

УНИВЕРСИТЕТ ИТМО

ФАКУЛЬТЕТ ПРОГРАММНОЙ ИНЖЕНЕРИИ И КОМПЬЮТЕРНОЙ ТЕХНИКИ

НАПРАВЛЕНИЕ ПОДГОТОВКИ «ПРОГРАММНАЯ ИНЖЕНЕРИЯ»

ОТЧЁТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №1

курса «Информатика»

Вариант 21

Выполнил студент:

Берестовский Святослав Сергеевич

группа: Р3111

Преподаватель:

Малышева Татьяна Алексеевна

Санкт-Петербург, 2021 г.

Содержание

1. Перевод чисел между различными системами счисления	2
1.1. Цель работы	2
1.2. Задание	2
1.3. Основная часть	2
1.3.1. Исходный код	2
1.3.2. Результат работы программы	4
1.4. Выводы	4
Литература	5

Лабораторная работа 1

Перевод чисел между различными системами счисления

1.1. Цель работы

Разобраться с принципом и основными техниками перевода чисел между разными системами счисления.

1.2. Задание

Перевести число A , заданное в системе счисления B , в систему счисления C . 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.

Таблица : Значения переменных

Задание	A	B	C
1	34106	10	15
2	16116	7	10
3	21104	5	15
4	51,96	10	2
5	41,6C	16	2
6	14,67	8	2
7	0,001101	2	16
8	0,001011	2	10
9	1B,08	16	10
10	42	10	Fib
11	147	-10	10
12	1000010101	Fib	10
13	1678	-10	10

1.3. Основная часть

1.3.1. Исходный код

Исходный код скрипта доступен также на моем GitHub: <https://github.com/anim3boy/itmo-labs/blob/main/computer-science/script.py>.

Листинг 1.1: Исходный код

```

1 from string import digits as numbers, ascii_uppercase as letters
2 from math import log
3 from itertools import product
4
5 def base_translate(source, a=10, b=10):
6     ALPHABET = numbers + letters
7
8     int_part_of_decimal, fract_part_of_decimal = 0, 0
9     int_part_of_source, fract_part_of_source = "", ""
10
11     if a != 10:
12         # a -> 10 int(source, 10)
13         if ',' in source:
14             int_part_of_source, fract_part_of_source = source.split(',')
15         else:
16             int_part_of_source = source
17
18         for i in enumerate(reversed(int_part_of_source)):
19             index, number = i
20             int_part_of_decimal += int(ALPHABET.index(number)) * (a ** index)
21
22         for i in enumerate(fract_part_of_source):
23             index, number = i
24             index = -(index + 1)
25             fract_part_of_decimal += int(ALPHABET.index(number)) * (a ** index)
26         decimal = int_part_of_decimal + fract_part_of_decimal
27
28     else:
29         decimal = float(source.replace(',', '.'))
30
31     # 10 -> b
32     if b == 10:
33         return str(int(decimal * 10 ** 5) / 10 ** 5).replace('.', ',') \
34             if int(decimal) != decimal else decimal
35
36     fract_part_of_decimal = decimal - int(decimal)
37     int_part_of_decimal = int(decimal)
38
39     int_part_of_result, fract_part_of_result = "", ""
40
41     for i in range(5):
42         fract_part_of_result += str(ALPHABET[int(fract_part_of_decimal*b)])
43         fract_part_of_decimal = fract_part_of_decimal*b - int(fract_part_of_decimal*b)
44
45     while int_part_of_decimal > 0:
46         int_part_of_decimal, m = divmod(int_part_of_decimal, b)
47         int_part_of_result += ALPHABET[m]
48
49     int_part_of_result = "0" if not int_part_of_result else int_part_of_result
50     while fract_part_of_result and fract_part_of_result[-1] == "0":
51         fract_part_of_result = fract_part_of_result[:-1].replace('0', '', 1)[-1]
52     return '%s%s%s' % (int_part_of_result[:-1], ',', fract_part_of_result)
53
54 def translate(source, a=10, b=10):
55     if a == b: return source
56     if b == 'Fib':
57         return fib_encoding(source, int(a))
58     if a == 'Fib':
59         return fib_decoding(source, int(b))
60     a, b = int(a), int(b)
61     if a > 0 and b > 0 and (int(log(b) / log(a)) == log(b) / log(a) \
62         or int(log(a) / log(b)) == log(a) / log(b)) and ',' not in source: ###
63         # "easy way"
64         block_length = int(log(b, a))
65         padding = len(source) % block_length
66         if padding:
67             source = "0" * (block_length - padding) + source
68         blocks = [source[i : i + block_length] for i in range(0, len(source), block_length)]
69         return ''.join([base_translate(i, a, b) for i in blocks])
70     else:
71         return base_translate(source, a, b)
72
73 def fib_encoding(n, a):
74     if a != 10: n = int(translate(n, a, 10))
75     n = int(n)
76     ALPHABET = []
77     a, b = 0, 1
78     while a <= n:
79         ALPHABET.append(a)
80         a, b = b, a + b
81     ALPHABET = sorted(list(set(ALPHABET[1:])))
82
83     for i in product([0,1], repeat=len(ALPHABET)):
84         if sum(i[j] * ALPHABET[j] for j in range(len(ALPHABET))) == n \

```

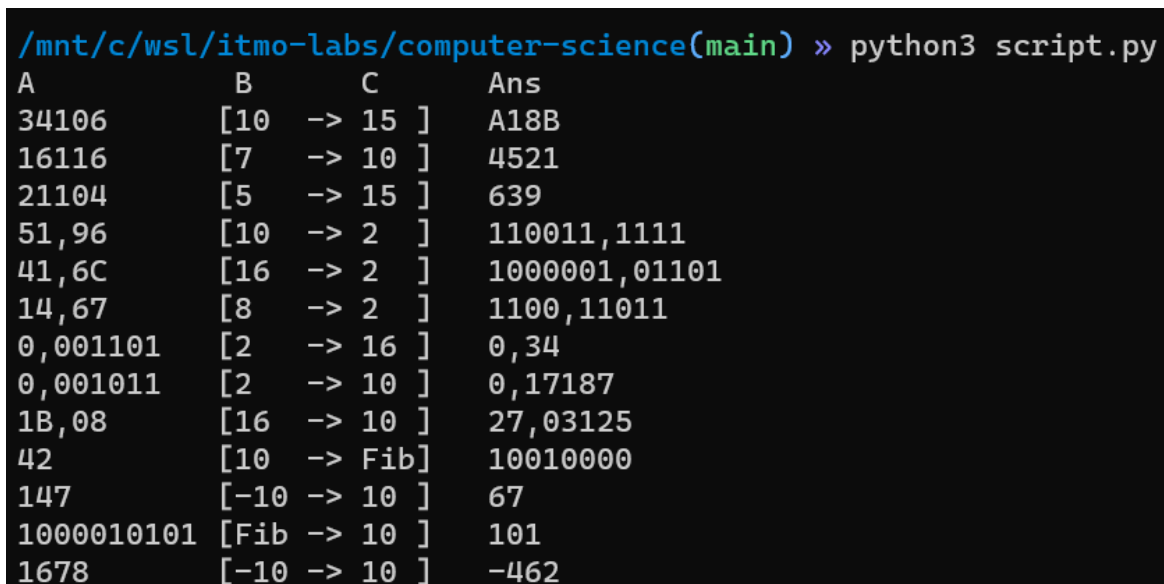
```

85         and '11' not in ''.join(list(map(str, i))):
86         return ''.join(list(map(str, i))[:-1])
87
88 def fib_decoding(n, b):
89     ALPHABET = []
90     a, b = 0, 1
91     for i in range(len(n) + 2):
92         ALPHABET.append(a)
93         a, b = b, a + b
94     ALPHABET = sorted(list(set(ALPHABET[1:])))
95
96     return sum(int(n[j]) * ALPHABET[::-1][j] for j in range(len(ALPHABET)))
97
98 with open('s.txt') as f:
99     s = list(map(lambda x: x[:-1].split(), f.readlines()))
100    print('{<10}  {<3}    {<3}    {}'.format("A", "B", "C", "Ans"))
101    for i in s:
102        print('{<10} [{<3} -> {<3}]    {}'.format(i[0], i[1], i[2], translate(i[0], i[1], i[2])))

```

Функции были реализованы с использованием алгоритмов из материалов лекции. Я проигнорировал ряд функций из стандартной библиотеки языка, например перевод чисел в десятичную СС с использованием `int()`

1.3.2. Результат работы программы



```

/mnt/c/wsl/itmo-labs/computer-science(main) » python3 script.py
A          B          C          Ans
34106      [10  -> 15 ]      A18B
16116      [7   -> 10 ]      4521
21104      [5   -> 15 ]      639
51,96      [10  -> 2  ]      110011,1111
41,6C      [16  -> 2  ]      1000001,01101
14,67      [8   -> 2  ]      1100,11011
0,001101   [2   -> 16 ]      0,34
0,001011   [2   -> 10 ]      0,17187
1B,08      [16  -> 10 ]      27,03125
42         [10  -> Fib]      10010000
147        [-10 -> 10 ]      67
1000010101 [Fib -> 10 ]      101
1678       [-10 -> 10 ]      -462

```

Рис. : Скриншот фрагмента сессии в ZSH

1.4. Выводы

В процессе выполнения данной лабораторной работы я систематизировал имеющиеся знания по переводу чисел между разными системами счисления, реализовал скрипт на питоне и поработал над читаемостью кода

Литература

- [1] Алексеев Е.Г., Богатырев С.Д. Информатика. Мультимедийный электронный учебник.
URL: <http://inf.ealekseev.ru/text/toc.html>
- [2] Орлов С. А., Цилькер Б. Я. Организация ЭВМ и систем: Учебник для вузов. 2-е изд. – СПб.: Питер, 2011. – 688 с.: ил.