Лабораторная работа №8

Целочисленная арифметика многократной точности

Доборщук Владимир Владимирович, НФИмд-02-22

Содержание

# 1 Цель и задачи работы

**Цель** — Изучить алгоритмы целочисленной арифметики многократной точности.

**Задачи:**

* Реализовать представленные алгоритмы

# 2 Теоретическая информация

Все теоретическое описание дано в описании лабораторной работы.

# 3 Выполнение лабораторной работы

При выполнении лабораторной работы мы строго следовали алгоритмике, представленной в описании.

## 3.1 Реалиазация и тестирование

Программный код выглядит следующим образом:

# Laboratory Work  
# Theme: Arithmetic for big numbers  
# Author: Vladimir Doborschuk  
  
# --- Functions ---  
  
def mod(a ,b):  
 return a % b  
  
def big\_sum(u, v, b):  
 u\_ = str(u)  
 v\_ = str(v)  
   
 j = len(u\_) - 1  
   
 if j != len(v\_) - 1:  
 print("bad N")  
 return None  
   
 k = 0  
   
 w = ""  
   
 while j >= 0:  
 w\_ = mod(int(u\_[j]) + int(v\_[j]) + k, b)  
 w += str(w\_)  
 k = (int(u\_[j]) + int(v\_[j]) + k) // b  
 j = j - 1  
   
 w += str(k)  
 return int(w[::-1])  
  
def big\_differ(u, v, b):  
 u\_ = str(u)  
 v\_ = str(v)  
   
 j = len(u\_) - 1  
   
 if j != len(v\_) - 1:  
 print("bad N")  
 return None  
   
 k = 0  
   
 w = ""  
   
 while j >= 0:  
 w\_ = mod(int(u\_[j]) - int(v\_[j]) + k, b)  
 w += str(w\_)  
 k = (int(u\_[j]) - int(v\_[j]) + k) // b  
 j = j - 1  
  
 return int(w[::-1])  
  
def big\_multiple(u, v, b):  
 u\_ = str(u)  
 v\_ = str(v)  
   
 j = len(v\_) - 1  
 w = [0] \* (j \* len(u\_))  
   
 while j >= 0:  
 if v\_[j] == 0:  
 w[j] = 0  
 j = j - 1  
 else:  
 i = len(u\_) - 1  
 k = 0  
 while i >= 0:  
 t = int(u\_[i]) \* int(v\_[j]) + w[i+j] + k  
 w[i+j] = mod(t, b)  
 k = t // b  
 i = i - 1  
 w[j] = k  
 j = j - 1  
  
 return int("".join(list(map(str, w))))  
  
# --- Main ---  
  
def main():  
 x = 874  
 y = 775  
   
 print(f"Sum: {x} + {y} (10)")  
 print(big\_sum(x, y, 10))  
   
 print(f"Differ: {x} - {y} (10)")  
 print(big\_differ(x, y, 10))  
   
 print(f"Multiplication: {x} \* {y} (10)")  
 print(big\_multiple(x, y, 10))  
   
if \_\_name\_\_ == "\_\_main\_\_":  
 main()

При запуске получаем следующие результаты:

Sum: 874 + 775 (10)  
1649  
Differ: 874 - 775 (10)  
99  
Multiplication: 874 \* 775 (10)  
684500

# 4 Выводы

В рамках выполненной лабораторной работы мы изучили и реализовали представленные алгоритмы целочисленной арифметики многократной точности.

# Список литературы