

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

ОТЧЕТ

по лабораторной работе №1

по дисциплине «Организация ЭВМ и систем»

Тема: Исследование внутреннего представления различных форматов данных.

Студенты гр. 3372

Беляев К.В.
Лазарев Ф.Н.

Преподаватель

Кочетков А.В.

Санкт-Петербург

2024

Задание на лабораторную работу

1. В зависимости от номера варианта задания разработать алгоритм ввода с клавиатуры требуемых типов данных и показать на экране их внутреннее представление в двоичной системе счисления.

2. Написать и отладить программу на языке C++, реализующую разработанный алгоритм.

3. В соответствии с заданием дополнить разработанный ранее алгоритм блоками для выполнения преобразования двоичного полученного кода исходного типа данных и последующего вывода преобразованного кода в двоичной системе счисления и в формате исходного данного. Задание: установить в заданное пользователем состояние определённое количество бит, номера которых, как и всё остальное, вводится с клавиатуры.

Блок-схема алгоритма

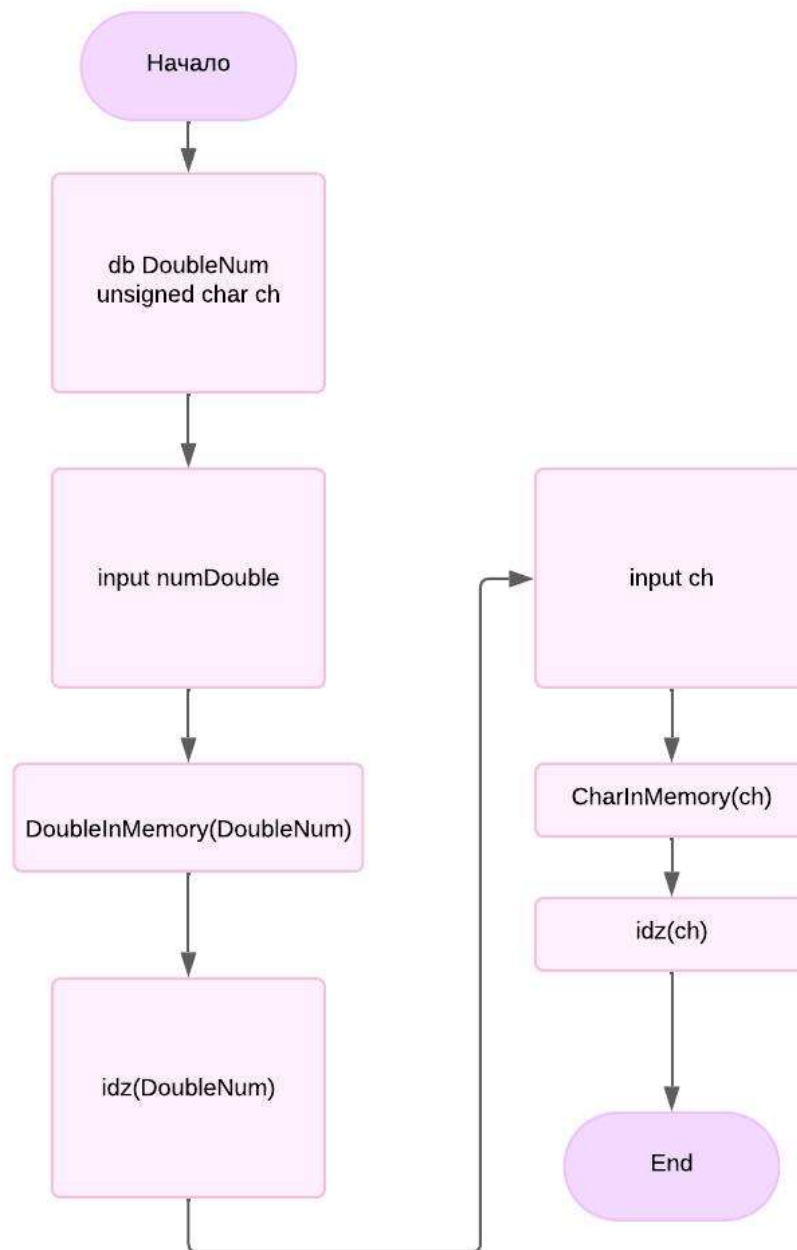


Рис. 1. Основной алгоритм программы

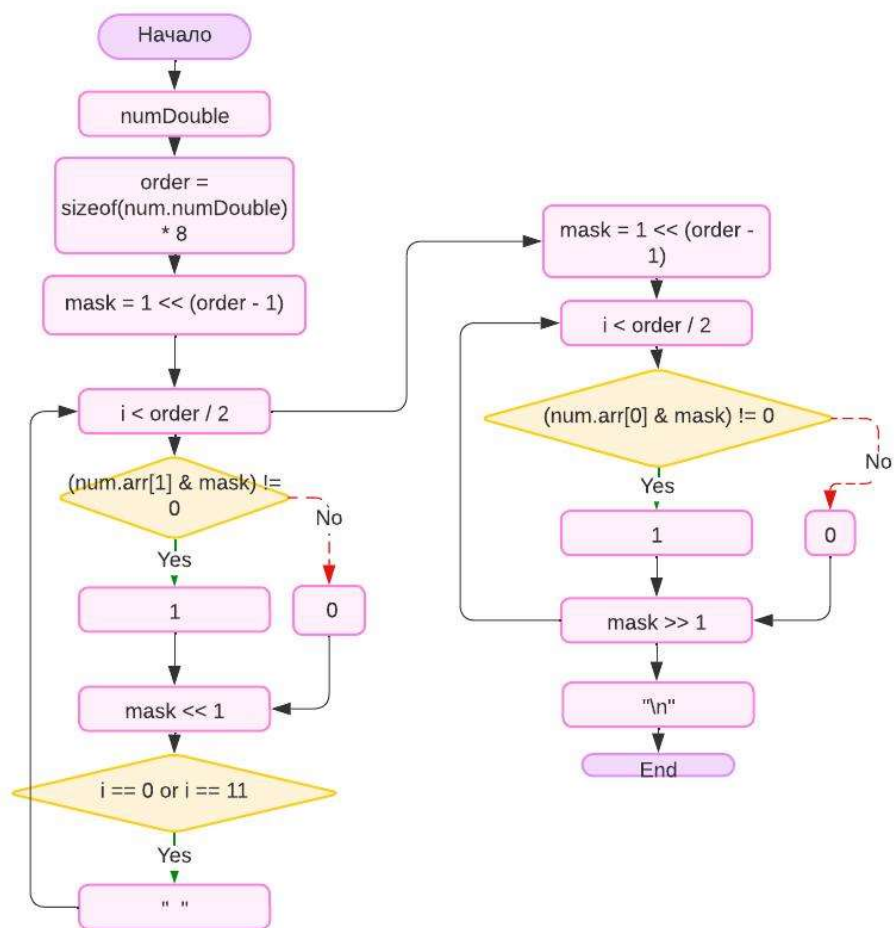


Рис. 2. Алгоритм функции *DoubleInMemory*

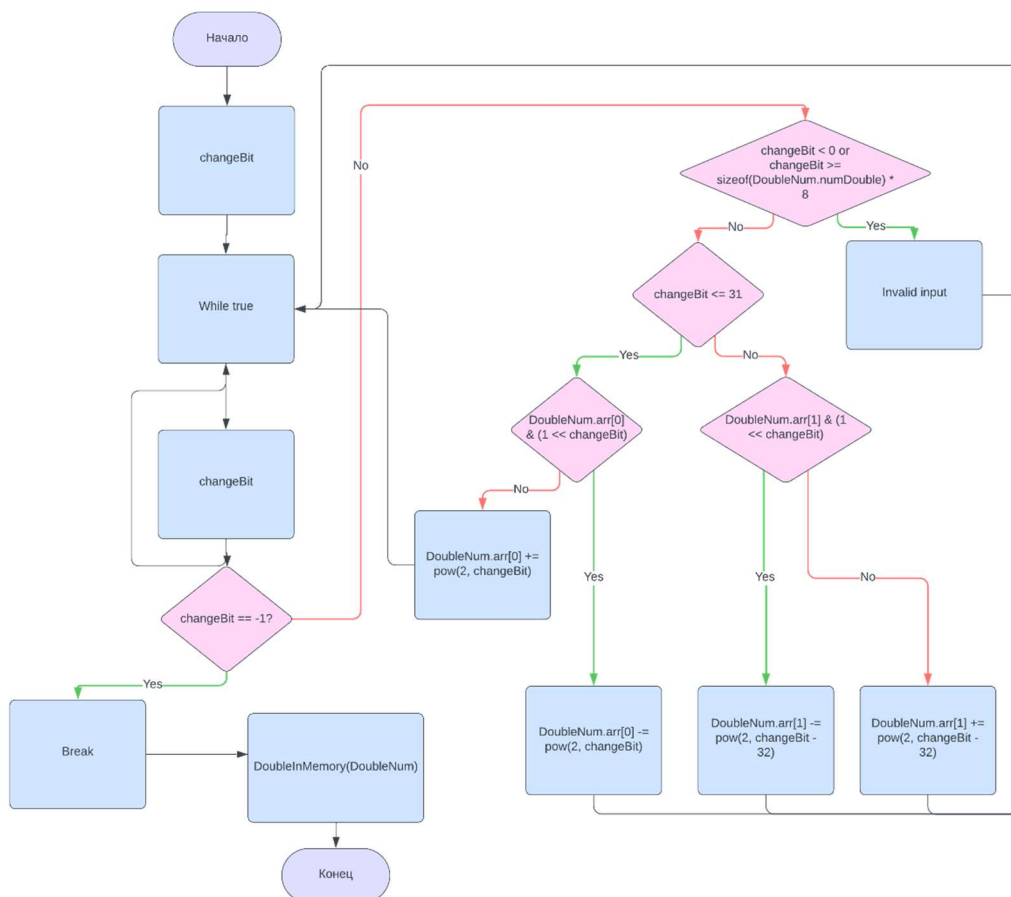


Рис. 3. Алгоритм *idz(DoubleNum)*

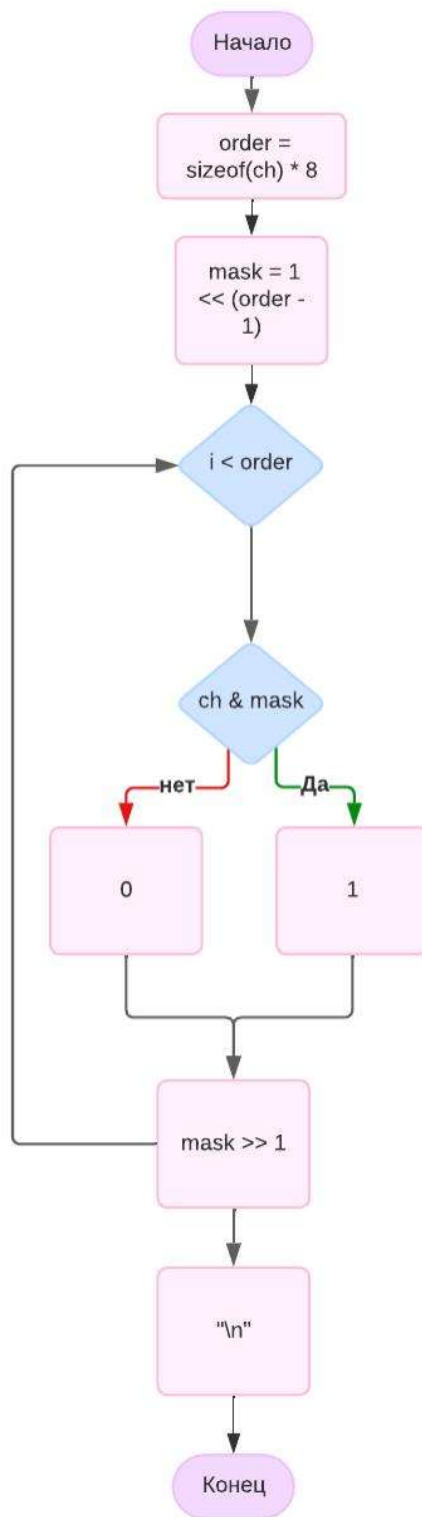


Рис. 4. Алгоритм CharInMemory

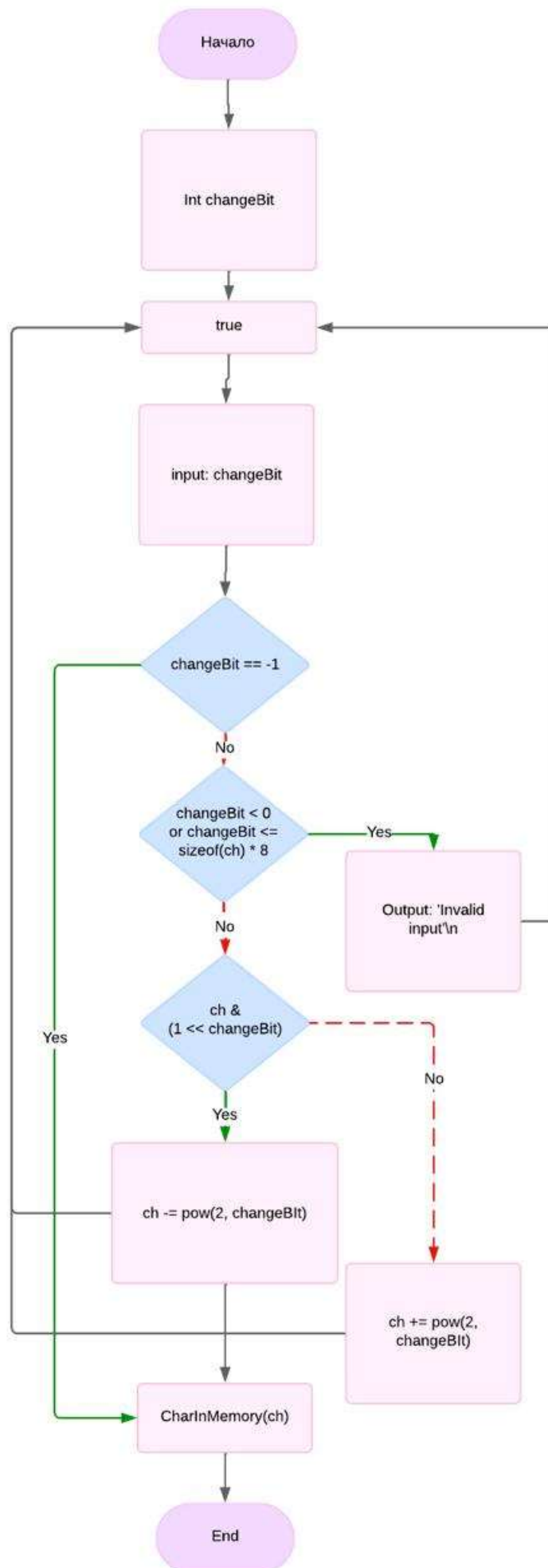


Рис. 5. Алгоритм $idz(ch)$

Текст программы

```
#include "iostream"

using namespace std;

int order;

unsigned int mask;

union db{
    long double numDouble;
    int arr[2];
};

void DoubleInMemory(union db num){
    cout << "\nДвоичное представление числа " << num.numDouble << ":\n";
    order = sizeof(num.numDouble) * 8;
    mask = 1 << (order - 1);

    for (int i = 0; i < order / 2; ++i){
        cout << ((num.arr[1] & mask)? 1 : 0);
        mask >>= 1;
        if (!i || i == 11){
            cout << " ";
        }
    }

    mask = 1 << (order - 1);

    for (int i = 0; i < order / 2; ++i){
        cout << ((num.arr[0] & mask)? 1 : 0);
        mask >>= 1;
    }
    cout << "\n";
}
```

```

void CharInMemory(unsigned char ch){
    order = sizeof(ch) * 8;
    mask = 1 << (order - 1);
    for ( int i = 0; i < order; i++ ){
        cout << ((ch & mask) ? 1 : 0);
        mask >>= 1;
    }
    cout << "\n";
}

```

```

void idz(db DoubleNum){
    int changeBit;
    while (true) {
        cout << "Введите номер бита: ";
        cin >> changeBit;
        if (changeBit == -1) {
            break;
        } else if (changeBit < 0 || changeBit >= sizeof(DoubleNum.numDouble) * 8 {
            cout << "Неверный ввод\n";
        } else {
            if ( changeBit <= 31 ){
                if ( DoubleNum.arr[0] & (1 << changeBit)){
                    DoubleNum.arr[0] -= pow(2, changeBit);
                }else{
                    DoubleNum.arr[0] += pow(2, changeBit);
                }
            }else{
                if ( DoubleNum.arr[1] & (1 << changeBit)){
                    DoubleNum.arr[1] -= pow(2, changeBit - 32);
                }else{
                    DoubleNum.arr[1] += pow(2, changeBit - 32);
                }
            }
        }
    }
    DoubleInMemory(DoubleNum);
}

```



```

void idz(unsigned char ch){
    int changeBit;
    while (true) {
        cout << "Введите номер бита: ";
        cin >> changeBit;
        if (changeBit == -1) {
            break;
        } else if (changeBit < 0 || changeBit >= sizeof(ch) * 8 {
            cout << "Неверный ввод\n";
        } else {
            if ( ch & (1 << changeBit) ){
                ch -= pow(2, changeBit);
            }else{
                ch += pow(2, changeBit);
            }
        }
    }
    CharInMemory(ch);
}

```

```

int main(){
    db DoubleNum;
    unsigned char ch;

    cout << "Введите число типа long double: ";
    cin >> DoubleNum.numDouble;
    DoubleInMemory(DoubleNum);
    idz(DoubleNum);
    cout << "\n";

    cout << "Введите числа типа unsigned char: ";
    cin >> ch;
    CharInMemory(ch);
    idz(ch);
    cout << "\n";

}

```

Примеры запуска программы

[illegible]

Рис. 6. Работа с вещественным типом данных

```
Введите числа типа unsigned char: 3
00110011
Введите номер бита: 5
Введите номер бита: -1
00010011
```

Рис. 7. Работа с символьным типом данных