

Лабораторная работа №8

Целочисленная арифметика многократной точности

Доборщук Владимир Владимирович, НФИмд-02-22

Содержание

1	Цель и задачи работы	5
2	Теоретическая информация	6
3	Выполнение лабораторной работы	7
3.1	Реализация и тестирование	7
4	Выводы	11
	Список литературы	12

Список иллюстраций

Список таблиц

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 Реализация 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_multiply(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_multiply(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