

## Вариант 2.27.

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:  $y = \frac{\sin(a) - b}{|b| + \cos(b^2)}$ .

### Часть 2

Дана последовательность строк. Каждая строка состоит из слов, разделенных пробелами. Написать программу, обеспечивающую ввод строк и их корректировку. Корректировка заключается в удалении или замене слов. Если слово стоит на четном месте и начинается на букву «е» — оно удаляется; если слово стоит на четном месте и начинается на букву «с», оно замещается на слово, введенное с клавиатуры. 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. Вычислить сумму ряда:  $S = \sum_{k=1}^{\infty} \frac{1}{k(k+1)}$  расчет продолжать до тех пор, пока приращение  $R$  будет больше заданной точности  $\xi < 10^{-3}, 10^{-4}$ . Определить, как изменится число итераций при изменении точности.

## Часть 2

Решить предыдущее задание с помощью Enumerable или Enumerator.

## Часть 3

Составить метод differ для вычисления производных функции  $Y(X)$  в некоторых 3 соседних точках, отстоящих на величину шага  $h$ . Для вычислений использовать формулы Лагранжа:  $y'_0 = \frac{-3y_0 + 4y_1 - y_2}{2h}$ ;  $y'_1 = \frac{-y_0 + y_2}{2h}$ ;  $y'_2 = \frac{y_0 - 4y_1 + 3y_2}{2h}$ , где  $y_0, y_1$  и  $y_2$  – координаты точек. В основной программе использовать метод differ для вычисления производных функций  $\sin(x)$  и  $\lg(x+1)$  в точках 0.49, 0.5 и 0.51.

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

## ЛР 7

### Часть 1

Организовать программным способом файл **F**, компонентами которого являются целые числа, отличные от 0. Числа в файле идут в следующем порядке: десять положительных, десять отрицательных и т.д. Переписать компоненты файла **F** в файл **P** так, чтобы числа расположились в следующем порядке:

1. пять положительных, пять отрицательных и т.д.
2. двадцать положительных, двадцать отрицательных и т.д.

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 символ, умеющий выводить на экран свое значение и отвечать на запрос о значении своего символьного поля и числовом коде символа.

Объект запись, состоящая из полей: символ и целое число — количество повторений символа при выводе. Объект умеет выводить на экран строку из указанного количества символов, возвращать по запросу содержимое своих полей и код символа.

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:

Число из  $n$  цифр является числом Армстронга, если сумма цифр этого числа, возведенных в степень  $n$  равна самому этому числу. Например:  $153 = 1^3 + 5^3 + 3^3$ . Написать программу, определяющую все числа Армстронга, состоящие из  $n$  цифр ( $n$  вводится с клавиатуры). Вывести на печать сами числа их количество.