

## Вариант 3.9.

Все консольные приложения Ruby следует реализовывать в виде трех отдельных файлов:

1. основная программа;
2. программа для взаимодействия с пользователем через консоль;
3. программа для автоматического тестирования на основе MiniTest::Unit или RSpec. Везде, где это возможно, данные для проверки должны формироваться автоматически по правилам, указанным в задании.

При реализации программ везде, где это возможно, следует избегать использования циклов `for`, `do`, `while`. Вместо них используйте методы из примеси `Enumerable`.

Все тексты программ должны быть проверены на соответствие стилю программирования Ruby при помощи `rubocop` или `reek`.

## ЛР 5

### Часть 1

Вычислить:  $y = \frac{x^2 - 10}{tg(6 \cdot z - 6 \cdot x)}$ .

### Часть 2

Данные о температуре воздуха хранятся в виде двух массивов, где попарно представлены дата и температура. Определить, сколько раз температура опускалась ниже  $-10$  градусов за количество дней, введенных с клавиатуры. Распечатать в виде таблицы эти даты и температуры.

### Часть 3

Дана последовательность строк. Каждая строка состоит из слов, разделенных пробелами. Написать программу, обеспечивающую ввод строк и их корректировку. Корректировка заключается в обмене местами слов максимальной и минимальной длины в каждой строке. Если среди слов строки есть несколько таких слов, то в обмене участвуют только первые обнаруженные слова. Вывести на печать исходную и скорректированную последовательности строк.

Автоматический тест программы обязательно должен генерировать случайные строки в соответствии с правилами, перечисленными в задании.

## ЛР 6

### Часть 1

Решить задачу, организовав итерационный цикл с точностью  $\xi = 10^{-3}, 10^{-4}$ .

Вычислить определенный интеграл методом трапеций:  $\int_0^1 x\sqrt{1-x^2} dx$ .

Точное значение интеграла равно  $\frac{1}{3}$ . Определить, как изменяется число отрезков деления при изменении точности.

### Часть 2

Решить предыдущее задание с помощью Enumerable или Enumerator.

### Часть 3

Составить метод scale отыскания масштаба графического изображения функции  $f(x)$  на экране размером  $B$  единиц раstra по формуле  $M = \frac{B}{\max f(x)}$ . В основной программе использовать метод для отыскания масштаба функций  $x \cdot \sin(x)$  и  $tg(x)$ , при  $|x| < 1$ .

Реализовать вызов метода двумя способами: в виде передаваемого lambda-выражения и в виде блока.

## ЛР 7

### Часть 1

Организовать программным способом символьные файлы **F** и **G**. Переписать в файл **H** все начальные совпадающие компоненты файлов **F** и **G**. При возникновении непредвиденных ситуаций выдать соответствующие сообщения.

Автоматический тест программы обязательно должен проверять работу с файлами.

## Часть 2

Разработать и реализовать иерархию классов для описанных объектов предметной области, используя механизмы наследования. Проверить ее на тестовом примере, с демонстрацией всех возможностей разработанных классов на конкретных данных.

Объект — линия, заданная координатами двух точек. Умеет выводить свои параметры на экран и определять, проходит ли через третью точку.

Объект — линия, заданная координатами двух точек и толщиной. Умеет выводить свои параметры на экран и определять, проходит ли через третью точку с учетом толщины.

В тестирующей программе обеспечить автоматическую проверку того, что созданные объекты действительно соответствуют заданной иерархии классов.

## ЛР 8. Ruby on Rails

Разработать веб-приложение, имеющее HTML-страницу с формой ввода данных и HTML-страницу для представления результатов. Результат расчёта должен быть представлен в форме таблицы, оформленной с помощью элемента `table` или отдельными ячейками `div` и имеющей не менее двух колонок. Если по условию задания результат может быть представлен только в виде одной строки таблицы, необходимо реализовать вывод промежуточных результатов расчёта в качестве дополнительных строк. В этом случае первой колонкой таблицы будет порядковый номер итерации.

Под вводом с клавиатуры в тексте заданий следует понимать ввод в поле ввода данных формы на HTML-странице.

### Текст задания:

Дано натуральное число  $n$ . Написать программу, определяющую, есть ли среди чисел  $n, n + 1, n + 2, \dots, 2n$  близнецы, то есть простые числа, разность между которыми равна 2. Например, 11 и 13, 17 и 19. Вывести на печать все обнаруженные пары, а если таких пар нет, то соответствующее сообщение. Для определения близнецов и их наличия использовать функции.