**Лабораторная работа № 1**

Задание 1. Напишите функцию get\_fractions, которая принимает на вход два числа, m и n, и возвращает аналогичный вектор, содержащий все дроби вида {i/m, i = 0, 1, ..., m} и  {j/n, j = 0, 1, ..., n}. Вектор не должен содержать повторов и должен быть упорядочен в порядке убывания.

Задание 2. Вычислите значения следующих выражений:

* -1:1;
* as.logical(-1:1);
* as.character(as.logical(-1:1));
* as.numeric(as.logical(-1:1));
* as.character(as.numeric(as.logical(-1:1)));
* as.numeric(as.character(as.logical(-1:1))).

Задание 3. Выполните в сессии R следующие команды:

set.seed(1337)  
x <- runif(1e6, min = -1, max = 1)

Теперь в вашем распоряжении вектор x длиной один миллион. Определить, сколько среди них чисел в диапазоне (-0.2, 0.3). Для определённости не включая границы интервала.

Задание 4. Напишите функцию dice\_roll(n), которая должна выдавать n независимых бросков игрального кубика. Допустимые значения находятся в диапазоне от 1 до 6.

Задание 5. Пусть имеется вектор x произвольного типа. Напишите конструкцию, которая может быть использована, чтобы получить содержимое вектора x без каждого седьмого элемента?

Задание 6. Пусть имеется следующий шифр: 1, 23, 5, 19, 15, 13, 5. При помощи одного из встроенных массивов в R, расшифруйте эпитет, применимый к языку R. Регистр не имеет значение.

Задание 7. Пусть вектор называется нестрого возрастающим, если каждый следующий элемент в нём не меньше, чем предыдущий. Точно так же, вектор назовём нестрого убывающим, если каждый следующий элемент в нём не больше, чем предыдущий. Напишите функцию, которая принимает один аргумент (числовой вектор) и возвращает TRUE, если вектор обладает свойством нестрогой монотонности, то есть является либо нестрого возрастающим, либо нестрого убывающим. В противном случае функция возвращает FALSE.

Пример. x=c(0, 0, 3, 4, 4, 8) нестрого возрастает, возвращаем TRUE.

y=c(3:0, 1) был бы нестрого убывающим, если бы не последняя единица, поэтому возвращаем FALSE.

Задание 8. Пусть у нас есть n предметов, из которых нужно выбрать k штук.

Известнейшая комбинаторная формула задаёт количество всевозможных сочетаний. Похожий вид имеет и количество сочетаний с повторениями (мультикомбинаций).

Запрограммируйте оба этих значения в виде функции, зависящей от n и k. Аргумент with\_repetitions будет отвечать за вариант подсчёта: если он FALSE, то пусть считается количество сочетаний, а если TRUE, то сочетаний с повторениями.

Задание 9. Предположим, что у нас есть целочисленный вектор v и число n. Наша задача — найти позицию элемента в векторе, который ближе всего к числу n. При этом если таких элементов несколько, необходимо указать все позиции.

Напишите функцию, которая принимает на вход вектор и число и возвращает вектор индексов, отвечающих указанному условию. Индексы должны быть выстроены по возрастанию.

Пример. Пусть v <- c(5, 2, 7, 7, 7, 2, 0, 0) и n=1. Ответом будет вектор 2 6 7 8.

Задание 10. Напишите функцию, которая принимает одно целое число n, а возвращает «ступенчатую» матрицу, состоящую из n этажей. Этажи нумеруются с первого, ширина каждой ступеньки равна одной строке или столбцу.

Пример. Пусть n=4, тогда необходимо вернуть матрицу вида

  1   1   1   1   1   1   1  
  1   2   2   2   2   2   1  
  1   2   3   3   3   2   1  
  1   2   3   4   3   2   1  
  1   2   3   3   3   2   1  
  1   2   2   2   2   2   1  
  1   1   1   1   1   1   1