Министерство науки и высшего образования Российской Федерации Федеральное государственное автономное образовательное учреждение высшего образования «Национальный исследовательский университет ИТМО»

Факультет Программной инженерии и компьютерной техники

**­­­­­Лабораторная работа №1**

Перевод чисел между

различными системами счисления

Вариант №14

Группа: P3112

Выполнил: Балин А. А.

Проверил: к.т.н. преподаватель Белозубов А.В.

Оглавление

[Введение 3](#_Toc115230959)

[Задания 4](#_Toc115230960)

[Решение заданий 5](#_Toc115230961)

[Программа для перевода чисел между СС 9](#_Toc115230962)

[Вывод 12](#_Toc115230963)

[Список литературы 13](#_Toc115230964)

# Введение

В данной работе я выполняю перевод чисел из одной СС (здесь и далее – система счисления) в другую с помощью собственных вычислений по стандартным алгоритмам, проверяя себя с помощью написанной собственноручно программы на языке Python.

# Задания

|  |  |  |  |
| --- | --- | --- | --- |
| № | Само число | Начальная СС | Конечная СС |
| 1 | 76779 | 10 | 13 |
| 2 | 53255 | 7 | 10 |
| 3 | 53441 | 7 | 13 |
| 4 | 69,47 | 10 | 2 |
| 5 | 8A,63 | 16 | 2 |
| 6 | 36,37 | 8 | 2 |
| 7 | 0,110111 | 2 | 16 |
| 8 | 0,111011 | 2 | 10 |
| 9 | 14,12 | 16 | 10 |
| 10 | 430121 | Факт | 10 |
| 11 | 159 | 10 | Фиб |
| 12 | {4}{1}{4}{2}1 | 9C | 10 |
| 13 | 2656 | 10 | Факт |

# Решение заданий

№1

7677910 = X13

|  |  |
| --- | --- |
| Целая часть при делении на 13 | Остаток от деления на 13 |
| 5906 | 1 |
| 454 | 4 |
| 34 | 12 (C) |
| 2 | 8 |
| 0 | 2 |

Ответ: 7677910 = 28С4113

№2

532557 = X10

532557 = 5\*74+3\*73+2\*72+5\*71+5\*70 = 1317210

Ответ: 532557 = 1317210

№3

534417 = X13

1. 534417 = Y10

534417 = 5\*74+3\*73+4\*72+4\*71+1\*70 = 1325910

1. 1325910 = X13

|  |  |
| --- | --- |
| Целая часть при делении на 13 | Остаток от деления на 13 |
| 1019 | 12 (C) |
| 78 | 5 |
| 6 | 0 |
| 0 | 6 |

1325910 = 605C13

Ответ: 534417 = 605C13

№4

69,4710 = X2

1. 6910 = 64+4+1 = 26+22+20 = 10001012
2. 0,4710 = Y2

|  |  |
| --- | --- |
| Число | Целая часть |
| 0,47\*2 = 0,94 | 0 |
| 0,94\*2 = 1,88 | 1 |
| 0,88\*2 = 1,76 | 1 |
| 0,76\*2 = 1,52 | 1 |
| 0,52\*2 = 1,04 | 1 |
| 0,04\*2=0,08 | 0 |

0,4710 ≈ 0,0111102

Ответ: 69,4710 ≈ 1000101,011112

Примечание для № 5, 6, 7

Для указанных номеров я буду пользоваться таблицей (Рисунок 1) для перевода по сокращённому правилу.



Рисунок 1

№5

8A,6316 = X2

1. 8 → 1000
2. А → 1010
3. 6 → 0110
4. 3 → 0011
5. 8А,6316 → 10001010,011000112 ≈ 10001010,011002

Ответ: 8А,6316 ≈ 10001010,011002

№6

36,378 = X2

1. 3 → 011
2. 6 → 110
3. 3 → 011
4. 7 → 111
5. 36,378 → 011110,0111112 ≈ 11110,100002

Ответ: 36,378 ≈ 11110,100002

№7

0,1101112 = X16

1. 0,1101112 = 0,110111002
2. 1101 → D
3. 1100 → C
4. 0,110111002 → 0,DC16

Ответ: 0,1101112 = 0,DC16

№8

0,1110112 = X10

0,1110112 = 2–1 + 2–2 + 2–3 + 2–5 + 2–6 = 0,5 + 0,25 + 0,125 + 0,03125 + 0,015625 = 0,92187510 ≈ 0,9218810

Ответ: 0,1110112 ≈ 0,9218810

№9

14,1216 = X10

14,1216 = 1\*161 + 4\*160 + 1\*16–1 + 2\*16–2 = 16 + 4 + 0,0625 + 0,0078125 = 20,070312510 ≈ 20,0703110

Ответ: 14,1216 ≈ 20,0703110

№10

430121факт = X10

430121факт = 4\*6! + 3\*5! + 0\* 4! +1\*3! + 2\*2! + 1\*1! = 2880 + 360 + 0 + 6 + 4 + 1 = 325110

Ответ: 430121факт = 325110

№11

15910 = XФиб

1. Числа Фибоначчи: 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144…
2. 159 = 1\*144 + 0\*89 + 0\*55 + 0\*34 + 0\*21 + 1\*13 + 0\*8 + 0\*5 + 0\*3 + 1\*2 + 0\*1 = 10000100010Фиб

Ответ: 15910 = 10000100010Фиб

№12

{4}{1}{4}{2}19С = X10

{4}{1}{4}{2}19С = (–4) \* 94 + (–1) \* 93 +(–4) \*92 + (–2) \*91 + 1 \* 90 = –26244 + (– 729) + (– 324) + (– 18) + 1 = –2731410

Ответ: {4}{1}{4}{2}19С = –2731410

№13

265610 = Xфакт

2656 = 2160 + 480 + 12 + 4 = 3\*6! + 3\*5! + 0\*4! + 2\*3! + 2\*2! + 0\*1! = 330220факт

Ответ: 265610 = 330220факт

# Программа для перевода чисел между СС

Описание программы

Программа выполнена на языке Python. Интерактивно общается с пользователем, сообщает ему, что нужно ввести для правильной работы программы, ошибки обрабатываются и выводятся в консоль. Есть два режима работы: перевод между стандартными СС и перевод из 10С в Фибоначчиеву СС, выбор режима происходит в самом начале пользователем. Первый алгоритм способен переводить целые неотрицательные числа из СС p в СС q, где 1 <p, q <37, p и q целые. На вход подаются p, q и само число n через пробел, после выполнения алгоритма число выводится на экран или выдаётся ошибка. Второй обрабатывает входное число и выводит его в Фибоначчиевой СС.

Код программы

global Alphabet

Alphabet = '0123456789ABCDEFGHIGKLMNOPQRSTUVWXYZ'

def from\_p\_to\_q(p, q, n):

    return from\_10\_to\_q(q, from\_p\_to\_10(p, n),)

def from\_10\_to\_q(q, n):

    s = ''

    n = int(n)

    while n != 0:

        s = Alphabet[n % q] + s

        n = n//q

    return (s)

def from\_p\_to\_10(p, n):

    n = str(n)

    out = 0

    for i in range(len(n)):

        out += (Alphabet.index(n[len(n)-i-1]))\*p\*\*i

    return (out)

def check(p, q, n):

    if not (p.isnumeric) or not (q.isnumeric) or float(p) > 36 or float(q) > 36 or float(p) < 2 or float(q) < 2 or int(float(p)) != float(p) or int(float(q)) != float(q):

        return False

    p = int(float(p))

    for i in range(p, len(Alphabet)):

        if Alphabet[i] in n:

            return False

    for i in range(len(n)):

        if n[i] not in Alphabet:

            return False

    return True

def Fib\_CC(n):

    if n==0:

        return 0

    a = [1, 2]

    while a[-1] <= n:

        a.append(a[-1]+a[-2])

    out = ''

    for i in range(len(a)-1, -1, -1):

        if a[i] <= n:

            out += '1'

            n -= a[i]

        else:

            out += '0'

    return out

s = ''

while s != '1' and s != '2':

    s = input(

        'Выберите режим:\n 1. CC c целыми основаниями p,q ∈ [2;36] \n 2. СС Цекендорфа (фибоначчиева СС)\n (Введите 1 или 2)\n')

if s == '1':

    S = input(

        'Введите через пробел начальное основание СС, конечное и само число\n').split()

    if len(S) != 3:

        print("Неверный формат ввода!")

    else:

        p, q, n = map(str, S)

        if check(p, q, n):

            p = int(float(p))

            q = int(float(q))

            n = int(n)

            print(from\_p\_to\_q(p, q, n))

        else:

            print('Проверьте возможность существования числа в заданных СС')

else:

    s = input(

        'Введите число в десятичной системе счисления, чтобы перевести его в СС Цекендорфа:\n')

    if s.isnumeric() and int(s)==float(s) and int(s)>=0:

        print(int(Fib\_CC(int(s))))

    else:

        print("Неверный формат ввода!")

# Вывод

Я потренировался переводить числа между различными СС, научился делать это несколькими способами. Опираясь на алгоритм, который я использую сам для перевода, я смог написать алгоритм для такого перевода. Наиболее трудной частью показалось исключение человеческого фактора в программе. Самая интересная идея: использовать объявленный глобально алфавит для удобного перевода СС с основанием больше 10.

# Список литературы

[В Интернете] / авт. Справочник24 // Справочник студенческий. - 2022 г.. - https://spravochnick.ru/assets/files/articles/inf31.png.

**Требования к оформлению отчета** [В Интернете] / авт. Белозубов Александр Владимирович. - 14 Сентябрь 2021 г.. - https://docs.google.com/document/d/14uQaMlKxcORgtCVijschvSnpTdHhZ8D6/edit.