

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра вычислительной техники

ОТЧЕТ
по лабораторной работе №1
по дисциплине «Организация ЭВМ и систем»
Тема: ИССЛЕДОВАНИЕ ВНУТРЕННЕГО ПРЕДСТАВЛЕНИЯ РАЗЛИЧНЫХ
ФОРМАТОВ ДАННЫХ

Студент гр. 1361	_____	Голубев Д. В.
Студентка гр. 1361	_____	Горбунова Д. А.
Преподаватель	_____	Гречухин М. Н.

Санкт-Петербург
2022

Цель работы

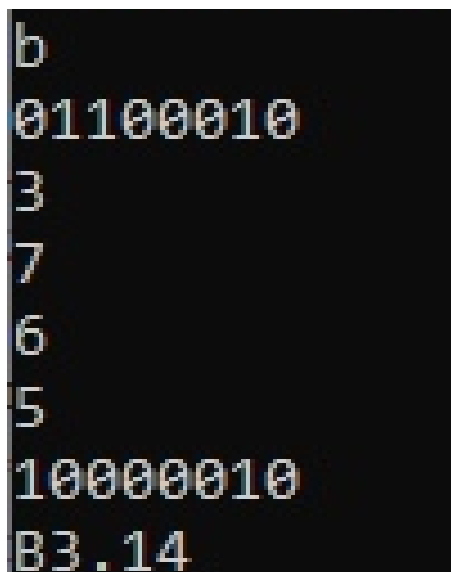
Знакомство с внутренним представлением различных типов данных, используемых компьютером при их обработке.

Текст задания

1. Разработать алгоритм ввода с клавиатуры требуемых типов данных (char и double) и показать их внутреннее представление в двоичной системе счисления на экране.
2. Написать и отладить программу на языке C++, реализующую разработанный алгоритм.
3. Инвертировать значения определённого количества бит, номера которых, как и их количество, вводится с клавиатуры.

Результаты запуска программы

Программа на вход принимает один элемент. Выводит этот элемент в битовом виде и принимает на вход биты, которые нужно инвертировать. Далее программа выводит полученный элемент в битовом и десятичном виде. Результат работы приведен на рисунке 1.



```
b
01100010
3
7
6
5
10000010
83.14
```

Рисунок 1 – Консоль при работе с unsigned int

Программа на вход принимает одно вещественное число. Выводит

это число в битовом виде и принимает на вход биты, которые нужно инвертировать. Далее программа выводит полученное число в битовом и десятичном виде. Результат работы приведен на рисунке 2.

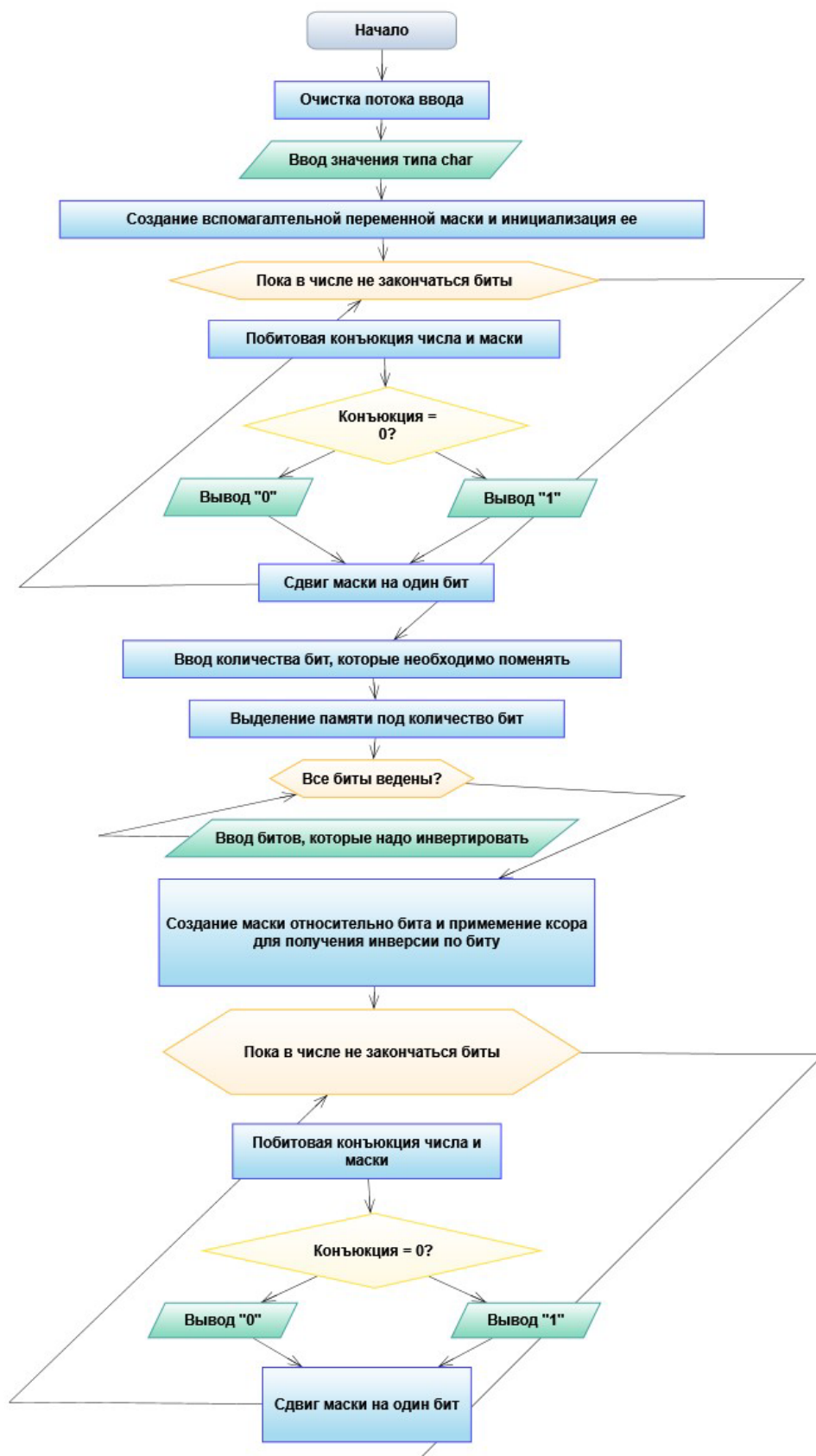
```

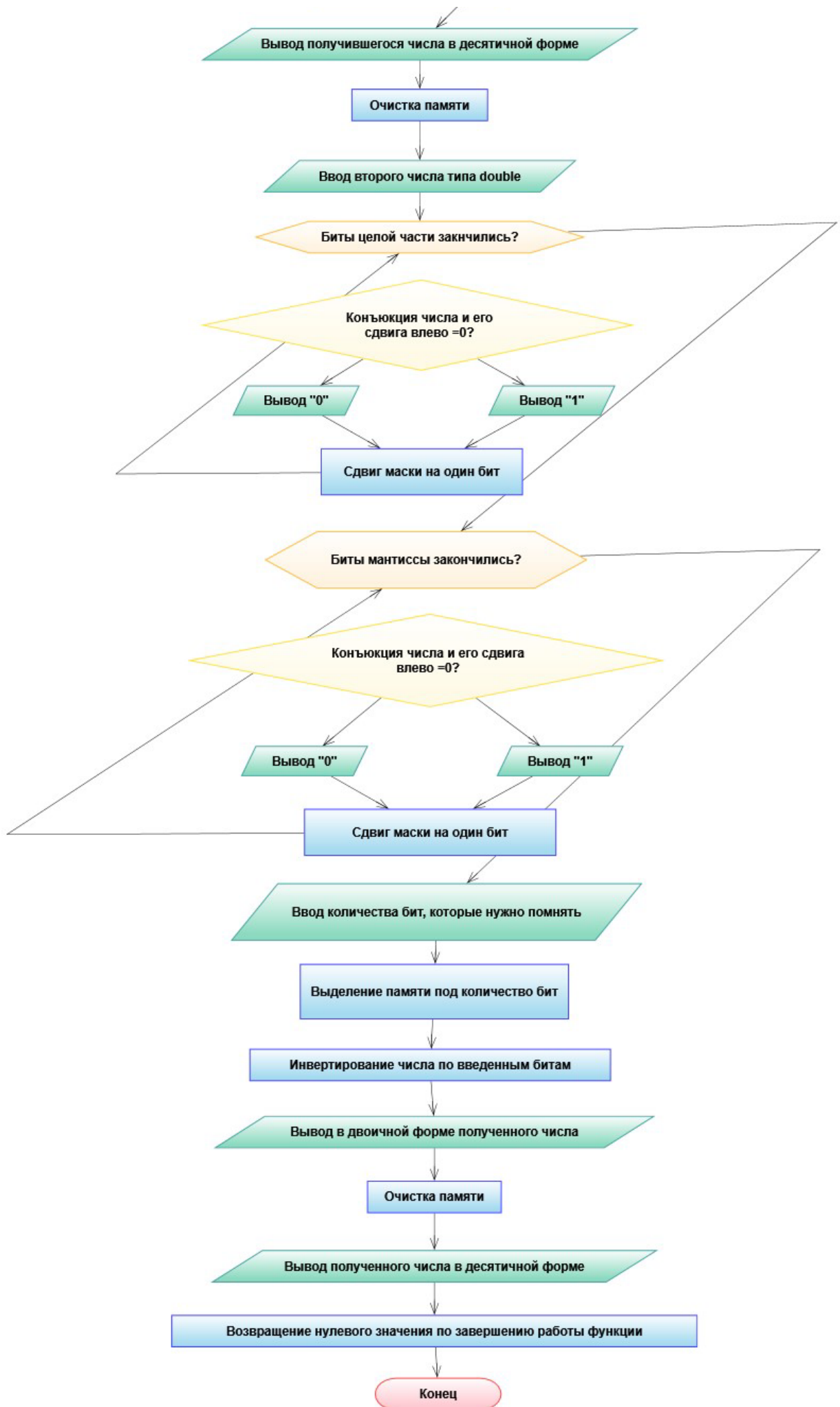
B3.14
0100000000001001000111101011100001010001111010111000010100011111
1
63
1100000000001001000111101011100001010001111010111000010100011111
-3.140000
Process returned 0 (0x0)   execution time : 20.500 s
Press any key to continue.

```

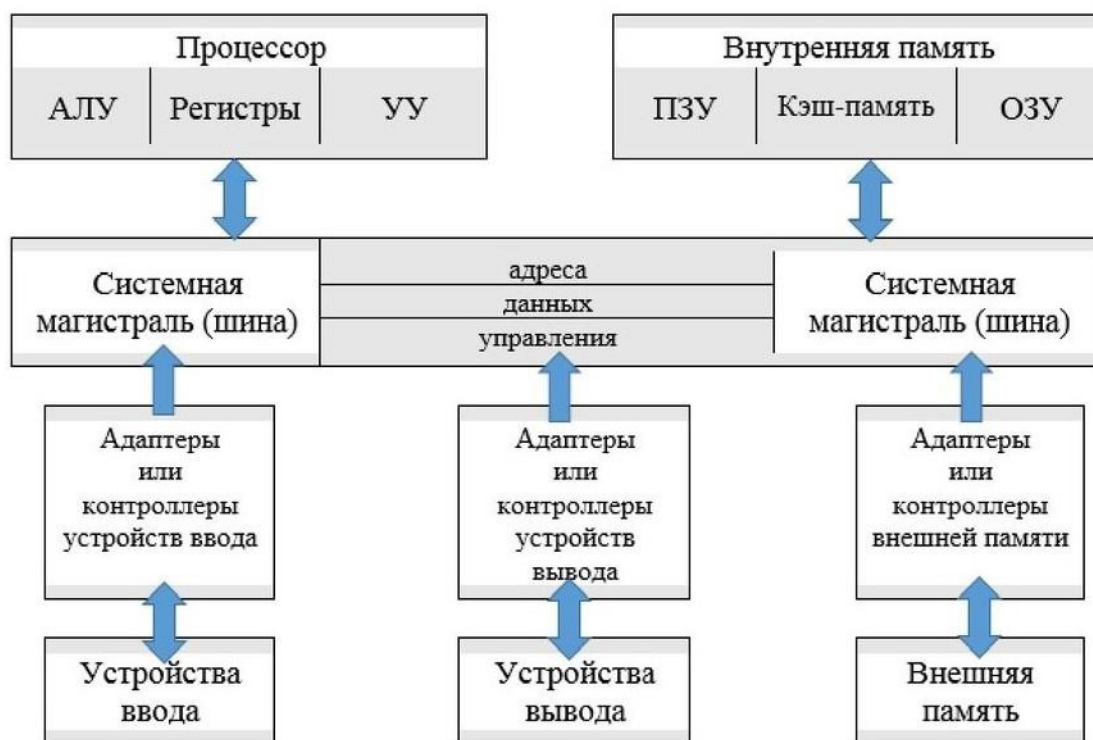
Рисунок 2 – Консоль при работе с long double

Приложение 1. Блок-схема реализуемого кода





Приложение 2. Структурная схема аппаратных средств



Исходный код программы

```
#include <stdio.h>
#include <stdlib.h>
#include <malloc.h>
#include <math.h>

union {
    struct {
        int  int1;
        int  int2;
    };
    double a;
}double_int;

int main()
{
    int i;
    int j;
    int x;
    int lent;
    unsigned char locator;
    unsigned char num;
    unsigned char *mass;
    char art;
    unsigned char mask;
    unsigned char res;
    fflush (stdin);
    scanf("%c", &art);
    fflush (stdin);
    mask = 128;
    for (i = 0; i < 8; i++){
        res = art & mask;
        if (res == 0){
            printf("0");
        } else{
            printf("1");
        }
        mask = mask >> 1;
    }
    printf("\n");
    scanf("%d", &x);
    mass = malloc(x);
    for (i = 0; i < x; i++){
        scanf("%d", &mass[i]);
    }
    for (i = 0; i < x; i++){
        num = pow(2, mass[i]);
        art = art^num;
    }
    mask = 128;
    for (i = 0; i < 8; i++){
        res = art & mask;
        if (res == 0){
            printf("0");
        } else{
            printf("1");
        }
        mask = mask >> 1;
    }
    printf("\n");
    printf("%c", art);
    free(mass);
    scanf("%lf", &double_int.a);
    lent = 32;
    for (int i = 0; i < lent; ++i) {
        if ((double_int.int2 & 1 << lent - i - 1) == 0) printf("0");
    }
}
```

```

        else printf("1");
    }
    for (int i = 0; i < lent; ++i) {
        if ((double_int.int1 & 1 << lent - i - 1) == 0) printf("0");
        else printf("1");
    }
    printf("\n");
    scanf("%d", &x);
    mass = malloc(x);
    for (i = 0; i < x; i++){
        scanf("%d", &mass[i]);
    }
    for (i = 0; i < x; i++){
        if (mass[i] < 32){
            double_int.int1 = double_int.int1 ^ (1 << mass[i]);
        } else {
            double_int.int2 = double_int.int2 ^ (1 << (mass[i]%32));
        }
    }
    lent = 32;
    for (int i = 0; i < lent; ++i) {
        if ((double_int.int2 & 1 << lent - i - 1) == 0) printf("0");
        else printf("1");
    }
    for (int i = 0; i < lent; ++i) {
        if ((double_int.int1 & 1 << lent - i - 1) == 0) printf("0");
        else printf("1");
    }
    printf("\n");
    free(mass);
    printf("%lf", double_int.a);
    return 0;
}

```