

Вариант 4.2.

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: $y = tg \frac{x^2 \cdot (x - 2)}{e^x}$.

Часть 2

С клавиатуры вводится две строки: в первой — слова, разделенные пробелами, во второй — числа, разделенные пробелами. Написать, программу, обеспечивающую ввод строк и их корректировку. Корректировка заключается в следующем. Каждому слову из первой строки ставится в соответствие число из второй, их соответствие определяется порядковым номером (первое слово — первое число, второе слово — второе число и т.д.). Числа представляют желаемую длину слов. Если длина слова больше желаемой, сократить длину слова до желаемой длины, если меньше или равна — оставить без изменений. Вывести скорректированную строку. Например, первая строка «Мама мыла раму», вторая строка «2 2 10», результат «Ма мы раму».

Указание: при выполнении задания рекомендуется использовать метод `zip` класса `Array`.

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 $S = \sum_{k=1}^{\infty} \frac{1}{k(k+1)(k+2)(k+3)}$ с точностью $\xi = 10^{-2}, 10^{-3}$. Точное значение: $\frac{1}{3 \cdot 3!}$. Определить, как изменяется число итераций при изменении точности.

Часть 2

Решить предыдущее задание с помощью Enumerable или Enumerator.

Часть 3

Составить метод root отыскания минимального корня уравнения $f(x) = 0$ с точностью 0.1 на отрезке $[a, b]$. В основной программе использовать метод для решения уравнений $x^2 + \sin \frac{x}{2} = 0$ и $\arctg(x) + x = 1$.

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

ЛР 7

Часть 1

Дан текстовый файл **F**. Записать в перевернутом виде строки файла **F** в файл **G**. Порядок строк в файле **G** должен быть обратным по отношению к порядку строк исходного файла **F**.

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 — окно, хранящее размеры и умеющее выводить на экран площадь.

Объект — окно, хранящее размеры и сведения о наличии жалюзи и умеющее выводить на экран площадь и признак наличия жалюзи.

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:

Написать программу, определяющую количество целых чисел $\leq n$ (n вводится с клавиатуры), квадрат которых является палиндромом. Например: $26^2 = 676$. Вывести на печать числа, квадраты и их количество.