



Федеральное государственное автономное образовательное

учреждение высшего образования

«Национальный исследовательский университет ИТМО»

Факультет программной инженерии и компьютерной техники

Направление подготовки: 09.03.04 – Системное и прикладное программное обеспечение

Дисциплина «Информатика»

Отчёт по лабораторной работе №1

Перевод чисел между различными системами счисления

Вариант №17

Выполнил: студентка группы Р3115

Галак Екатерина Анатольевна

Проверил:

Белокон Юлия Алексеевна

Санкт – Петербург, 2024

Оглавление

Задание:	3
Основные этапы вычисления:	4
Задание 1	4
Задание 2	4
Задание 3	4
Задание 4	5
Задание 5	6
Задание 6	7
Задание 7	8
Задание 8	8
Задание 9	8
Задание 10	8
Задание 11	8
Задание 12	9
Задание 13	9
Дополнительное задание	9
Заключение	16
Список используемых источников	16

Задание:

Перевести число "А", заданное в системе счисления "В", в систему счисления "С".
Числа "А", "В" и "С" взять из таблицы 1.

Таблица 1

№	А	В	С
1	25334	10	9
2	22211	5	10
3	3CAAD	15	5
4	53,54	10	2
5	72,98	16	2
6	25,11	8	2
7	0,011111	2	16
8	0,000001	2	10
9	7A,87	16	10
10	142121	Факториальная	10
11	175	10	Фибоначчи
12	10100010	Фибоначчи	10
13	1000001.000001	Бергмана	10

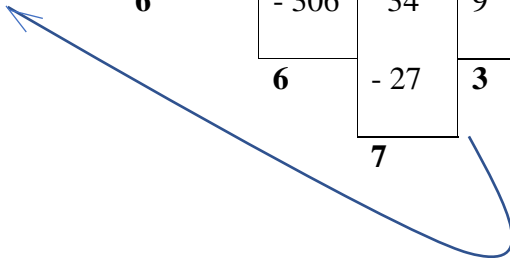
Всего нужно решить 13 примеров. Для примеров с 5-го по 7-й выполнить операцию перевода по сокращенному правилу (для систем с основанием 2 в системы с основанием 2^k). Для примеров с 4-го по 6-й и с 8-го по 9-й найти ответ с точностью до 5 знака после запятой.

Основные этапы вычисления:

Задание 1

$25334_{10} = x_9$, найти x

25334	9			
- 25326	2814	9		
8	- 2808	312	9	
	6	- 306	34	9
		6	- 27	3
			7	



$$x = 37668$$

Ответ: $25334_{10} = 37668_9$

Задание 2

$22211_5 = x_{10}$, найти x

$$22211_5 = 2 * 5^4 + 2 * 5^3 + 2 * 5^2 + 1 * 5^1 + 1 * 5^0 = 1250 + 250 + 50 + 5 + 1 = 1556_{10}$$

$$x = 1556$$

Ответ: $22211_5 = 1556_{10}$

Задание 3

$3CAAD_{15} = x_5$, найти x

Сначала переведём $3CAAD_{15}$ из пятнадцатиричной системы счисления в десятичную.

$$3CAAD_{15} = 3 * 15^4 + 12 * 15^3 + 10 * 15^2 + 10 * 15^1 + 13 * 15^0 = 151875 + 40500 + 2250 + 150 + 13 = 194788_{10}$$

Теперь переведём 194788_{10} из десятичной системы счисления в пятеричную.

$$53_{10} = 110101_2$$

Перевод дробной части:

$$0,54 * 2 = 1,08$$

$$0,08 * 2 = 0,16$$

$$0,16 * 2 = 0,32$$

$$0,32 * 2 = 0,64$$

$$0,64 * 2 = 1,28$$

$$53,54_{10} = 110101,10001_2 \text{ (точность 5 знаков после запятой)}$$

$$x = 110101,10001$$

$$\text{Ответ: } 53,54_{10} = 110101,10001_2$$

Задание 5

$$72,98_{16} = x_2, \text{ найти } x$$

Используя таблицу перевода из шестнадцатеричной системы счисления в двоичную и обратно (рисунок 1), решим задание, воспользовавшись методом сокращенного перевода систем счисления, основание которых являются степенью 2.

Шестнадцатеричная система	Двоичная система
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001
A	1010
B	1011
C	1100
D	1101
E	1110
F	1111

(Рисунок 1 – Таблица перевода из двоичной системы счисления в шестнадцатеричную систему счисления и обратно)

$$7_{16} = 0111_2$$

$$2_{16} = 0010_2$$

$$9_{16} = 1001_2$$

$$8_{16} = 1000_2$$

$$72,98_{16} = 0111\ 0010, 1001\ 1000_2 = 1110010,10011_2$$

$$x = 1110010,10011$$

$$\text{Ответ: } 72,98_{16} = 1110010,10011_2$$

Задание 6

$$25,11_8 = x_2, \text{ найти } x$$

Используя таблицу перевода из двоичной системы счисления в восьмеричную и обратно (рисунок 2), решим задание, воспользовавшись методом сокращенного перевода систем счисления, основание которых являются степенью 2.

Двоичная <-> Восьмеричная
000 <-> 0
001 <-> 1
010 <-> 2
011 <-> 3
100 <-> 4
101 <-> 5
110 <-> 6
111 <-> 7

(Рисунок 2 – Таблица перевода из двоичной системы счисления в восьмеричную систему счисления и обратно)

$$2_8 = 010_2$$

$$5_8 = 101_2$$

$$1_8 = 001_2$$

$$\text{Целая часть: } 25_8 = 010101_2 = 10101_2$$

$$\text{Дробная часть: } 0,11_8 = 0,001001_2$$

Теперь округлим дробную часть для получения точности в 5 знаков после запятой:

$$0,001001_2 \approx 0,00101_2$$

$$25,11_8 = 10101,00101_2$$

$$x = 10101,00101$$

Ответ: $25,11_8 = 10101,00101_2$

Задание 7

$0,011111_2 = x_{16}$, найти x

Используя таблицу перевода в шестнадцатеричную систему счисления (рисунок 1), решим задание, воспользовавшись методом сокращенного перевода систем счисления, основание которых являются степенью 2.

$$0,011111_2 = 0,0111\ 1100_2 = 0,7C_{16}$$

$$x = 0,7C$$

Ответ: $0,011111_2 = 0,7C_{16}$

Задание 8

$0,000001_2 = x_{10}$, найти x

$$0,000001_2 = 1 * 2^{-6} = 0,015625_{10} \approx 0,01563_{10}$$

$$x = 0,01563$$

Ответ: $0,000001_2 \approx 0,01563_{10}$

Задание 9

$7A,87_{16} = x_{10}$, найти x

$$7A,87_{16} = 7 * 16^1 + 10 * 16^0 + 8 * 16^{-1} + 7 * 16^{-2} = 112 + 10 + \frac{1}{2} + \frac{7}{256} \approx 122,52734_{10}$$

$$x \approx 122,52734_{10}$$

Ответ: $7A,87_{16} \approx 122,52734_{10}$

Задание 10

$142121_{\text{ФАКТ}} = x_{10}$, найти x

$$142121_{\text{ФАКТ}} = 1 * 1! + 2 * 2! + 1 * 3! + 2 * 4! + 4 * 5! + 1 * 6! = 1 + 4 + 6 + 48 + 480 + 720 = 1259_{10}$$

$$x = 1259$$

Ответ: $142121_{\text{ФАКТ}} = 1259_{10}$

Задание 11

$175_{10} = x_{\text{Фиб}}$, найти x

Ряд Фибоначчи до последнего числа, меньшего 175:

1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144

$$175_{10} = 144 * 1 + 89 * 0 + 55 * 0 + 34 * 0 + 21 * 1 + 13 * 0 + 8 * 1 + 5 * 0 + 3 * 0 + 2 * 1 + 1 * 0 \Rightarrow 175_{10} = 10001010010_{\text{Фиб}}$$

$$x = 10001010010$$

$$\text{Ответ: } 175_{10} = 10001010010_{\text{Фиб}}$$

Задание 12

$$10100010_{\text{Фиб}} = x_{10}, \text{ найти } x$$

Последовательность Фибоначчи:

1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, ...

$$10100010_{\text{Фиб}} = 1 * 0 + 2 * 1 + 3 * 0 + 5 * 0 + 8 * 0 + 13 * 1 + 21 * 0 + 34 * 1 = 2 + 13 + 34 = 49_{10}$$

$$x = 49$$

$$\text{Ответ: } 10100010_{\text{Фиб}} = 49_{10}$$

Задание 13

$$1000001.000001_{\text{Берг}} = x_{10}, \text{ найти } x$$

$$1000001.000001_{\text{Берг}} = z^6 + z^0 + z^{-6}, \text{ где } z = \frac{1 + \sqrt{5}}{2}$$

$$1000001.000001_{\text{Берг}} \approx 17,94427 + 1 + 0,0557281 \approx 19_{10}$$

$$x = 19$$

$$\text{Ответ: } 1000001.000001_{\text{Берг}} \approx 19_{10}$$

Дополнительное задание

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.

Программа написана на языке программирования C++:

```
#include <iostream>
#include <string>
#include <cstring>
```

```

#include <algorithm>
#include <vector>

enum class NumberSystem {
    Fibonacci,
    Factorial,
    Classic
};

void ThrowError(const std::string& string_error) {
    throw std::invalid_argument("Input error: " + string_error);
}

void TypeSystem(NumberSystem& type_system_, std::string& number_system_)
{
    if (std::all_of(number_system_.begin(), number_system_.end(),
::isdigit) ||
        number_system_[0] == '-' && std::all_of(number_system_.substr(1,
std::string::npos).begin(),
        number_system_.substr(1, std::string::npos).end(), ::isdigit)) {

        type_system_ = NumberSystem::Classic;
    } else if (number_system_ == "Fib" || number_system_ == "Fibonacci")
    {
        type_system_ = NumberSystem::Fibonacci;
    } else if (number_system_ == "Fac" || number_system_ == "Factorial")
    {
        type_system_ = NumberSystem::Factorial;
    } else if (std::find(number_system_.begin(), number_system_.end(),
'S') == number_system_.end() - 1) {
        type_system_ = NumberSystem::Classic;

        std::string system_without_S = "";
        system_without_S = number_system_.substr(0, number_system_.size()
- 1);

        if (!(std::stoi(system_without_S) % 2)) {
            ThrowError("Symmetric number system does not happen with an
even base");
        }

        number_system_ = system_without_S;
    }
}

```

```

    } else {
        ThrowError(number_system_ + " - this number system is not
supported");
    }
}

void ParsingNumberVector(std::vector<int>& vector_number, const
std::string& number) {
    bool is_negative = false;
    for (auto digit_char : number) {
        if (digit_char == '{') {
            is_negative = true;
            continue;
        }
        if (digit_char == '}') {
            is_negative = false;
            continue;
        }

        vector_number.push_back(std::isdigit(digit_char) ? (digit_char -
'0') : (digit_char - 'A' + 10));

        if (is_negative) {
            vector_number[vector_number.size() - 1] *= -1;
        }
    }
}

std::vector<int> DecimalToClassical(int old_number, int future_system) {
    std::vector<int> ans_num;

    do {
        ans_num.push_back(((old_number % future_system) + future_system)
% future_system);
        old_number /= future_system;
    } while (old_number != 0);

    std::reverse(ans_num.begin(), ans_num.end());

    return ans_num;
}

```

```

void ClassicalToDecimal(std::vector<int>& old_number, int old_system) {
    int degree = 1;
    int ans = 0;

    for (int i = old_number.size() - 1; i >= 0; --i) {
        ans += degree * old_number[i];
        degree *= old_system;
    }

    std::vector<int> ans_vector;
    ParsingNumberVector(ans_vector, std::to_string(ans));

    old_number.resize(ans_vector.size());
    for (int i = 0; i < ans_vector.size(); ++i) {
        old_number[i] = ans_vector[i];
    }
}

std::vector<int> DecimalToFib(int old_number) {
    std::vector<int> fib_posl = {1, 2};

    int size_posl_fib = 2;

    while (fib_posl[size_posl_fib - 1] < old_number) {
        fib_posl.push_back(fib_posl[size_posl_fib - 1] +
fib_posl[size_posl_fib - 2]);
        ++size_posl_fib;
    }

    std::vector<int> ans;

    --size_posl_fib;

    while (old_number != 0) {
        if (old_number >= fib_posl[size_posl_fib - 1]) {
            ans.push_back(1);
            old_number -= fib_posl[size_posl_fib - 1];
        } else {
            ans.push_back(0);
        }
    }
}

```

```

        --size_posl_fib;
    }

    while (size_posl_fib--) {
        ans.push_back(0);
    }

    return ans;
}

std::vector<int> DecimalToFact(int old_number) {
    std::vector<int> fact = {1};

    int i = 2;
    int size_fact = 1;
    while (fact[size_fact - 1] < old_number) {
        fact.push_back(fact[size_fact - 1] * i);
        ++size_fact;
        ++i;
    }

    --size_fact;
    std::vector<int> ans;

    while (old_number != 0) {
        if (old_number >= fact[size_fact - 1]) {
            ans.push_back(old_number / fact[size_fact - 1]);
            old_number -= (old_number / fact[size_fact - 1]) *
fact[size_fact - 1];
        } else {
            ans.push_back(0);
        }

        --size_fact;
    }

    while (size_fact--) {
        ans.push_back(0);
    }

    return ans;
}

```

```

    }

    void GenAns(const std::vector<int>& vec_ans) {
        for (auto i : vec_ans) {
            if (i > 9) {
                std::cout << ('A' + i - 10);
                continue;
            }

            std::cout << i;

            std::cout << std::endl;
        }

        void FromClassic(const NumberSystem& type_system_future, std::string&
            number_system_old,
                std::string& number_system_future, const std::string&
            number) {

            std::vector<int> vector_number;

            ParsingNumberVector(vector_number, number);

            ClassicalToDecimal(vector_number, std::stoi(number_system_old));

            int vector_to_num = 0;
            int pow_10 = 1;
            for (int i = vector_number.size() - 1; i >= 0; --i) {
                vector_to_num += vector_number[i] * pow_10;
                pow_10 *= 10;
            }

            if (type_system_future == NumberSystem::Classic) {
                GenAns(DecimalToClassical(vector_to_num,
                    std::stoi(number_system_future)));
            } else if (type_system_future == NumberSystem::Factorial) {
                GenAns(DecimalToFact(vector_to_num));
            } else {
                GenAns(DecimalToFib(vector_to_num));
            }
        }
    }
}

```

```

void FromFib(const NumberSystem& type_system_future,
            std::string& number_system_future, const std::string&
number) {

    std::vector<int> fib = {1, 2};

    int int_num = 0;
    for (int i = 0; i < number.size(); ++i) {
        int_num += fib[i] * (number[number.size() - i - 1] - '0') ;
        fib.push_back(fib[fib.size() - 1] + fib[fib.size() - 2]);
    }

    std::string old_system = "10";
    FromClassic(type_system_future, old_system, number_system_future,
std::to_string(int_num));
}

void FromFact(const NumberSystem& type_system_future,
            std::string& number_system_future, const std::string&
number) {

    int calc_dec = 0;
    int factorial = 1;

    for (int i = 0; i < number.size(); ++i) {
        calc_dec += factorial * (number[number.size() - i - 1] - '0');
        factorial *= i + 2;
    }

    std::string old_system = "10";
    FromClassic(type_system_future, old_system, number_system_future,
std::to_string(calc_dec));
}

main() {
    std::string number_;
    std::string number_system_now_;
    std::string number_system_future_;

    std::cin >> number_ >> number_system_now_ >> number_system_future_;

```

```

NumberSystem type_system_now_;
NumberSystem type_system_future_;

TypeSystem(type_system_now_, number_system_now_);
TypeSystem(type_system_future_, number_system_future_);

switch (type_system_now_) {
    case NumberSystem::Classic:
        FromClassic(type_system_future_, number_system_now_,
number_system_future_, number_);
        break;
    case NumberSystem::Factorial:
        FromFact(type_system_future_, number_system_future_,
number_);
        break;
    case NumberSystem::Fibonacci:
        FromFib(type_system_future_, number_system_future_, number_);
        break;
    default:
        break;
}
}

```

Заключение

Во время выполнения лабораторной работы по переводу из одной системы счисления в другую я изучила и повторила различные методы и алгоритмы перевода, а также узнала о системах счисления Фибоначчи, Факториальной и Бергмана.

Список используемых источников

1. Балакшин Е.А., Соснин П.В., Информатика. Методическое пособие – Санкт-Петербург, 2015
2. Гашков С. Б. Системы счисления и их применение / С. Б. Гашков. — М. : МЦНМО, 2004