

Вариант 2.26.

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: $y = \frac{\cos(x + 3.1 \cdot z)}{tg(x/r)}$.

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 нечетном месте и начинается на букву «л» — оно удаляется; если слово стоит на четном месте и начинается на букву «н», 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 с точностью $\xi = 10^{-3}, 10^{-4}$.
Вычислить значение определенного интеграла методом трапеции:

$\int_0^{\pi} (1 + \sin x) dx$, точное значение: $\frac{\pi}{2} + 1$. Определить, как изменяется число итераций при изменении точности.

Часть 2

Решить предыдущее задание с помощью Enumerable или Enumerator.

Часть 3

Составить метод trap для вычисления определенного интеграла по формуле трапеций

$\int_a^b f(x) dx \approx \frac{f(a) + f(b)}{2} + \sum_{i=1}^{n-1} f(x_i)$, где $f(x)$ подынтегральная функция, $[a, b]$ - интервал интегрирования, n - число отрезков разбиения. В основной программе использовать метод trap для вычисления интегралов:
 $\int_{-1}^4 (x + \cos x) dx$ и $\int_0^2 \frac{tg(x+1)}{x+1} dx$.

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, включающий поле — слово. Объект умеет выводить на экран значение своего поля и отвечать на запрос о его значении и количестве букв в слове.

Объект, включающий поля: целое число (длина слова) и слово. Объект умеет выводить на экран содержимое своих полей, возвращать по запросу их содержимое и количество гласных букв в слове.

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:

Два числа называются взаимно дружественными, если каждое из них равно сумме всех делителей другого, кроме самого этого числа. Например: 220 и 284, так как

Делители 220 \Rightarrow 1, 2, 4, 5, 10, 11, 20, 22, 44, 55, 110 сумма = 284

Делители 284 \Rightarrow 1, 2, 4, 71, 142 сумма = 220

Написать программу, определяющую дружественные числа, лежащие в диапазоне от 1 до n (n вводится с клавиатуры). Вывести на печать все найденные числа. При программировании использовать функции.