

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 P3115

Галак Екатерина Анатольевна

Проверил:

Белокон Юлия Алексеевна

Санкт – Петербург, 2024

Оглавление

[Задание: 3](#_Toc177765473)

[Основные этапы вычисления: 4](#_Toc177765474)

[Задание 1 4](#_Toc177765475)

[Задание 2 4](#_Toc177765476)

[Задание 3 4](#_Toc177765477)

[Задание 4 5](#_Toc177765478)

[Задание 5 6](#_Toc177765479)

[Задание 6 7](#_Toc177765480)

[Задание 7 7](#_Toc177765481)

[Задание 8 8](#_Toc177765482)

[Задание 9 8](#_Toc177765483)

[Задание 10 8](#_Toc177765484)

[Задание 11 8](#_Toc177765485)

[Задание 12 9](#_Toc177765486)

[Задание 13 9](#_Toc177765487)

[Дополнительное задание 9](#_Toc177765488)

[Заключение 16](#_Toc177765489)

[Список используемых источников 16](#_Toc177765490)

# Задание:

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

Таблица 1

|  |  |  |  |
| --- | --- | --- | --- |
| № | A | B | C |
| 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

2533410 = x9, найти x

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

x = 37668

Ответ: 2533410 = 376689

## Задание 2

222115 = x10, найти x

222115 = 2 \* 54 + 2 \* 53 + 2 \* 52 + 1 \* 51 + 1 \* 50 = 1250 + 250 + 50 + 5 + 1 = 155610

x = 1556

Ответ: 222115 = 155610

## Задание 3

3CAAD15 = x5, найти x

Сначала переведём 3CAAD15 из пятнадцатиричной системы счисления в десятичную.

3CAAD15 = 3 \* 154 + 12 \* 153 + 10 \* 152 + 10 \* 151 + 13 \* 150 = 151875 + 40500 + 2250 + 150 + 13 = 19478810

Теперь переведём 19478810 из десятичной системы счисления в пятеричную.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 194788 | 5 |  |  |  |  |  |  |
| - 194785 | 38957 | 5 |  |  |  |  |  |
| **3** | - 38955 | 7791 | 5 |  |  |  |  |
|  | **2** | - 7790 | 1558 | 5 |  |  |  |
|  |  | **1** | - 1555 | 311 | 5 |  |  |
|  |  |  | **3** | - 310 | 62 | 5 |  |
|  |  |  |  | **1** | - 60 | 12 | 5 |
|  |  |  |  |  | **2** | - 10 | **2** |
|  |  |  |  |  |  | **2** |  |

19478810 = 222131235 => 3CAAD15 = 222131235

x = 22213123

Ответ: 3CAAD15 = 222131235

## Задание 4

53,5410 = x2, найти x

Перевод целой части числа:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 53 | 2 |  |  |  |  |
| - 52 | 26 | 2 |  |  |  |
| **1** | - 26 | 13 | 2 |  |  |
|  | **0** | - 12 | 6 | 2 |  |
|  |  | **1** | - 6 | 3 | 2 |
|  |  |  | **0** | - 2 | **1** |
|  |  |  |  | **1** |  |

5310 = 1101012

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

53,5410 = 110101,100012 (точность 5 знаков после запятой)

x = 110101,10001

Ответ: 53,5410 = 110101,100012

## Задание 5

72,9816 = x2, 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.

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

716 = 01112

216 = 00102

916 = 10012

816 = 10002

72,9816 = 0111 0010, 1001 10002 = 1110010,100112

x = 1110010,10011

Ответ: 72,9816 = 1110010,100112

## Задание 6

25,118 = x2, 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.

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

28 = 0102

58 = 1012

18 = 0012

Целая часть: 258 = 0101012 = 101012

Дробная часть: 0,118 = 0,0010012  
Теперь округлим дробную часть для получения точности в 5 знаков после запятой:

0,0010012 ≈ 0,001012

25,118 = 10101,001012

x = 10101,00101

Ответ: 25,118 = 10101,001012

## Задание 7

0,0111112 = x16, найти x

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

0,0111112 = 0, 0111 11002 = 0,7C16

x = 0,7C

Ответ: 0,0111112 = 0,7C16

## Задание 8

0,0000012 = x10, найти x

0,0000012 = 1 \* 2-6 = 0,01562510 ≈ 0,0156310

x = 0,01563

Ответ: 0,0000012 ≈ 0,0156310

## Задание 9

7A,8716 = x10, найти x

7A,8716 = 7 \* 161 + 10 \* 160 + 8 \* 16-1 + 7 \* 16-2 = 112 + 10 + + ≈ 122,5273410

x ≈ 122,5273410

Ответ: 7A,8716 ≈ 122,5273410

## Задание 10

142121Факт = x10, найти x

142121Факт = 1 \* 1! + 2 \* 2! + 1 \* 3! + 2 \* 4! + 4 \* 5! + 1 \* 6! = 1 + 4 + 6 + 48 + 480 + 720 = 125910

x = 1259

Ответ: 142121Факт = 125910

## Задание 11

17510 = xФиб, найти x

Ряд Фибоначчи до последнего числа, меньшего 175:

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

17510 = 144 \* 1 + 89 \* 0 + 55 \* 0 + 34 \* 0 + 21 \* 1 + 13 \* 0 + 8 \* 1 + 5 \* 0 + 3 \* 0 + 2 \* 1 + 1 \* 0 => 17510 = 10001010010Фиб

x = 10001010010

Ответ: 17510 = 10001010010Фиб

## Задание 12

10100010Фиб = x10, найти x

Последовательность Фибоначчи:

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

10100010Фиб = 1 \* 0 + 2 \* 1 + 3 \* 0 + 5 \* 0 + 8 \* 0 + 13 \* 1 + 21 \* 0 + 34 \* 1 = 2 + 13 + 34 = 4910

x = 49

Ответ: 10100010Фиб = 4910

## Задание 13

1000001.000001Берг = x10, найти x

1000001.000001Берг = z6 + z0 + z-6, где z =

1000001.000001Берг ≈ 17,94427 + 1 + 0,0557281 ≈ 1910

x = 19

Ответ: 1000001.000001Берг ≈ 1910

## 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 "B" 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