

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 №20

Выполнил:
Решетников Сергей Евгеньевич
Группа Р3108
Проверил:
Преподаватель ПИиКТ
Бострикова Дарья Константиновна

Санкт-Петербург 2024

Оглавление

1. Задание.....	3
2. Основные этапы вычисления.....	3
2.1 $68981_{10}=?_7$	3
2.2 $40403_5=?_{10}$	3
2.3 $B9235_{15}=?_5$	4
2.4 $58,88_{10}=?_2$	4
2.5 $BA,12_{16}=?_2$	5
2.6 $34,43_8=?_2$	5
2.7 $0,111101_2=?_{16}$	5
2.8 $0,100001_2=?_{10}$	5
2.9 $52,A1_{16}=?_{10}$	5
2.10 $613301_{(факт)}=?_{10}$	5
2.11 $229_{10}=?_{(фиб)}$	5
2.12 $10100000_{(фиб)}=?_{10}$	5
2.13 $100010.001001_{(Берг)}=?_{10}$	6
3. Дополнительное задание.....	6
4. Вывод.....	7
5. Источники.....	7

1. Задание

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 Р3102 соответствует 15-й вариант ($=02 + 13$). Если полученный вариант больше 40, то необходимо вычесть из него 40. Т.е. 21-му человеку из группы Р3121 соответствует 2-й вариант ($=21 + 21 - 40$).

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.

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 "В" из примера 11. В случае выполнения этого задания предоставить листинг программы в отчёте.

2. Основные этапы вычисления

2.1 $68981_{10}=?_7$

Число	Остаток
68981	3
9854	5
1407	0
201	5
28	0
4	4

$68981_{10}=405053_7$

2.2 $40403_5=?_{10}$

$$40403_5 = 3 \cdot 5^0 + 0 \cdot 5^1 + 4 \cdot 5^2 + 0 \cdot 5^3 + 4 \cdot 5^4 = 2603_{10}$$

2.3 B9235₁₅=?₅

$$B9235_{15} = 11 \cdot 15^4 + 9 \cdot 15^3 + 2 \cdot 15^2 + 3 \cdot 15^1 + 5 \cdot 15^0 = 587750_{10}$$

Число	Остаток
587750	0
117550	0
23510	0
4702	2
940	0
188	3
37	2
7	2
1	1

$$587750_{10} = 122302000_5$$

$$B9235_{15} = 122302000_5$$

2.4 58,88₁₀=?₂

Перевод целой части:

Число	Остаток
58	0
29	1
14	0
7	1
3	1
1	1

$$58_{10} = 111010_2$$

Перевод дробной части:

Число	Домножение на основание	Целая часть
0,88	0,88*2=1,76	1
0,76	0,76*2=1,52	1
0,52	0,52*2=1,04	1
0,04	0,04*2=0,08	0
0,08	0,08*2=0,16	0

Т.к. точность – 5 знаков после запятой, то 58,88₁₀=111010,11100₂

2.5 $BA,12_{16}=?_2$

$$BA,12_{16}=10111010,0001001_2$$

Т.к. точность – 5 знаков после запятой, то $BA,12_{16}=10111010,00010_2$

2.6 $34,43_8=?_2$

$$34,43_8=11100,100011_2$$

Т.к. точность – 5 знаков после запятой, то $34,43_8=11100,10001_2$

2.7 $0,111101_2=?_{16}$

$$0,111101_2=0,F4$$

2.8 $0,100001_2=?_{10}$

$$0,100001_2=2^{-1}+2^{-6}=0,515625$$

Т.к. точность – 5 знаков после запятой, то $0,100001_2=0,51563_{10}$

2.9 $52,A1_{16}=?_{10}$

$$52,A1_{16}=5*16^1 + 2*16^0 + 10*16^{-1}+1*16^{-2} = 80 + 2 + 0,625 + 0,00390625 = 82,6289063_{10}$$

Т.к. точность – 5 знаков после запятой, то $52,A1_{16}=82,62891_{10}$

2.10 $613301_{(факт)}=?_{10}$

$$613301_{(факт)}=1*1! + 0*2! + 3*3! + 3*4! + 1*5! + 6*6!= 1 + 3*6 + 3*24 + 120 + 720*6 = 1 + 18 + 72 + 120 + 4320 = 4531_{10}$$

2.11 $229_{10}=?_{(фиб)}$

Ряд Фибоначчи (со 2-го члена): 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233

$$229 = 144 + 55+21+8+1$$

$$229_{10}=10101010001_{(фиб)}$$

2.12 $10100000_{(фиб)}=?_{10}$

Ряд Фибоначчи (со 2-го члена): 1, 2, 3, 5, 8, 13, 21, 34, 55, 89

$$10100000_{(фиб)}=13+34=47_{10}$$

2.13 $100010.001001_{(\text{Берн})} = ?_{10}$

$100010.001001_{(\text{Берн})} = z^5 + z^1 + z^{-3} + z^{-6} \approx 11.090169943 + 1.618033988 + 0.236067977 + 0.055728090 = 12.999999998$

Т.к. точность – 5 знаков после запятой, то $100010.001001_{(\text{Берн})} = 13_{10}$

3. Дополнительное задание

Ниже представлен листинг фрагмента кода программы (см. Рисунок 1), которая может переводить из Фибоначчевой системы счисления в десятичную.

```
convert > Converters > fib.py > Convert_fib > convert_to
1 class Convert_fib:
2
3     def __init__(self) -> None:
4         sqrt_5 = 5**0.5
5         phi = (1 + sqrt_5)/2
6         self.calc_n_fib = lambda n: round(
7             (phi ** (n + 2) - (-phi) ** (-1*(n + 2))) / sqrt_5
8         )
9         # Формула Бине с округлением (почти) по правилам математики
10        # и поправкой на +2 члена (коррекция для СС + учёт нумерации в циклах)
11
12    @staticmethod
13    def create_fib_array(max_num: int) -> list:
14        """
15        Функция для создания массива чисел Фибоначчи не превосходящих данного
16        """
17        result = [1]
18        while max_num > result[-1]:
19
20            # 2-ой элемент массива добавляется этим ветвлением (чтоб корректно обработать max_num = 1)
21            if result == [1]:
22                result.append(2)
23                continue
24            result.append(result[-1]+result[-2])
25        return result
26
27    def convert_to(self, num: str) -> str:
28        """
29        Конвертация десятичного числа в Фибоначчевую СС
30        """
31        num = int(num)
32        result = ""
33        for elem in self.create_fib_array(num)[::-1]:
34
35            # проверяем, подходит ли число + исключение подряд идущих 1
36            if num - elem >= 0 and (result == "" or result[-1] != "1"):
37                num -= elem
38                result += "1"
39            else: result += "0"
40        return int(result) # убираем лишние нули
41
42    def convert_from(self, num: str) -> str:
43        """
44        Конвертация из Фибоначчевой СС в десятичную
45        """
46        result = 0
47        for num_idx, num_sym in enumerate(num[::-1]):
48            result += int(num_sym) * self.calc_n_fib(num_idx)
49        return result
```

Рисунок 1 — Листинг кода

Программа также может конвертировать из и в следующие системы счисления: Бергмана, Фибоначчева, симметрические СС, с основанием -10 и десятичную.

Полный код доступен по ссылке -
https://github.com/NF-coder/ITMO_repo/tree/main/informatics/sem1/lab1

4. Вывод

В ходе выполнения лабораторной работы я закрепил свои знания по переводу чисел между классическими системами счисления, а также узнал о симметрических, негепозиционных и факториальной системах счисления, научился переводить числа в них.

5. Источники

1. Информатика [Электронный ресурс]: Википедия. Свободная энциклопедия. – URL:
https://ru.wikipedia.org/wiki/Троичная_система_счисления#Симметричная_троичная_система_счисления (дата обращения: 20.09.2024).
2. Алексеев Е. Г., Богатырев С. Д. Информатика: Мультимедийный электронный учебник. Саранск: 2009.