**Федеральное агентство связи**

**Ордена Трудового Красного Знамени федеральное государственное бюджетное общеобразовательное учреждение высшего образования «Московский технический университет связи и информатики»**

Кафедра математической кибернетики и информационных технологий

Лабораторная работа №2

«Классы типов и рекурсивные функции»

Выполнила: студентка группы БВТ1702

Абрамова Ирина

Москва 2019

**Задание:** ознакомиться с классами типов и рекурсивными функциями в языке Haskell

**Выполнение:**

**Классы типов**

1. Класс Eq Класс Eq используется для типов, которые поддерживают проверку равенства. Типы, являющиеся его экземплярами, должны реализовывать функции == и /=.

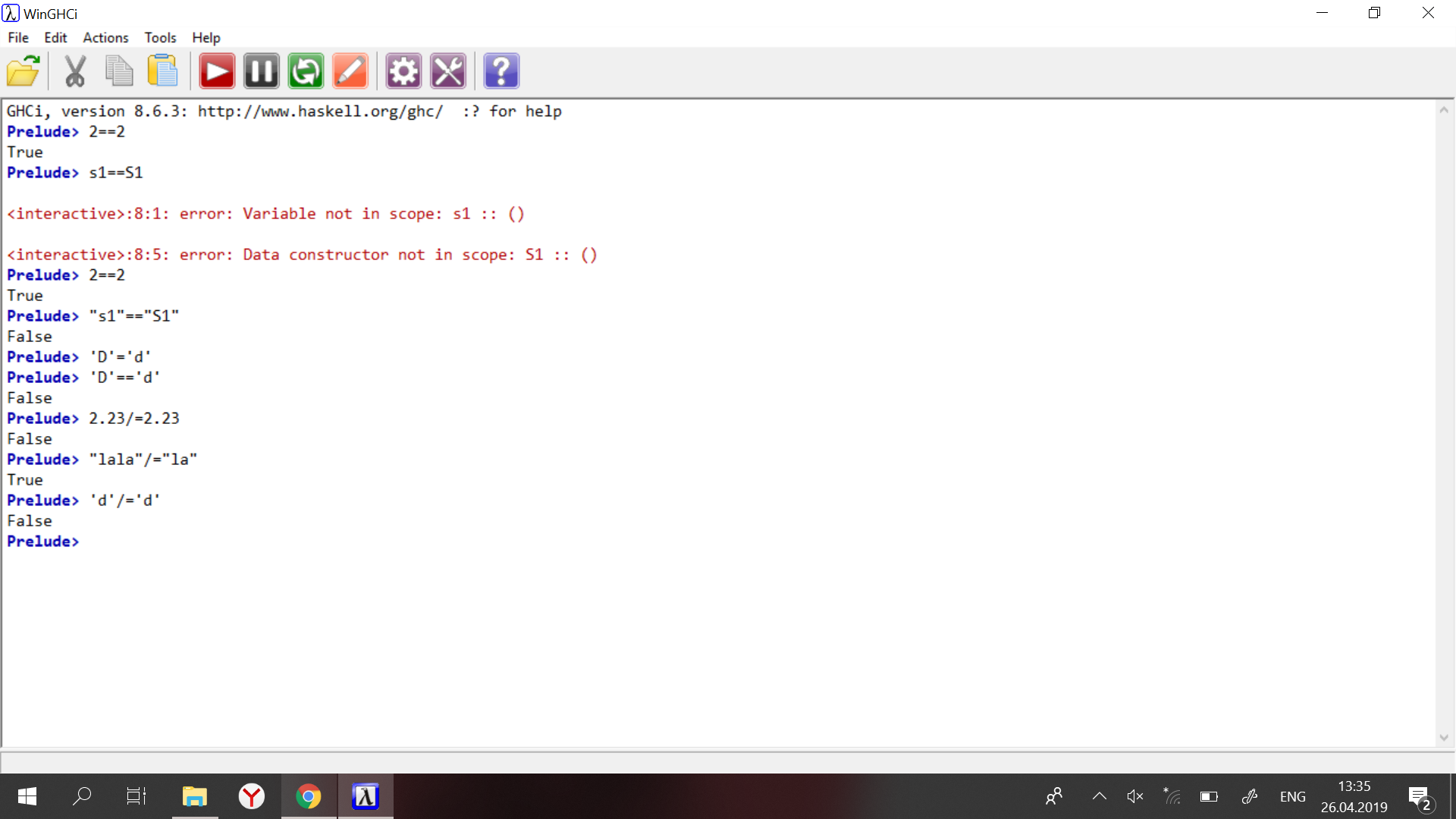


Рисунок 1 – Класс Eq

1. Класс Ord предназначен для типов, которые поддерживают отношение порядка. Класс Ord содержит все стандартные функции сравнения, такие как >, <, >= и <=. Функция compare принимает два значения одного и того же типа, являющегося экземпляром класса Ord, и возвращает значение типа Ordering. Тип Ordering может принимать значения GT-больше чем, LT-меньше чем или EQ-равно.

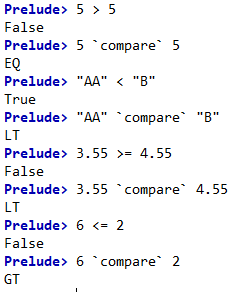


Рисунок 2 – Класс Ord

1. Класс Show

Значения, типы которых являются экземплярами класса типов Show, могут быть представлены как строки. Наиболее часто используемая функция в классе типов Show – это, собственно, функция show. Она берёт значение, для типа которого определён экземпляр класса Show, и представляет его в виде строки.

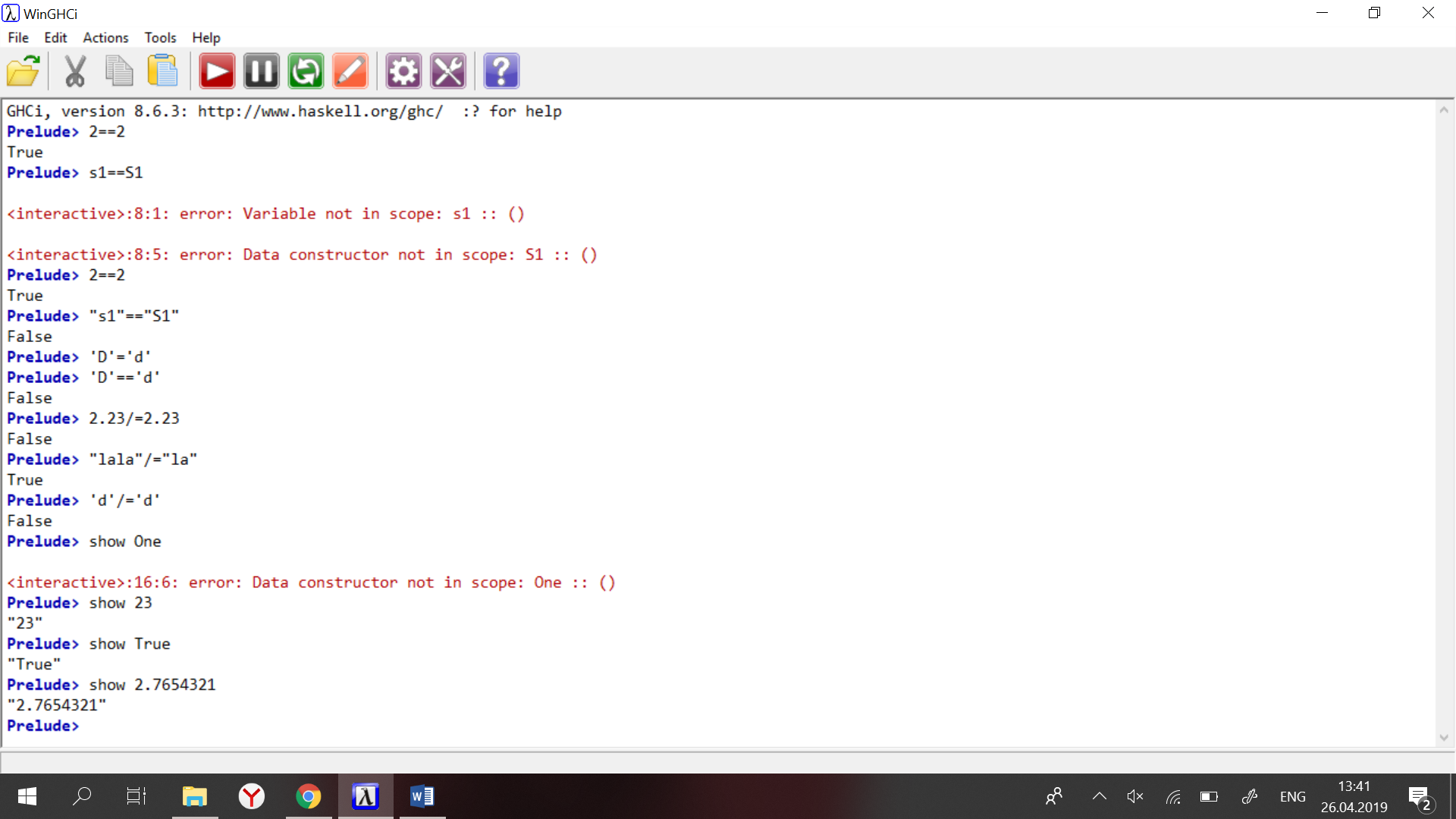


Рисунок 3 – Класс Show

1. Класс Read

Функция read принимает строку и возвращает значение, тип которого является экземпляром класса Read.

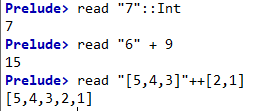


Рисунок 4 – Класс read

1. Класс Enum

Экземплярами класса Enum являются последовательно упорядоченные типы; их значения можно перенумеровать. Основное преимущество класса типов Enum в том, что мы можем использовать его типы в интервалах списков. Кроме того, у них есть предыдущие и последующие элементы, которые можно получить с помощью функций succ и pred. Типы, входящие в этот класс: (), Bool, Char, Ordering, Int, Integer, Float и Double

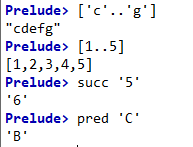


Рисунок 5 – Класс Enum

1. Класс Bounded

Экземпляры класса типов Bounded имеют верхнюю и нижнюю границу.

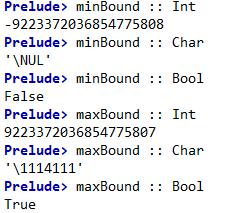


Рисунок 6 – Класс Bounded

1. Класс Num – это класс типов для чисел. Его экземпляры могут вести себя как числа.

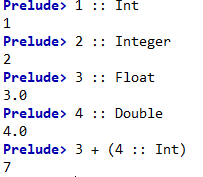


Рисунок 7 – Класс Num

1. Класс Floating

Класс Floating включает в себя только числа с плавающей точкой, то есть типы Float и Double. Некоторые примеры: функции sin, cos и sqrt.

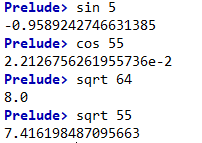


Рисунок 8 – Класс Floating

1. Класс Integral

Класс Integral – тоже числовой класс типов. В класс Integral входят только целые числа. Для типов Int и Integer определены экземпляры данного класса.

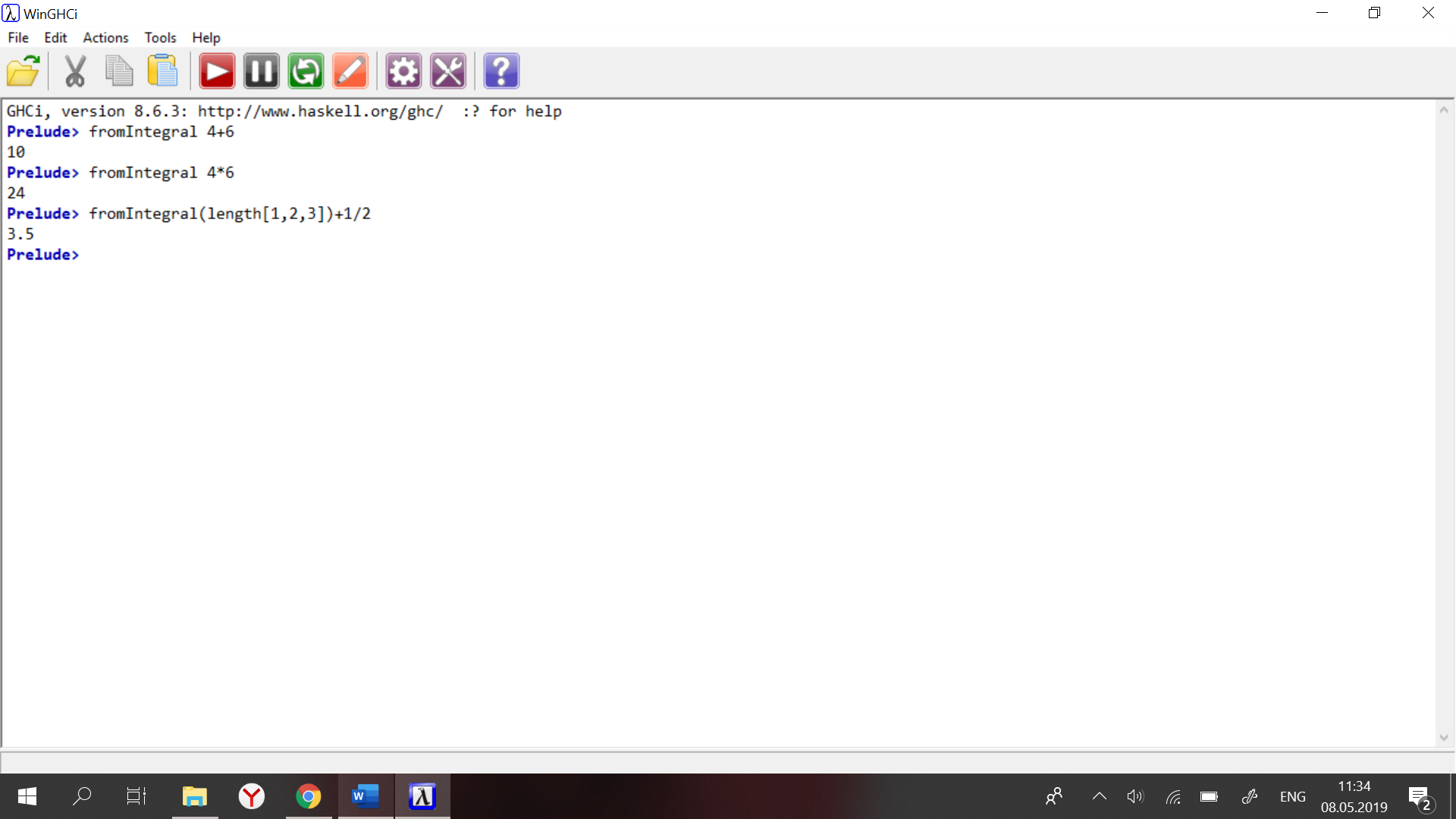


Рисунок 9 – Класс Integral

**Рекурсивные функции**

1. Функция replicate

Функция replicate берёт целое число (типа Int) и некоторый элемент и возвращает список, который содержит несколько повторений заданного элемента.

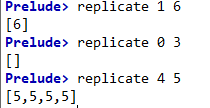


Рисунок 10 – Функция replicate

1. Функция take

Эта функция берёт определённое количество первых элементов из заданного списка.

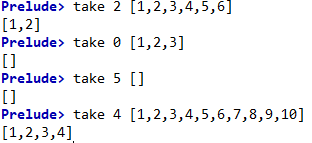


Рисунок 11 – Функция take

1. Функция reverse

Функция reverse обращает список, выстраивая элементы в обратном порядке.

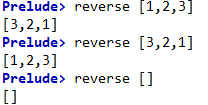


Рисунок 12 - Функция reverse

1. Функция repeat

Функция repeat принимает на вход некоторый элемент и возвращает бесконечный список, содержащий этот элемент.

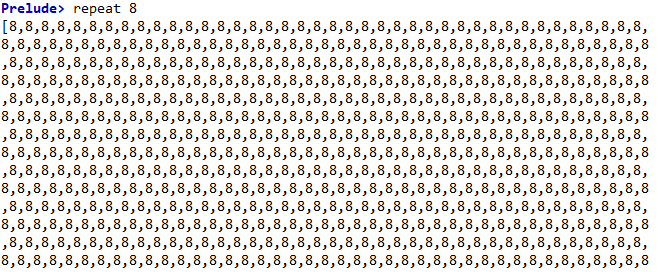


Рисунок – Функция repeat



Рисунок 14 – Использование функции take от repeat

1. Функция zip

Функция zip берёт два списка и стыкует их, образуя список пар

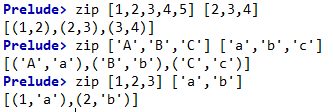


Рисунок 15 – Функция zip

1. Функция elem

Она принимает элемент и список и проверяет, есть ли заданный элемент в этом списке

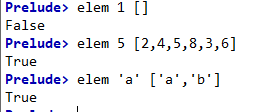


Рисунок 16 – Функция elem