

Контрольна робота №1

Виконав студент 3 курсу

Бондаренко Олег Анатолійович КС31

Варіант 13

Теоретичні питання

1. Як видалити елемент з масиву у Ruby?
2. Що таке масиви масивів (двовимірні масиви) у Ruby? Як їх використовувати?
3. Як реалізується множинне спадкування за допомогою модулів у Ruby?
4. Що таке інстансні методи та змінні? Як вони використовуються у Ruby?

Практичні завдання

1. Напишіть метод, який знаходить найбільший спільний дільник (НСД) двох чисел.
2. Створіть клас Student, що містить ім'я студента і список його оцінок. Додайте метод для обчислення середнього балу.

Виконання

Теоретичні запитання

1. На мові програмування рубі існує декілька способів видалити елементи масиву:
 - Метод Delete - видаляє всі елементи масиву, які дорівнюють заданому значенню.
 - Метод delete_at - видаляє елемент масиву за його індексом.
 - Метод reject - повертає новий масив без елементів, які задовольняють умову.
2. Масив масивів — це масив елементами якого є інші масиви. Вони потрібні для того щоб відображати данні у вигляді матриць або таблиць.

3. Ruby не підтримує множинне спадкування безпосередньо, але модулі надають можливість використовувати поведінку схожу на множинне спадкування через механізм міксинів.

Модуль - це набір методів, який можна "включити" у клас за допомогою ключового слова `include` або `extend`. Це дозволяє класу використовувати методи з декількох модулів одночасно, імітуючи множинне спадкування.

4. Інстансні методи - це методи, які належать до конкретного об'єкта. Вони можуть отримувати доступ до інстансних змінних та взаємодіяти з ними.

Інстансні змінні - це змінні, які є властивостями певного об'єкта. Вони зберігають стан цього об'єкта та доступні лише в межах цього екземпляра. Їх імена починаються з символу «@».

Практичні питання

1. Лістинг завдання 1:

```
def NSD(a,b)
  if b == 0
    return a
  end
  return NSD(b,a % b)
end
```

```
puts (NSD(30,15))
```

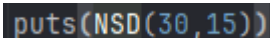


Рисунок 1 вхідні данні



Рисунок 2 вихідні данні

2. Лістинг класу Student:

```
class Student
  def initialize(name, bal)
    @name = name
    @bal = bal
  end

  def average
    total = 0
    count = 0
```

```

    @bal.each do |num|
      total += num
      count += 1
    end

    average_score = total.to_f / count
    puts "Середній бал: #{average_score}"
  end
end

s1 = Student.new("Oleg", [5, 4, 5, 3, 5, 3, 4, 2])
s1.average

```

```

s1 = Student.new( name "Oleg", bal [5, 4, 5, 3, 5, 3, 4, 2])

```

Рисунок 3 вхідні данні

```

Студент: Oleg
Середній бал: 3.875

```

Рисунок 4 вихідні данні