МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение

высшего образования

УЛЬЯНОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

Кафедра «Вычислительная техника»

Дисциплина «М и МАПР»

**Лабораторная работа №1**

**«Разработка ПО для правого побитового сдвига»**

Выполнил:

студентка 4 курса, гр.

ИВТАПбд-41

Матвеев П. О.

Проверил:

Войт Н. Н.

г. Ульяновск, 2017

**1. Задание**

Разобраться со схемой работы побитового сдвига вправо и разработать ПО, осуществляющий побитовый сдвиг влево в 10-ти битном числе.

**2. Выполнение работы**

**2.1. Теоретический материал**

Битовый сдвиг — изменение позиций бит в машинном слове.

Большинство компьютеров не могут напрямую адресовать биты, которые содержатся группами по 8, 16, 32 или 64 бита в машинном слове. Для обеспечения работы с битами существует множество машинных инструкций, включающие различные типы сдвигов. Все сдвиги похожи друг на друга поведением средних битов, которые просто сдвигаются влево или вправо на определённую величину. Однако, поведение крайних битов, которые уходят из слова и которые появляются в слове, зависит от типа сдвига.

В электронике битовые сдвиги выполняются на сдвиговых регистрах.

**Логический сдвиг.**

Сдвиг, при котором уходящий бит исчезает, не влияя на оставшиеся биты, а на месте появившегося бита записывается бит 0.

*Пример работы операции сдвига:*

Пусть у нас есть число 10101010b (в двоичной системе).

Если сделать сдвиг влево на 1 бит, то получим число 01010100b.

Если сделать сдвиг исходного числа вправо на 1 бит, то получим число 01010101b.

В большинстве процессоров уходящий бит сохраняется во флаге переноса. Эта функция широко используется при работе со многобайтными числами.

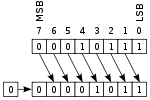


Рис. 1. Логический сдвиг вправо

## 2.2. Разработка ПО

Было принято решение разрабатывать ПО на языке Cи++

Создание функции, отвечающая за сдвиг.

for (j = i - 1; j >= 0; j--) {

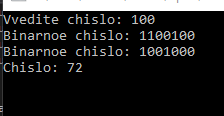
str += arr[j];

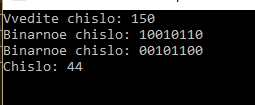
}

str2 = str.substr(1, str.length - 1);

cout << str2;

**3. Тестирование**





# 4. Вывод

В ходе лабораторной работы был повторен теоретический материал по логическим операциям над двоичными числами и повышены навыки программирования.

**5. Исходный код**

#include "stdafx.h"

#include <iostream>

#include <bitset>

#include "conio.h"

using namespace std;

int main() {

int arr[100], i = 0, j;

int n;

cout << "Vvedite chislo: ";

cin >> n;

while (n > 0) {

arr[i] = n % 2;

i++;

n = n / 2;

}

string str = "";

string str2 = "";

cout << "Binarnoe chislo:";

for (j = i - 1; j >= 0; j--) {

str += arr[j];

}

str2 = str.substr(1, str.length - 1);

cout << str2;

cout << "Chislo: ";

\_getch();

return 0;

}