

Федеральное государственное автономное образовательное учебное
учреждение высшего образования “Национальный исследовательский
университет ИТМО”

Факультет информационных технологий и программирования

Домашняя работа №1

“Выполнение арифметических операций с двоичными числами”

Вариант №11

Выполнил:

Ситдигов Рафаэль Ильдусович

Группа: М3115

Проверил:

Г. Санкт-Петербург, 2025г.

Оглавление:

1. Задание	3
2. Этапы вычислений	4
3. Использование программного решения	5
4. Заключение	7

Задание:

Овладеть простейшими навыками перевода чисел в различные системы счисления и выявить ошибки, возникающие из-за их ограниченной разрядности.

$$X_1 = A$$

$$X_2 = C$$

$$X_3 = A + C$$

$$X_4 = A + C + C$$

$$X_5 = C - A$$

$$X_6 = 65536 - X_5$$

$$X_7 = -X_1$$

$$X_8 = -X_2$$

$$X_9 = -X_3$$

$$X_{10} = -X_4$$

$$X_{11} = -X_5$$

$$X_{12} = -X_6$$

$$B_1 = A$$

$$B_2 = C$$

$$B_3 = A + C$$

$$B_4 = A + C + C$$

$$B_5 = C - A$$

$$B_6 = 65536 - (C - A)$$

$$B_7 = -A$$

$$B_8 = -C$$

$$B_9 = -(A + C)$$

$$B_{10} = -(A + C + C)$$

$$B_{11} = -(C - A)$$

$$B_{12} = -(65536 - (C - A))$$

$$B_1 + B_2 = A + C$$

$$B_2 + B_3 = A + C + C$$

$$B_7 + B_8 = -(A + C)$$

$$B_8 + B_9 = -(A + C + C)$$

$$B_2 + B_7 = -A + C$$

$$B_1 + B_8 = A - C$$

Этапы вычисления:

$$X_1 = A = 101011010110$$

$$B_1 + B_2 = A + C = 46f2$$

$$X_2 = C = 11110000011100$$

$$B_2 + B_3 = A + C + C = 830e$$

$$X_3 = A + C = 100011011110010$$

$$B_7 + B_8 = -(A + C) = -46f2$$

$$X_4 = A + C + C = 1000001100001110$$

$$B_8 + B_9 = -(A + C + C) = -830e$$

$$X_5 = C - A = 11000101000110$$

$$B_2 + B_7 = -A + C = 3146$$

$$X_6 = 65536 - X_5 = 1100111010111010$$

$$B_1 + B_8 = A - C = -3146$$

$$X_7 = -X_1 = 1.010100101010$$

$$X_8 = -X_2 = 1.00001111100100$$

$$X_9 = -X_3 = 1.011100100001110$$

$$X_{10} = -X_4 = 1.0111110011110010$$

$$X_{11} = -X_5 = 1.00111010111010$$

$$X_{12} = -X_6 = 1.0011000101000110$$

*1.00...1 – Это показатель отрицательного двоичного числа в ЭВМ

Программное решение:

```
def inver(x):
    a = ''
    for i in x:
        if i == '1':
            a += '0'
        elif i == '0':
            a += '1'
    a = '1' + a
    return a

def invers2(x):
    if x > 0:
        return hex(x)[2:]
    elif x < 0:
        a = hex(x)
        a = a.replace('0x', '')
        return a

def binary(lab):
    if len(lab) != 2:
        print('Недопустимое количество вводимых чисел')
    else:
        A = lab[0]
        C = lab[1]

        task1 = bin(A)[2:]
        task2 = bin(C)[2:]
        task3 = bin(A + C)[2:]
        task4 = bin(A + C + C)[2:]
        task5 = bin(C - A)[2:]
        task6 = bin(65536 - (C - A))[2:]
        task7 = inver(task1)
        task8 = inver(task2)
        task9 = inver(task3)
        task10 = inver(task4)
        task11 = inver(task5)
        task12 = inver(task6)
        taskB1 = invers2((A+C))
        taskB2 = invers2((A + C + C))
        taskB3 = invers2((-A - C))
        taskB4 = invers2((-A - C - C))
        taskB5 = invers2((C - A))
        taskB6 = invers2((A - C))

        task = [task1, task2, task3,
                task4, task5, task6,
                task7, task8, task9,
                task10, task11, task12,
                taskB1, taskB2, taskB3,
                taskB4, taskB5, taskB6]

        for i in task:
            print(i)

read = (open(f'Test1.txt'))
test = read.read().split(' ') #Открываем и читаем файл.
lab = [int(x) for x in test] #Заполняем список из test
binary(lab)
```

Вывод программы:

101011010110

11110000011100

100011011110010

1000001100001110

11000101000110

1100111010111010

1010100101001

100001111100011

1011100100001101

10111110011110001

100111010111001

10011000101000101

46f2

830e

-46f2

-830e

3146

-3146

Заключение:

В данной работе я изучил работу с переводом чисел в двоичную и шестнадцатеричную системы счисления, а также разрабатывать скрипт для перевода в системах счисления.