

Факультет Программной Инженерии и Компьютерной техники

“Перевод чисел между различными системами счисления”

Лабораторная работа №1

Вариант 19

Выполнил: Малых Кирилл Романович

Группа: P3132

Преподаватель: Бострикова Дарья Константиновна

Санкт-Петербург, 2025г.

Оглавление

[**1. Порядок выполнения работы** 2](#_Toc208586588)

[**2. Решение задач** 3](#_Toc208586589)

[*Задание №1* 3](#_Toc208586590)

[Рисунок 2.1: Последовательное деление на 13. 4](#_Toc208586591)

[*Задание №2* 4](#_Toc208586592)

[*Задание №3* 5](#_Toc208586593)

[Рисунок 2.2: Последовательное деление на 13 5](#_Toc208586594)

[*Задание №4* 5](#_Toc208586595)

[Рисунок 2.3: последовательное деление на 2. 6](#_Toc208586596)

[Рисунок 2.4: перевод в 2-ичную СС. 7](#_Toc208586597)

[*Задание №5* 7](#_Toc208586598)

[*Задание №6* 8](#_Toc208586599)

[*Задание №7* 8](#_Toc208586600)

[*Задание №8* 9](#_Toc208586601)

[*Задание №9* 9](#_Toc208586602)

[*Задание №10* 10](#_Toc208586603)

[*Задание №11* 10](#_Toc208586604)

[*Задание №12* 11](#_Toc208586605)

[*Задание №13* 11](#_Toc208586606)

[*Дополнительное задание №1* 12](#_Toc208586607)

[**3. Список Литературы** 13](#_Toc208586608)

# **1. Порядок выполнения работы**

1. Перевести число "А", заданное в системе счисления "В", в систему счисления "С". Числа "А", "В" и "С" взять из представленных ниже таблиц. Вариант выбирается как сумма последних двух цифр в номере группы и номера в списке группы согласно ISU. Т.е. 13-му человеку из группы P3102 соответствует 15-й вариант (=02 + 13). Если полученный вариант больше 40, то необходимо вычесть из него 40. Т.е. 21-му человеку из группы P3121 соответствует 2-й вариант (=21 + 21 - 40).

2. Обязательное задание (позволяет набрать до 85 процентов от максимального числа баллов БаРС за данную лабораторную). Всего нужно решить 13 примеров. Для примеров с 5-го по 7-й выполнить операцию перевода по сокращенному правилу (для систем с основанием 2 в системы с основанием 2^k). Для примеров с 4-го по 6-й и с 8-го по 9-й найти ответ с точностью до 5 знака после запятой. В примере 11 группа символов {^1} означает -1 в симметричной системе счисления.

3. Дополнительное задание №1 (позволяет набрать +15 процентов от максимального числа баллов БаРС за данную лабораторную). Написать программу на любом языке программирования, которая бы на вход получала число в системе счисления "С" из примера 11, а на выходе вы выдавала это число в системе счисления "B" из примера 11. В случае выполнения этого задания предоставить листинг программы в отчёте.

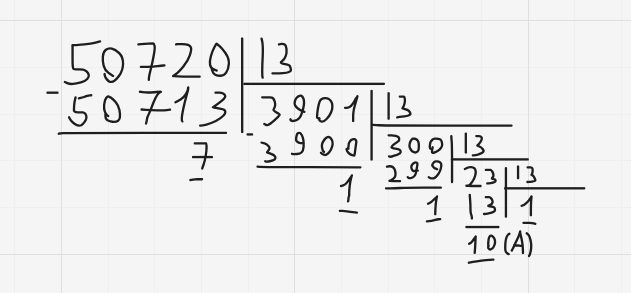
4. Оформить отчёт по лабораторной работе исходя из требований.

# **2. Решение задач**

## *Задание №1*

Дано число . Требуется перевести это число в 13-ричную СС.

*Решение*: при переводе чисел из 10-ричной СС в другие СС мы пользуемся методом последовательного деления. Для удобства применим деление столбиком:



### Рисунок 2.1: Последовательное деление на 13.

Запишем полученные остатки в обратном порядке, чтобы получить число в 13-ричной СС. Получаем: = .

*Ответ:*

## *Задание №2*

Дано число . Требуется перевести данное число в 10-ую СС

*Решение*: при переводе числа n-ной СС в 10-ую СС воспользуемся формулой записи числа в позиционной СС.

= 1\* + 2\*+2\*+8\*+5\* =

*Ответ:*

## *Задание №3*

Дано . Требуется перевести данное число в 13-ричную СС.

*Решение:*

1. Переведём число в 10-ричную СС:

= 5\* + 4\*+ 0\*+ 4\*+ 4\* = .

1. Получившееся число переведём в 13-ричную СС:

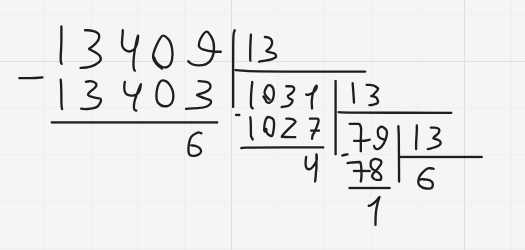


Рисунок 2.2: Последовательное деление на 13

Аналогично 1-ому заданию, записываем остатки в обратном порядке и получаем наш ответ:

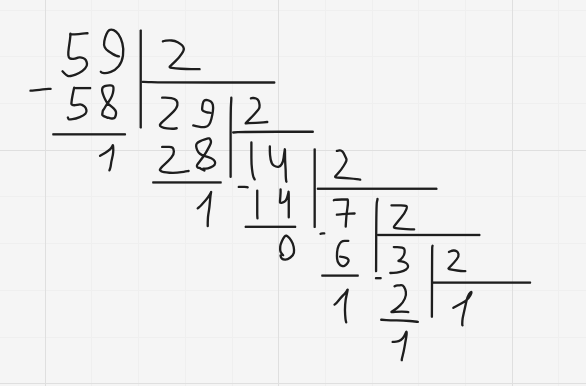
*Ответ:*

*Задание №4*

Дано число . Требуется перевести данное число в 2-ичную СС.

*Решение:* делим исходное число на целую и дробную части, которые мы должны рассмотреть отдельно.

*Целая часть* - . Последовательным делением переведём данное число в 2-ичную СС:



### Рисунок 2.3: последовательное деление на 2.

Получаем число, равное .

*Дробная часть* - . Для перевода дробной части воспользуемся методом умножения числа на основание новой СС и отделением целой части. Так как по условию задания нужно перевести число с точностью до 5 знаков, то проведём первые 6 итераций:

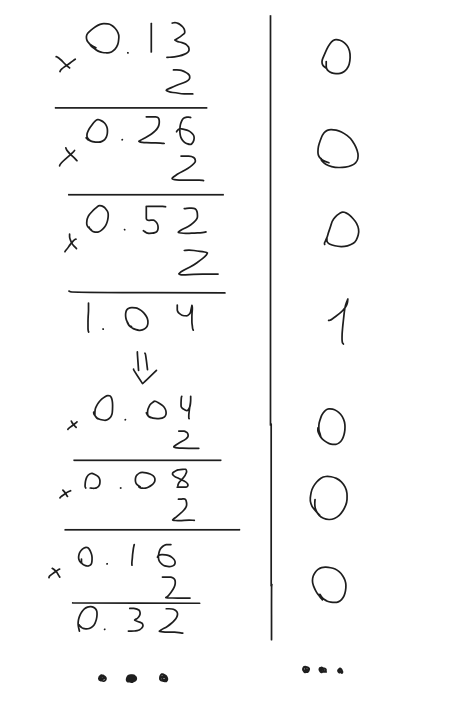


Рисунок 2.4: перевод в 2-ичную СС.

Получаем, что **≈** .

Объединяем итоговые части и получаем ответ:

**≈** .

*Ответ: .*

## *Задание №5*

Дано число . Требуется перевести данное число в 2-ичную СС.

*Решение:* решим задачу методом перевода числа -ричной СС в *N*-ричную СС. Аналогично предыдущему заданию, разделим число на дробную и целую части и рассмотрим их обособленно.

*Целая часть -* . По таблице оснований вида узнаём, что = . Тогда = .

*Дробная часть -* . По таблице оснований вида узнаём, что = . Следовательно, = .

Объединяем преобразованные части и получаем ответ: = .

*Ответ: .*

## *Задание №6*

Дано число . Требуется перевести данное число в 2-ичную СС.

*Решение:* делим число на целую и дробную части и по отдельности рассматриваем их.

*Целая часть - .* По таблице оснований вида узнаём, что = , = . Получаем, что = .

*Дробная часть -* . По таблице оснований вида узнаём, что = . Следовательно: = .

Объединяем итоговые части и получаем ответ: = .

*Ответ:*

## *Задание №7*

Дано число . Требуется перевести данное число в 16-ричную СС.

*Решение:* добавим в конец числа 2 незначащих нуля, чтобы количество цифр в дробной части было кратно = 4. Разбиваем дробную часть на фрагменты, длиной 4. Переведём их в 16-ричную СС:

= , = .

Соединяем получившиеся части и записываем ответ: = .

*Ответ: .*

## *Задание №8*

Дано число. Требуется перевести данное число в 10-ичную СС.

*Решение:* аналогично 2-ому заданию, воспользуемся формулой записи числа в позиционной СС:

= 1\* + 0\* + 0\* + 0\* + 1\* + 1\* = 35\*

35\*.

*Ответ: .*

## *Задание №9*

Дано число . Требуется перевести данное число в 16-ричную СС.

*Решение:* по подобию предыдущего задания, воспользуемся формулой записи числа в позиционной СС:

= 1\* + 9\* + 3\* + 15\* =

*Ответ: .*

## *Задание №10*

Дано число *.* Требуется перевести его в 10-ричную СС.

*Решение:* алгоритм перевода из факториальной СС в десятичную СС очень похож на алгоритм перевода из системы счисления с основанием N в десятичную, отличие заключается лишь в том, что множитель цифр является *k!*, где *k* – порядковый номер цифры. Тогда получаем:

= 2\*6! + 4\*5! + 4\*4! + 3\*3! + 2\*2! + 1\*1! =

*Ответ: .*

## *Задание №11*

Дано число . Требуется перевести данное число в СС Цекендорфа.

*Решение:* рассмотрим число . Оно меньше 233 - 12-е по списку число Фибоначчи (без учёта 1-й единицы). Значит, что в новой СС будет 11 цифр. Первая цифра в любом случае будет равна 1, остальные определяем “с помощью ума и смекалки”:

= 1\*144 + 0\*89 + 1\*55 + 0\*34 + 0\*21 + 0\*13 + 1\*8 + 0\*5 + 1\*3 + 0\*2 + 0\*1 =

*Ответ:*

## *Задание №12*

Дано число . Требуется перевести данное число в 10-ричную СС.

*Решение:* при переводе из СС Цекендорфа в 10-ричную СС воспользуемся формулой записи числа в позиционной СС:

= 1\*34 + 0\*21 + 0\*13 + 1\*8 + 0\*5 + 0\*3 + 1\*2 + 0\*1 =

*Ответ:*

## *Задание №13*

Дано число . Требуется перевести данное число в 10-ричную СС.

*Решение:* СС Бергмана основана на золотом сечении. Алгоритм перевода из СС Бергмана в десятичную СС очень похож на алгоритм перевода из системы счисления с основанием N в десятичную:

= 1\*+ 0\* + 0\* + 1\* + 0\* + 1\* + 0\* + 0\* + 1\* + 0\* + 0\* + 1\*

*Ответ:*

## *Дополнительное задание №1*

Написать программу на любом ЯП, которая на вход получала бы число СС Цекендорфа, а на выходе бы получала это же число в 10-ричной СС.

*Решение:* для поиска числа Фибоначчи воспользуемся матрицами, так как их умножение само на себя позволяет уменьшить нагрузку на поиск с n до :

def matrix\_mult(A, B):

    return [[A[0][0]\*B[0][0] + A[0][1]\*B[1][0],

            A[0][0]\*B[0][1] + A[0][1]\*B[1][1]], #умножение матриц

           [A[1][0]\*B[0][0] + A[1][1]\*B[1][0],

            A[1][0]\*B[0][1] + A[1][1]\*B[1][1]]]

def matrix\_power(matrix, power):

    result = [[1, 0], [0, 1]]

    while power:

        if power % 2 != 0: #возведение матрицы в степень

            result = matrix\_mult(result, matrix)

        matrix = matrix\_mult(matrix, matrix)

        power //= 2

    return result

def fib\_matrix(n):

    if n <= 1:

        return n #вычисление самого числа

    matrix = [[1, 1], [1, 0]]

    result = matrix\_power(matrix, n - 1)

    return result[0][0]

s\_fib = input("Введите число в системе счисления Цекендорфа: ")

if (s\_fib.replace('0', '').replace('1', '') == "" and "11" not in s\_fib and "1" in s\_fib):

    s\_fib = s\_fib.lstrip("0")[::-1] #обработка входного значения

    s\_10 = 0

    for i in range(0, len(s\_fib)):

        if s\_fib[i] == "1":

            s\_10+=fib\_matrix(i+2)

    print(f"Введённое число в 10-й системе счисления равняется: {s\_10}")

else: print("Ошибка: некорректный ввод числа")

# **3. Список Литературы**

1) Балакшин П.В., Соснин В.В., Машина Е.А. Информатика. – СПб: Университет ИТМО, 2020. – 122 с.

2) Abhinav Upadhyay - A Linear Algebra Trick for Computing Fibonacci Numbers Fast – URL: https://blog.codingconfessions.com/p/a-linear-algebra-trick-for-fibonacci-numbers