**Федеральное государственное образовательное бюджетное учреждение  
высшего образования  
«Финансовый университет при Правительстве РФ»**

**Отчет по практикуму по программированию. Вариант 9.**

Выполнил студент

Зайцев Никита Валерьевич

Группы ПИ20-2

Проверил доцент

Бочаров Михаил Иванович

Москва 2020

**Условие задачи:**

**Базовая часть:**

Написать калькулятор для строковых выражений вида '<число> <операция> <число>', где <число> - не отрицательное целое число меньшее 100, записанное словами, например "тридцать четыре", <арифмитическая операция> - одна из операций "плюс", "минус", "умножить". Результат выполнения операции вернуть в виде текстового представления числа. Пример calc("двадцать пять плюс тринадцать") -> "тридцать восемь"

Оформить калькулятор в виде функции, которая принимает на вход строку и возвращает строку.

**Дополнительное задание:**

Добавить комбинаторные операции перестановки, размещения и сочетания. Пример: calc("размещений из трех по два") -> "шесть".

**Листинг:**

*import* math  
  
operation\_list = [  
 'плюс',  
 'минус',  
 'умножить',  
 'поделить',  
 'размещение',  
 'сочетание',  
 'перестановка',  
]  
numbers = {  
 'один': 1,  
 'два': 2,  
 'три': 3,  
 'четыре': 4,  
 'пять': 5,  
 'шесть': 6,  
 'семь': 7,  
 'восемь': 8,  
 'девять': 9,  
 'десять': 10,  
 'одиннадцать': 11,  
 'двенадцать': 12,  
 'тринадцать': 13,  
 'четырнадцать': 14,  
 'пятнадцать': 15,  
 'шестнадцать': 16,  
 'семнадцать': 17,  
 'восемнадцать': 18,  
 'девятнаддцать': 19,  
 'двадцать': 20,  
 'тридцать': 30,  
 'сорок': 40,  
 'пятьдесят': 50,  
 'шестьдесят': 60,  
 'семьдесят': 70,  
 'восемьдесят': 80,  
 'девяносто': 90,  
 'сто': 100,  
}  
number1, number2 = 0, 0  
  
  
*def* get\_numbers(pos, str\_input):  
 *global* number1, number2  
 *for* num *in* numbers:  
 *for* j *in range*(*len*(str\_input)):  
 *if* num == str\_input[j]:  
 *if* j < pos:  
 number1 += numbers[num]  
 *elif* j > pos:  
 number2 += numbers[num]  
  
  
*def* calc():  
 *global* number1, number2  
 pos = 0  
 operation = 0  
 str\_input = *input*().split(' ')  
 *for* i *in range*(*len*(operation\_list)):  
 *for* j *in range*(*len*(str\_input)):  
 *if* operation\_list[i] == str\_input[j]:  
 pos = j  
 operation = i  
 *if* operation < 4:  
 get\_numbers(pos, str\_input)  
  
 *else*:  
 *if* operation < 6:  
 *for* j *in range*(*len*(str\_input)):  
 *if* 'по' == str\_input[j]:  
 pos = j  
 get\_numbers(pos, str\_input)  
 *else*:  
 *for* num *in* numbers:  
 *for* j *in range*(*len*(str\_input)):  
 *if* num == str\_input[j]:  
 number1 += numbers[num]  
  
 *if* operation == 0:  
 *print*('Ответ:', number1 + number2)  
 *elif* operation == 1:  
 *print*('Ответ:', number1 - number2)  
 *elif* operation == 2:  
 *print*('Ответ:', number1 \* number2)  
 *elif* operation == 3:  
 *print*('Ответ:', number1 / number2)  
 *elif* operation == 4:  
 *print*('Ответ:', math.factorial(number1) / math.factorial(number1 - number2))  
 *elif* operation == 5:  
 *print*('Ответ:', math.factorial(number1) / (math.factorial(number2) \* math.factorial(number1 - number2)))  
 *elif* operation == 6:  
 *print*('Ответ:', math.factorial(number1))  
  
  
*if* \_\_name\_\_ == '\_\_main\_\_':  
 calc()

Демонстрация работы программы. На рисунках 1-4 представлен пример работы программы.

Изображение выглядит как текст

Автоматически созданное описание

Рис. 1 – Результат работы программы

Изображение выглядит как текст

Автоматически созданное описание

Рис. 2 – Результат работы программы

Изображение выглядит как текст

Автоматически созданное описание

Рис. 3 – Результат работы программы

Изображение выглядит как текст

Автоматически созданное описание

Рис. 4 – Результат работы программы

Интернет источники

1. <https://docs.python.org/3/>