Отчет по лабораторной работе №8

Дисциплина: Математические основы защиты информации и информационной безопасности

Живцова Анна

Содержание

# 1 Цель работы

Изучить и реализовать алгоритмы арифметических операций для больших чисел, записанных в -нарной системе счисления.

# 2 Задание

Реализовать сложение, вычитание, умножение и деление с отсатком для чисел, записанных в виде последовательности символов в -нарной системе счисления.

# 3 Теоретическое введение

В криптографии часто возникает необходимость работы с большими числами. Для эффективного проведения арифметических операций данные числа удобно хранить в виде последовательности символов в -нарной системе счисления. Подробнее в источниках [1,2].

# 4 Выполнение лабораторной работы

Тут и – числа между которыми требуется выполнить бинарную операцию, а – основание системы счисления. В переменной хранится результат выполнения операции.

## 4.1 Сложение и вычитание

Для реализации сложения и вычитания больших чисел на языке Python были написанны следующие функции.

def add(u, v, b):  
 w = []  
 k = 0  
 for i in range(len(u)-1, -1, -1):  
 w.append((u[i] + v[i] + k)%b)  
 k = (u[i] + v[i] + k)//b  
 w.append(k)  
 return w[::-1]  
  
def substract(u, v, b):  
 w = []  
 k = 0  
 for i in range(len(u)-1, -1, -1):  
 w.append((u[i] - v[i] + k)%b)  
 k = (u[i] - v[i] + k)//b  
 return w[::-1]

## 4.2 Умножение

Было реализовано два вида умножения

def mult(u, v, b):  
 w = [0]\*(len(u) + len(v) + 1)  
 u = u[::-1]  
 v = v[::-1]  
 for i in range(len(u)):  
 for j in range(len(v)):  
 w[i + j] += u[i]\*v[j]  
 for i in range(len(u)+len(v)):  
 w[i + 1] += w[i]//b  
 w[i] %= b  
 return w[::-1]  
  
  
def fast\_mult(u, v, b):  
 w = [0]\*(len(u) + len(v))  
 t = 0  
 for s in range(len(u) + len(v)):  
 for i in range(s+1):  
 t += u[len(u) - i - 1]\*v[len(v)-s+i - 1]  
 w[len(u) + len(v) - s - 1] = t%b  
 t = t//b  
 return w

## 4.3 Деление с остатком

Реализовано с помощью функции

def div(u, v, b):  
 q = [0]\*(len(u) - len(v) + 1)  
 uu = sum([u[i]\*b\*\*(len(u) - i - 1) for i in range(len(u))])  
 vv = sum([v[i]\*b\*\*(len(v) - i - 1) for i in range(len(v))])  
 while uu >= vv\*(b\*\*(len(u) - len(v))):  
 q[len(u) - len(v)] += 1  
 u = substract(u, v + [0]\*(len(u) - len(v)), b)  
 uu -= vv\*(b\*\*(len(u) - len(v)))   
 for i in range(len(u), len(v), -1):  
 if u[i] >= v[len(v)]:  
 q[i - len(v) - 1] = b - 1  
 else:  
 q[i - len(v) - 1] = (u[i]\*b + u[i-1])//v[len(v)]  
 while q[i - len(v) - 1]\*(u[len(v)]\*b + u[len(v)-1]) > u[i]\*b\*b + u[i-1]\*b + u[i-2]:  
 q[i - len(v) - 1] -= 1  
 arr = mult(v+[0]\*(i - len(v) - 1), q[i - len(v) - 1], b)  
 u = substract(u, arr, b)  
 uu -= q[i - len(v) - 1]\*b\*\*(i - len(v) - 1)\*vv  
 if uu < 0:  
 uu += vv\*b\*\*(i - len(v) - 1)  
 u = add(u, v+[0]\*(i - len(v) - 1))  
 q[i - len(v) - 1] -= 1  
 return [i%b for i in q], u

## 4.4 Тестирование функций

Реализованные функции были протестированы на примере трехзначных десятичных чисел (см. рис. **¿fig:005?**). Все результаты оказались верными.

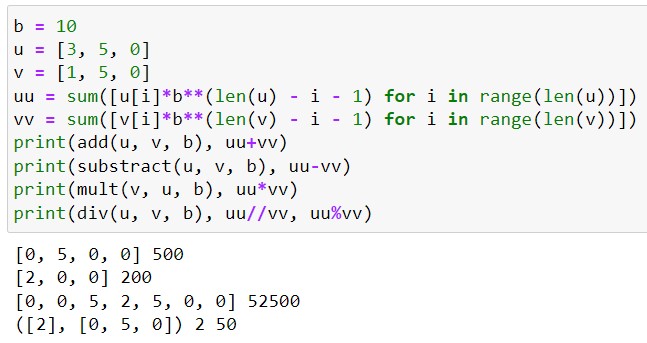


Рис. 1: Тестирование арифметических операций с трехзначными десятичными числами

# 5 Выводы

В данной работе я изучила и реализовала алгоритмы арифметических операций для больших чисел, записанных в -нарной системе счисления. Реализованные мной сложение, вычитание, умножение и деление с остатком были протестированы на трехзначных десятичных числах.

# Список литературы

1. Kulyabov D., Korolkova A., Gevorkyan M. Информационная безопасность компьютерных сетей: лабораторные работы. 2015.

2. Самуйлов К.Е. и др. Сети и телекоммуникации : Учебник и практикум. Издательство Юрайт, 2019. С. 1–363.