Міністерство освіти та науки України

Національний технічний університет України

«Київський Політехнічний Інститут ім. Ігоря Сікорського»

ФТІ

Кафедра ФТЗЗІ

**Лабораторна робота №5**

з дисципліни: «Програмування 4»

на тему:

**«Використання функцій у мові Python»**

**Варіант 11**

Виконав:

Ст. гр. ФЕ-81

Корнеєнков Кирило

Перевірив:

доцент Прогонов Д.О.

Київ 2020

Мета роботи: Оволодіння практичними навичками у використанні функцій .

**1. Порядок виконання роботи**

1. Проаналізувати умову задачі.

2. Розробити алгоритм та створити програму розв’язання задачі згідно з номером варіанту.

3. Результати роботи оформити протоколом.

Завдання**(5А)**:

Визначити функцію повного скорочення раціонального числа, яке задано чисельником та замінником, а також процедури складання та множення раціональних чисел.

Код реалізації **(5A)**:

class Rational():

def \_\_init\_\_(self, numerator=0, demoniator=1):

self.numer = numerator

self.denom = demoniator

def \_\_add\_\_(self,other):

n1 = self.numer

d1 = self.denom

n2 = other.numer

d2 = other.denom

n3 = d2\*n1 + d1\*n2

d3 = d1\*d2

return Rational(n3,d3)

def \_\_mul\_\_(self, other):

n1 = self.numer

d1 = self.denom

n2 = other.numer

d2 = other.denom

n3 = n1\*n2

d3 = d1\*d2

return Rational(n3,d3)

def gcd(self, a, b):

if b == 0:

return a

return self.gcd(b, a % b)

def reduce(self):

d = 1

if self.denom != 0 and self.numer != 0:

d = self.gcd(self.numer, self.denom)

if d > 1:

self.numer /= d

self.denom /= d

a = Rational(int(self.numer),int(self.denom))

return a

def \_\_str\_\_(self):

return "{0}/{1}".format(self.numer,self.denom)

# Создаём два объекта рациональных чисел:

r1 = Rational(4, 2)

r2 = Rational(1, 2)

# Проверяем перегруженную операцию сложения:

r3 = r1 + r2

# Проверяем перегруженный метод вывода:

print(r3)

# Проверяем перегруженную операцию уножения:

r5 = r1 \* r2

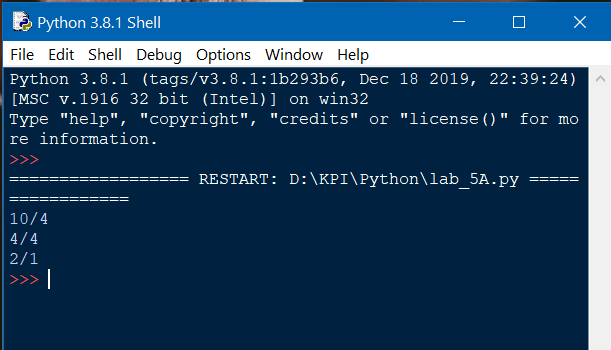
print(r5)

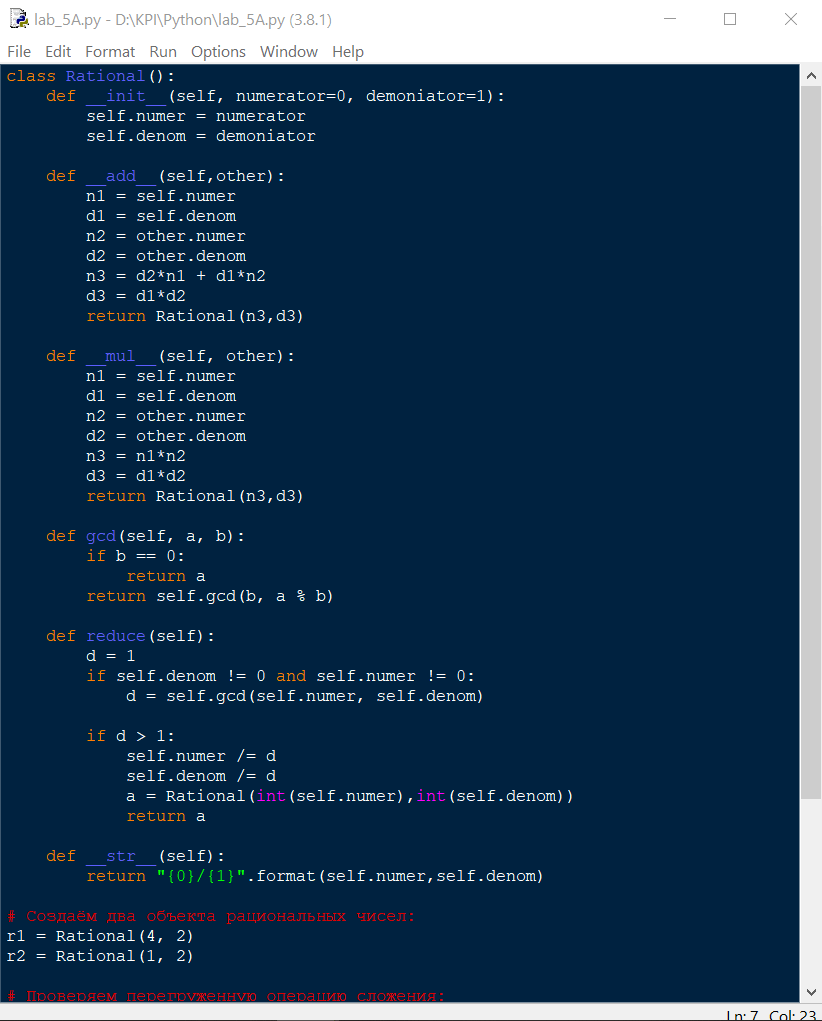
# Проверяем метод сокращения числа:

r3 = r1.reduce()

print(r3)

Приклад виконання програми**(5A)**:





Завдання**(5B)**:

Знайти суму n членів геометричної прогресії із заданим початковим членом та кроком.

Код реалізації **(5B)**:

def sum\_geom\_progr(N, first, znam):

return (first\*(1-pow(znam,N)))/(1-znam)

try:

n = abs(int(input("Количество элементов в прогрессии: ")))

if n != 0:

first = float(input("Первый элемент прогрессии: "))

if n == 1:

print("В Вашей прогрессии 1 элемент, сумма -", first)

else:

step = float(input("Знаменатель прогрессии: "))

print("Сумма", n , "членов Вашей геометрической прогрессии -", round(sum\_geom\_progr(n, first, step),3))

else:

print("В Вашей прогрессии 0 элементов, сумма - 0")

except ValueError:

print("Ошибка! Проверьте данные и попробуйте снова!")

Приклад виконання програми**(5B)**:

