МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение

высшего образования

**«КУБАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»**

**(ФГБОУ ВО «КУБГУ)**

**Факультет компьютерных технологий и прикладной математики**

**Кафедра вычислительных технологий**

**ОТЧЕТ**

**о выполнении лабораторной работы №1**

**по дисциплине «Паттерны программирования»**

Работу выполнил\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_студент 36 группы, Грицаненко Н. А.

Работу проверил\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ преподаватель Климец А.А.

Краснодар 2021

**Содержание**

[Постановка задачи 3](#_Toc66493310)

[Задание 1 3](#_Toc66493311)

[Задание 2 3](#_Toc66493312)

[Задание 3 6](#_Toc66493313)

[Задание 4 7](#_Toc66493314)

[Задание 5 8](#_Toc66493315)

[Задание 6 9](#_Toc66493316)

[Задание 7 11](#_Toc66493317)

[Задание 8 13](#_Toc66493318)

[Задание 9 14](#_Toc66493319)

[Задание 10 17](#_Toc66493320)

[Заключение 18](#_Toc66493321)

# Постановка задачи

Потренироваться в написании классических алгоритмов на новом языке Ruby.

# Задание 1

Реализовать и вызвать Hello World c комментарием, рисунок 1:

****

Рисунок 1 – вывод на экран “Hello world”.

# Задание 2

Принять имя пользователя как аргумент программы. Поздороваться с пользователем. Рассмотреть тип строка. получить все методы, рассмотреть irb режим. Разобрать методы с использованием официальной документации языка.

На рисунке 2 изображен irb режим, в котором отображены все методы типа строка:

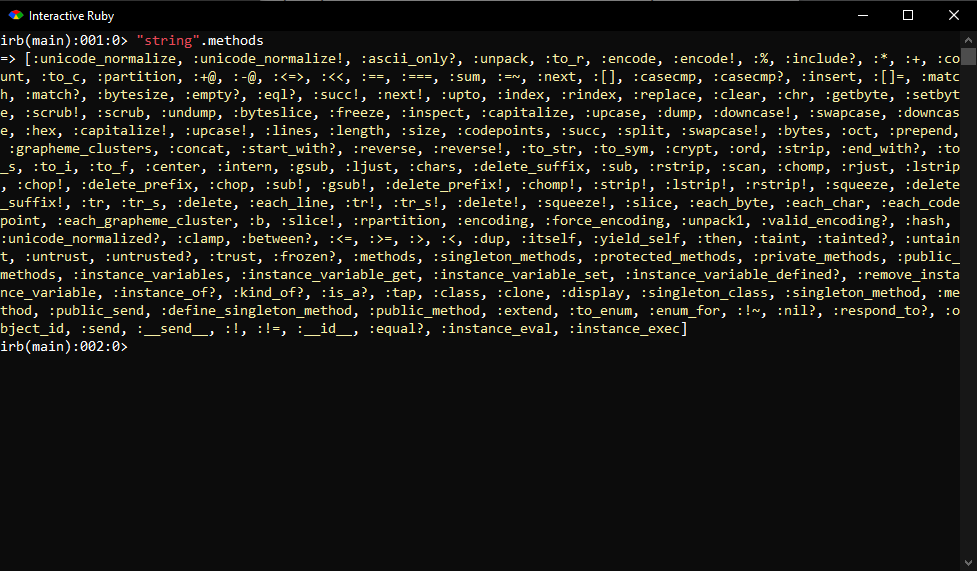


Рисунок 2 – irb режим.

Для ввода аргумента программы используется метод gets, как показано на рисунке 3:

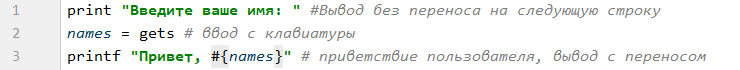


Рисунок 3 – Ввод аргумента программы.

Различные способы задания строки, рисунок 4:

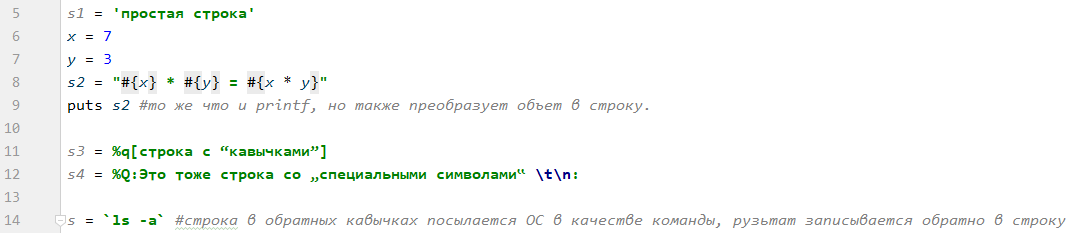


Рисунок 4 – Способы задания строки.

Далее на рисунках 5, 6, 7 изображены различные методы типа строка с поясняющими комментариями:

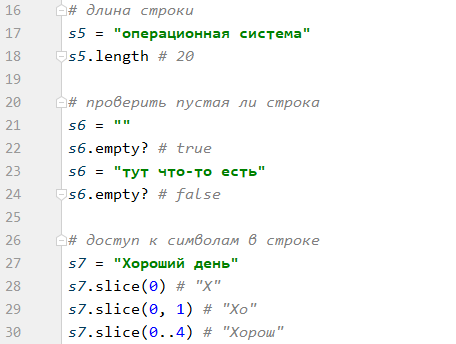


Рисунок 5 – Методы типа строка.

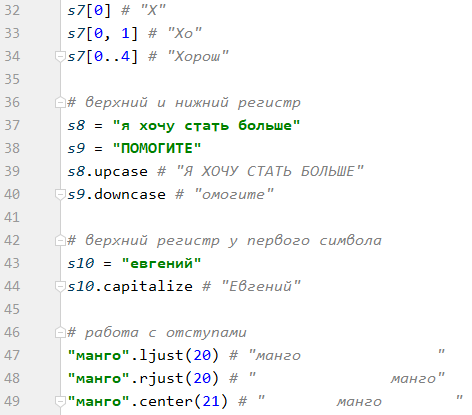


Рисунок 6 – Методы типа строка.

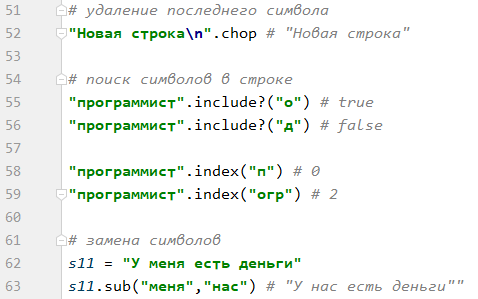


Рисунок 7 – Методы типа строка.

На рисунке 8 рассмотрены встроенные переменные и константы:

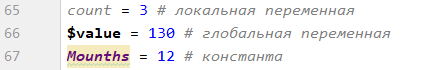


Рисунок 8 – Встроенные переменные и константы.

# Задание 3

Спросить какой язык у пользователя любимый, в случае, если это ruby, ответить что пользователь подлиза, иначе обязательно ответить, что скоро будет ruby и поставить различный комментарий для каждого языка. Рассмотреть условный оператор и разные способы его построения. Рассмотреть оператор switch case.

Взаимодействие пользователя с программой, рисунок 9:

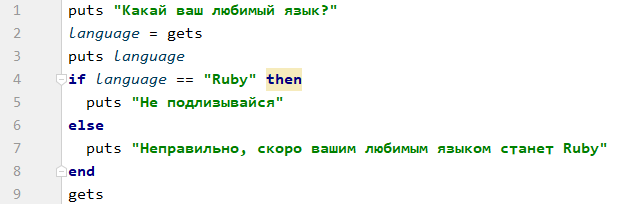


Рисунок 9 – Ввод значений пользователем с клавиатуры

На рисунках 10, 11, 12 изображены различные способы задания условного оператора (ex3.2, ex3.3, ex3.4 на github соответственно):

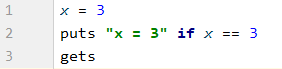


Рисунок 10 – Способ задания условного оператора

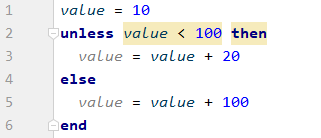


Рисунок 11 – Способ задания условного оператора

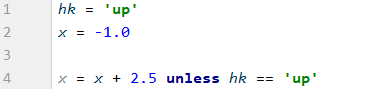


Рисунок 12 – Способ задания условного оператора

Оператор case в Ruby:

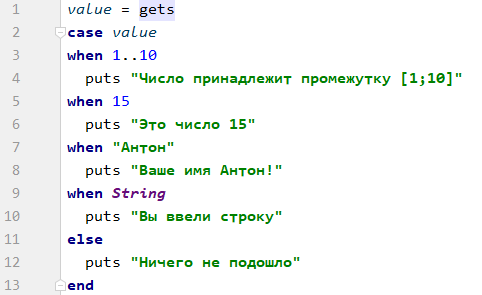


Рисунок 13 – оператор case.

# Задание 4

Попросить пользователя ввести команду языка ruby. И команду OC. Выполнить команду руби и команду операционной системы, рисунок 14:

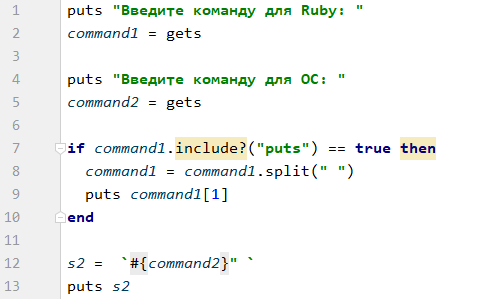
****

Рисунок 14 – Задание 4.

# Задание 5

Получить число как аргумент программы, найти сумму цифр числа, рисунок 15. Рассмотреть тип целое число. получить все методы в irb режиме. Разобрать методы с использованием официальной документации языка.

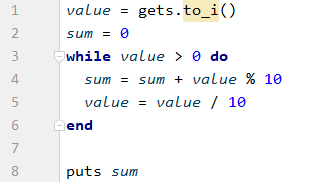
****

Рисунок 15 – Сумма цифр числа.

На рисунке 16 изображен irb режим со всеми методами типа цело число:

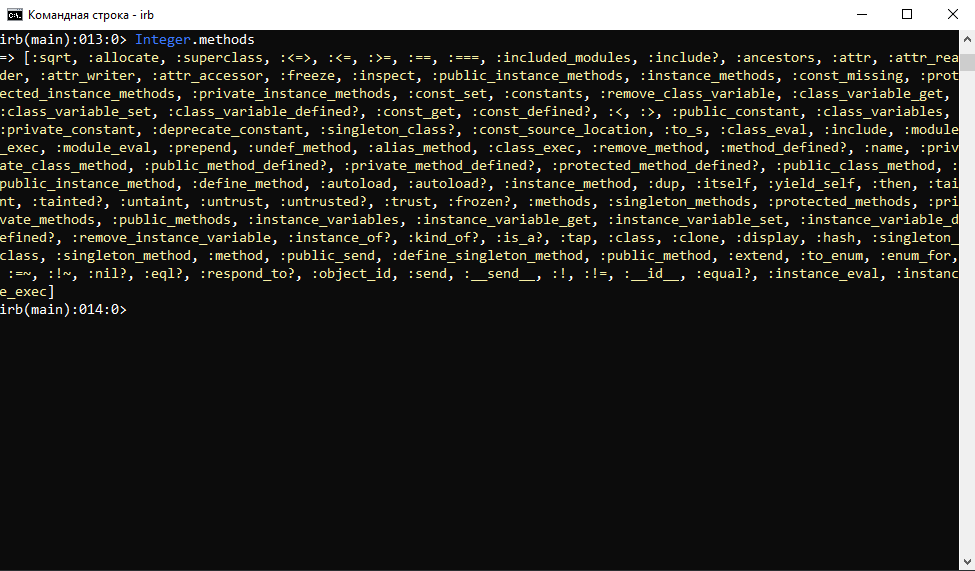


Рисунок 16 – Все методы типа целое число.

На рисунках 17 и 18 изображены разобранные методы типа целое число:

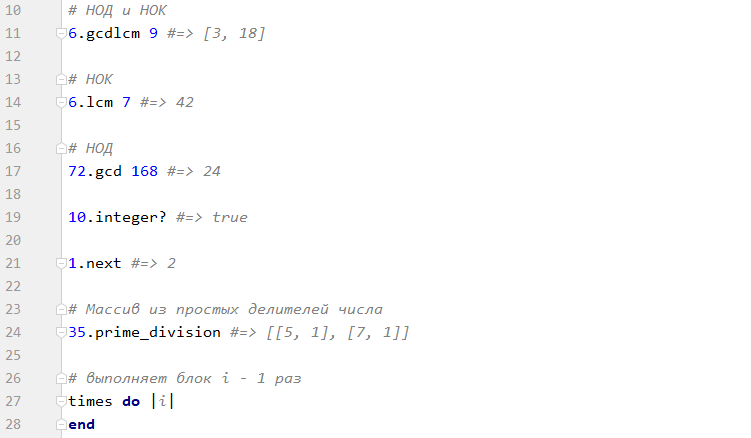


Рисунок 17 – Методы типа целое чило.

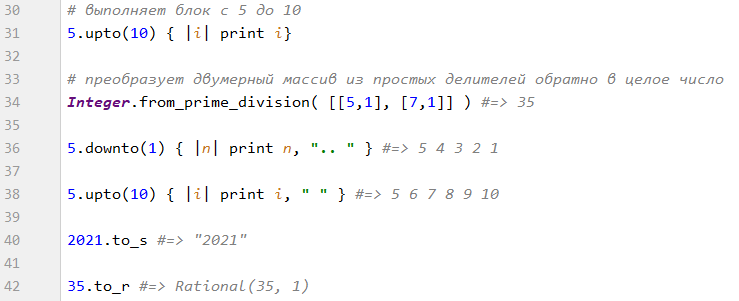


Рисунок 18 – Методы типа целое чило.

# Задание 6

Вынести нахождение суммы цифр числа в отдельный метод. Реализовать методы поиска максимальной и минимальной цифры числа, произведения цифр числа.

На рисунке 19 изображены методы нахождения суммы и произведения цифр цисла:

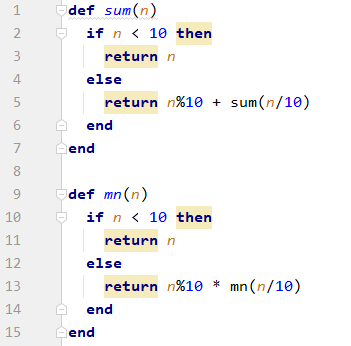
****

Рисунок 19 – Методы для нахождения суммы и произведения цифр числа.

На рисунке 20 изображены методы для поиска минимальной и максимальной цифры числа:

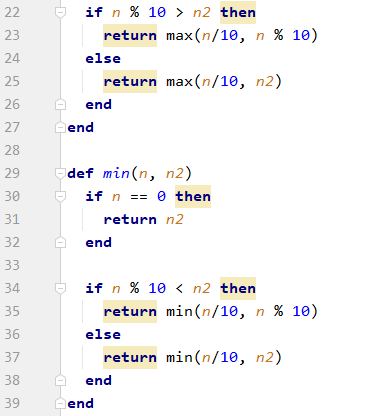


Рисунок 20 – Методы для поиска минимальной и максимальной цифры числа.

# Задание 7

Составить 3 метода для работы с цифрами или делителей числа на основании варианта. Ввести отдельный аргумент программы, в котором указывать, какой метод должен отработать из 7 составленных. Реализовать возможность работы от 1 до 7 методов в зависимости от переданных параметров. Если не передан ни один параметр, программа просто выводит Hello World.

Вариант № 3.

Метод 1. Найти максимальный простой делитель числа, рисунок 21.

Метод 2. Найти произведение цифр числа, не делящихся на 5, рисунок 22.

Метод 3. Найти НОД максимального нечетного непростого делителя числа и прозведения цифр данного числа, рисунок 23.

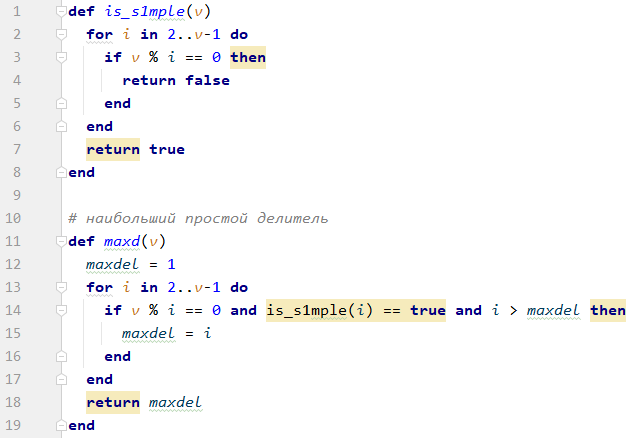


Рисунок 21 – Метод 1.

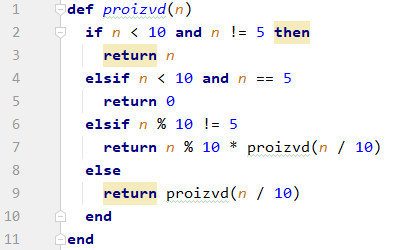


Рисунок 22 – Метод 2.

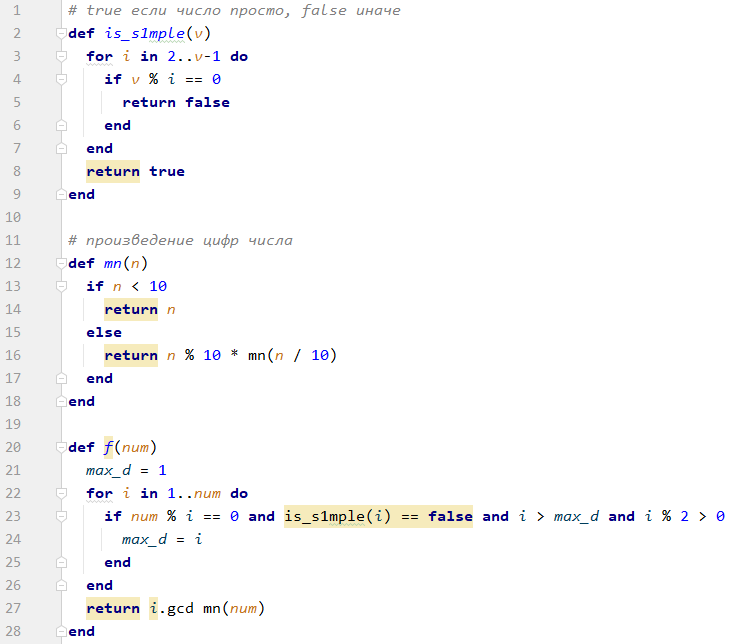


Рисунок 23 – Метод 3.

Реализация считывания параметра с клавиатуры и вызов определенного метода изображена на рисунке 24:

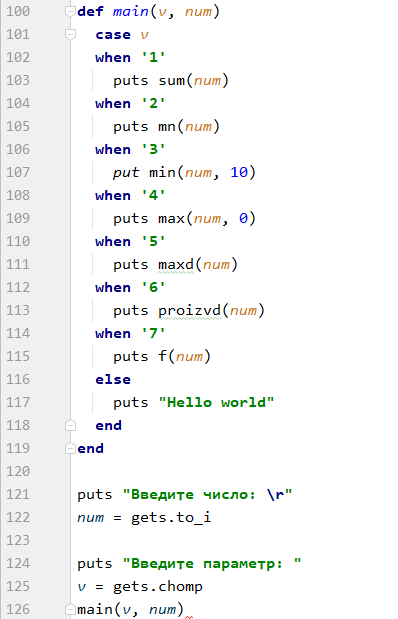


Рисунок 24 – Вызов метода в зависимости от переданного параметра.

# Задание 8

Ввести дополнительные аргументы программы, в которых указывать, какой метод должен отработать из 7 составленных. Реализовать возможность работы от 1 до 7 методов в зависимости от переданных параметров. Если не передан ни один параметр, или параметры переданы не верно реализовать параметр -h, описывающий все возможные параметры для Вашей программы, рисунок 25:

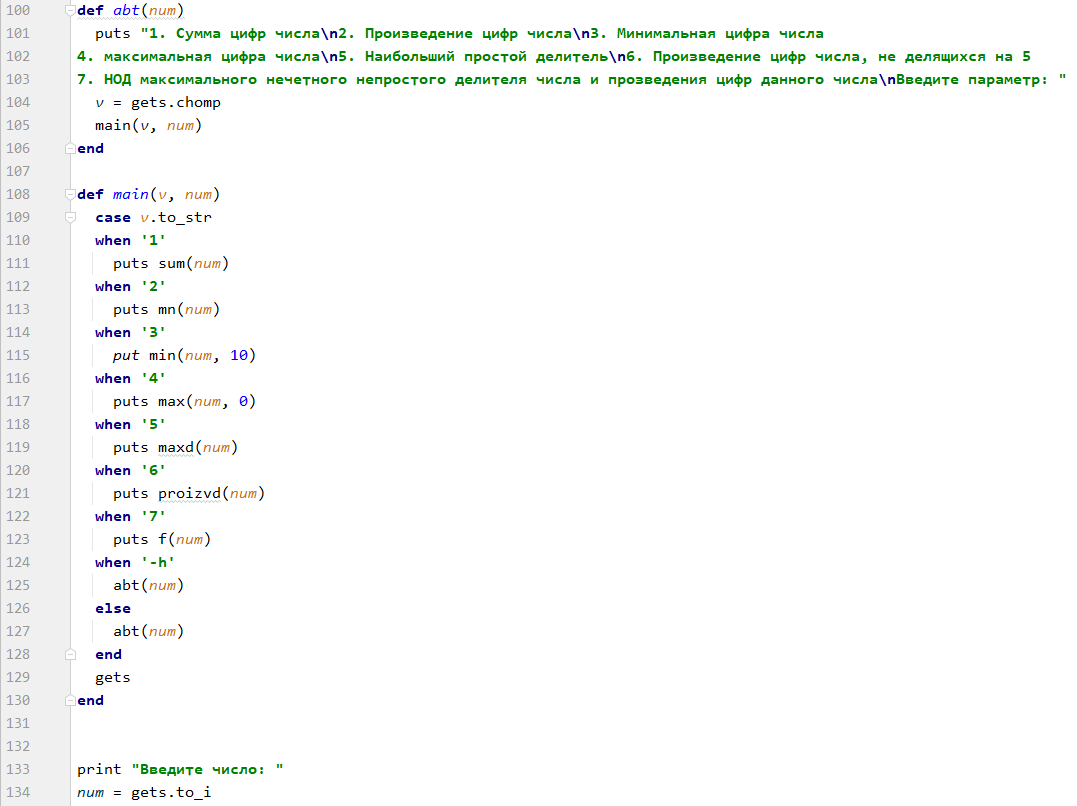
****

Рисунок 25 – Задание 8.

# Задание 9

На основании математических возможностей языка Ruby переписать методы предыдущего задания с использованием методов и классов соответствующих модулей.

На рисунках 26, 27, 28, 29, 30, 31 и 32 изображены методы для нахождения суммы цифр числа, произведение цифр числа, максимальной цифры числа, минимальной цифрой числа, НОД, произведение цифр числа, не делящихся на 5 и НОД максимального нечетного простого делителя числа и произведения цифр данного числа соответственно, с применением методов.

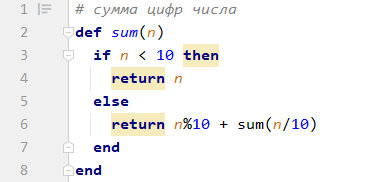


Рисунок 26 – Сумма цифр числа.

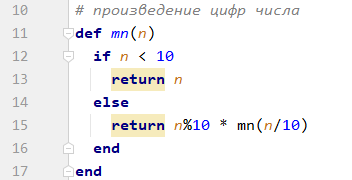


Рисунок 27 – Произведение цифр числа.

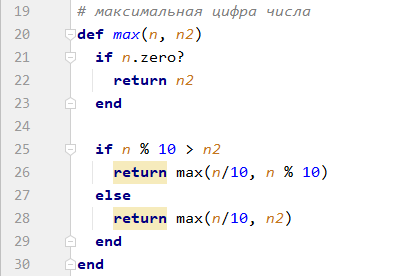


Рисунок 28 – Наибольшая цифра числа.

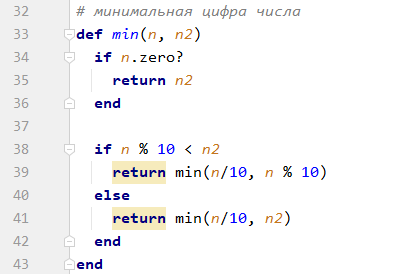


Рисунок 29 – Минимальная цифра числа.

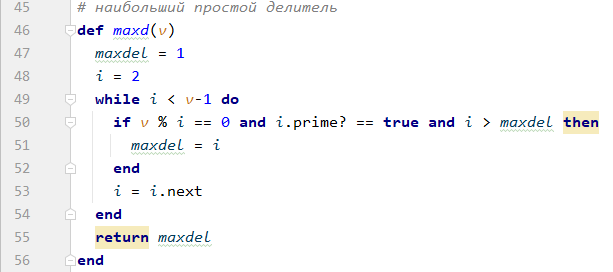


Рисунок 30 – НОД.

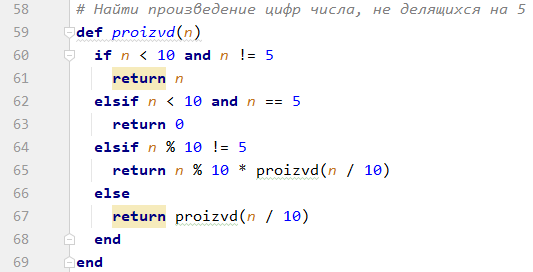


Рисунок 31 – Произведение цифр числа, не делящихся на 5.

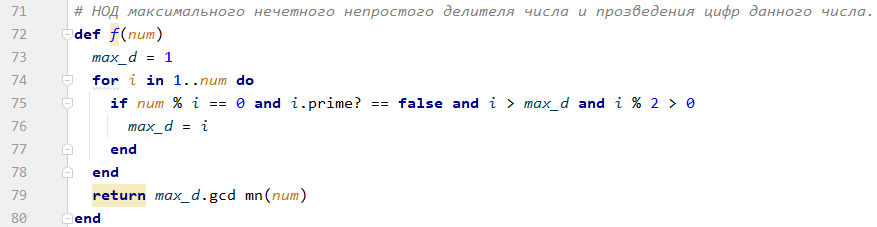


Рисунок 32 - НОД максимального нечетного непростого делителя числа и прозведения цифр данного числа.

Использованные методы: .zero?, .prime?, .next, .gcd

# Задание 10

С учетом итогов предыдущего задания выполнить задачи с ресурса https://projecteuler.net/archives под номерами 10+n, 30+n, где n номер варианта, n = 3

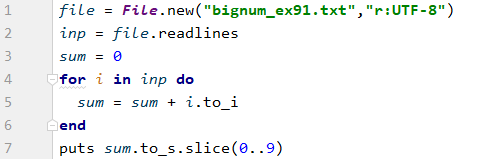
****

Рисунок 33 – 13 задача.

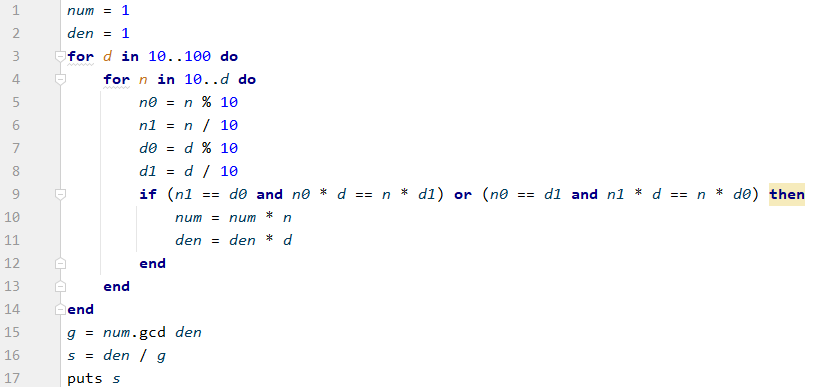


Рисунок 34 – 33 задача.

# Заключение

В течение лабораторной работы были рассмотрены основы языка программирования Ruby, работа со строками, числовыми типами. Были рассмотрены различные реализации условных операторов, а также разные реализации одинаковых методов.