

## Вариант 4.1.

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`.  
Везде, где это возможно, данные для проверки должны формироваться автоматически по правилам, указанным в задании.

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

## ЛР 5

### Часть 1

Вычислить:  $a = x(\cos(z) + e^{-(x+3)})$ .

### Часть 2

Дана последовательность строк. Каждая строка состоит из слов, разделенных пробелами. Написать программу, обеспечивающую ввод строк и их корректировку. Корректировка заключается в следующем. В каждой строке вычеркнуть все слова, не содержащие гласных букв и подсчитать количество таких слов. Если среди слов строки нет таких слов, то вывести сообщение. Вывести на печать исходную и скорректированную последовательности строк и полученное количество искомых слов.

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

## ЛР 6

### Часть 1

Вычислить  $\arctg(x) = x - \frac{x^3}{3} + \frac{x^5}{5} - \frac{x^7}{7} + \dots$  в точке  $x = 0,5$  и с точностью  $\xi = 10^{-3}, 10^{-4}$ . Определить, как изменяется число итераций с изменением точности.

## Часть 2

Решить предыдущее задание с помощью Enumerable или Enumerator.

## Часть 3

Составить метод `peibr` проверки принадлежности точки плоскости с координатами  $(x, y)$  данной кривой  $y = f(x)$ . В основной программе использовать метод `peibr` для проверки принадлежности десяти различных точек кривым  $y = \cos(x)$  и  $y = \sin(x^2)$ .

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

## ЛР 7

### Часть 1

Дан символьный файл **F**. Найти самое длинное слово среди слов, первая буква которых есть «а». Если слов с наибольшей длиной несколько, то найти последнее из них. Если таких слов нет вообще, то сообщить об этом.

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 — доска. Параметры: два целых числа: длина и ширина. Методы: инициализирующий, определения площади и возвращающие значения.

Объект — ящик. Параметры: длина, ширина и высота. Методы: инициализирующий, определения объема и вывода параметров.

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:

Аutomorphic numbers are called numbers, which are contained in the last digits of their squares. For example:  $5^2 = 25$ ,  $25^2 = 625$ . Write a program, determining automorphic numbers in the range 1 to  $n$  ( $n$  is entered from the keyboard and  $n \leq 100$ ). Output to the screen all found numbers. Use functions in programming.