В “базе данных” хранятся пары значений, первое из которых представляет собой объект недвижимости, а второе - его цену. Определите тип данных, представляющий информацию о таких объектах недвижимости. Определите следующие функции:

- a. `getHouses` - выбирает из “базы данных” только частные дома
- b. `getByPrice` - выбирает из “базы данных” только те объекты недвижимости, цена которых НИЖЕ указанной
- c. `getByLevel` - выбирает из “базы данных” квартиры, находящиеся строго на указанном этаже

- d. `getExceptBounds` - выбирает из “базы данных” квартиры, не находящиеся на первом и последнем этажах
 - e. `query` - выбирает из “базы данных” объекты недвижимости по списку требований (желаемый тип объекта, минимальная площадь, максимальная цена, ограничения на этаж); для списка требований разработайте отдельный тип данных.
4. В библиотеке хранятся книги, газеты и журналы. Книга характеризуется именем автора и названием; журнал - названием, месяцем и годом выпуска; газета - названием и датой выпуска. “База данных” представляет собой список таких объектов. Разработайте тип данных, представляющий объекты библиотечного хранения. Определите следующие функции:
- a. `isPeriodic` - проверяет, что аргумент - периодическое издание
 - b. `getByTitle` - выбирает объекты с указанным именем
 - c. `getByMonth` - выбирает из “базы данных” периодические издания, выпущенные в указанный месяц и указанный год (газеты выходят несколько раз в месяц)
 - d. `getByMonths` - аналогично предыдущей, но для списка месяцев
 - e. `getAuthors` - возвращает список всех авторов изданий, хранимых в “базе данных” (без повторов)
5. В некотором языке программирования существуют следующие типы данных: а) простые (целые, вещественные и строки); б) сложные (структуры, которые имеют названия и состоят из нескольких полей, каждое из которых имеет название и простой тип). “База данных” идентификаторов программы представляет собой список пар, состоящих из имени идентификатора и его типа. Разработайте тип данных, представляющий данную информацию. Определите следующие функции:
- a. `isStructured` - проверяет, что аргумент является сложным типом
 - b. `getType` - по заданному имени типа и списку идентификаторов (“базе данных”) возвращающая тип идентификатора с указанным именем

- c. `getFields` - по заданному имени возвращает список полей идентификатора, если он имеет тип структуры
 - d. `getByType` - возвращает список имен идентификаторов указанного типа из “базы данных”
 - e. `getByTypes` - аналогично, но принимает список типов
6. Определим следующий набор операций над строками: очистка (удаление всех символов из строк), удаление (удаление всех вхождений указанного символа), замена (замена всех вхождений одного символа на другой), добавление (добавление в начало строки указанного символа). Разработайте тип данных, характеризующий операции над строками. Определите основные функции:
- a. `process` - получает в качестве аргумента действие и строку, возвращает строку, модифицированную в соответствии с указанным действием
 - b. `processAll` - аналогично предыдущей, но получает список действий и выполняет их по порядку
 - c. `deleteAll` - принимает две строки, и удаляет из второй строки все символы первой (использовать при реализации `processAll`)
7. В электронной записной книжке хранятся записи следующих видов: напоминания о днях рождения знакомых, телефоны знакомых и назначенные встречи. Напоминание состоит из имени знакомого и даты (день и месяц). Запись о телефоне должна содержать имя человека и его телефон. Информация о назначенной встрече содержит дату встречи (день, месяц и год) и краткое описание (строка). Разработайте тип данных, представляющий такую запись. Записная книжка является списком записей. Определите следующие функции:
- a. `getByName` - возвращает информацию о человеке с указанным именем (телефон и дату рождения)
 - b. `getByLetter` - возвращает список людей, о которых есть информация в записной книжке и чье имя начинается на указанную букву
 - c. `getAssignment` - возвращает по указанной дате список дел (информация о назначенных встречах и телефоны друзей, которых нужно поздравить в этот день)
8. Клавиши на клавиатуре могут быть либо управляющими, либо алфавитно-цифровыми. Нажатие алфавитно-цифровой

клавиши может сопровождаться нажатием клавиши Shift. Из управляющих клавиш нас интересует только клавиша CapsLock, остальные можно не различать. Каждое нажатие алфавитно-цифровой клавиши несет с собой информацию в виде символа. После нажатия CapsLock последующие символы переводятся в верхний регистр (если они не были нажаты вместе с Shift) до следующего нажатия CapsLock. Если CapsLock не активирован, Shift переводит символы в верхний регистр. Разработайте тип данных, представляющий указанную информацию. Последовательность нажатий клавиш представляется в виде списка. Основная задача состоит в том, чтобы разработать функцию getString, переводящую эту последовательность в строку символов. Например, последовательность нажатий Shift + 'h', 'e', CapsLock, 'l', 'l', Shift + 'o', CapsLock даст в результате строку HeLLo. Определите следующие функции (при реализации можно использовать методы toUpperCase, toLowerCase):

- a. getAlNum - возвращает из списка нажатий только нажатия алфавитно-цифровых клавиш
- b. getRaw - возвращает строку, составленную из нажатых символов без учета информации о Shift и CapsLock
- c. isCapsLocked - по последовательности нажатий определяет, остался ли CapsLock в активном состоянии
- d. getString - переводит последовательность нажатий в строку

9. За время учебы в семестре студенты должны сдать определенное количество лабораторных работ, расчетно-графических заданий и рефератов. Лабораторная работа характеризуется названием предмета и номером, РГЗ - названием предмета, реферат - названием предмета и названием темы реферата. Разработайте тип данных, представляющий информацию по заданию. Учебный план студента представляет собой список из пар, первый элемент которых является заданием, а второй - номером недели, в который оно должно быть сдано. Если задание ещё не сдано, второй элемент - Unit. Определите следующие функции:

- a. getByTitle - возвращает задания, которые необходимо сдать по данному предмету
- b. getReferats - возвращает список тем рефератов

- c. `getRest` - возвращает список не сданных заданий
- d. `getRestForWeek` - возвращает список заданий, оставшихся не сданными на указанной неделе
- e. `getPlot` - создает список, состоящий из пар, первый элемент которых равен номеру недели, второй - количество сданных за эту неделю заданий