**Белорусский государственный технологический университет**

**Факультет информационных технологий**

**Кафедра программной инженерии**

Лабораторная работа 11

По дисциплине «Основы алгоритмизации и программирования»

На тему «Битовые операции»

Выполнила:

Студентка 1 курса 7 группы

Шинкевич Марина Дмитриевна

Преподаватель: асс. Андронова М.В.

2023, Минск

Дополнительные варианты к лабораторной 11

Вариант 9

1. Извлечь 5 битов числа **A**, начиная с третьего по счету справа, и вставить их в число **B**, начиная со 2 по счету справа.

#include <iostream>

int main()

{

using namespace std;

setlocale(LC\_CTYPE, "Russian");

char tmp[33];// объявляем массив tmp типа char с размером 33 элемента.

int A, B, maskA = 31, maskC = 62, maskB = ~maskC;

cin >> A;//ввод

cout << "Первое число: " << A << endl;//вывод

cin >> B;

cout << "Второе число: " << B << endl;

\_itoa\_s(A, tmp, 2); //преобразуем значение переменной A в двоичное представление и сохраняем результат в массив tmp.

cout << "A= " << tmp << endl;

\_itoa\_s(B, tmp, 2);//преобразуем значение переменной B в двоичное представление и сохраняем результат в массив tmp.

cout << "B= " << tmp << endl;

\_itoa\_s(maskA, tmp, 2);

cout << "Маска для A " << tmp << endl;

\_itoa\_s(maskB, tmp, 2);

cout << "Маска для B " << tmp << endl;

\_itoa\_s((A >> 2 & maskA), tmp, 2);//выполняем операции сдвига переменной A на 2 бита вправо, применяем побитовое И с переменной maskA и преобразуем результат в двоичное представление, сохраняя его в массив tmp.

cout << "Извлеченные биты " << tmp << endl;

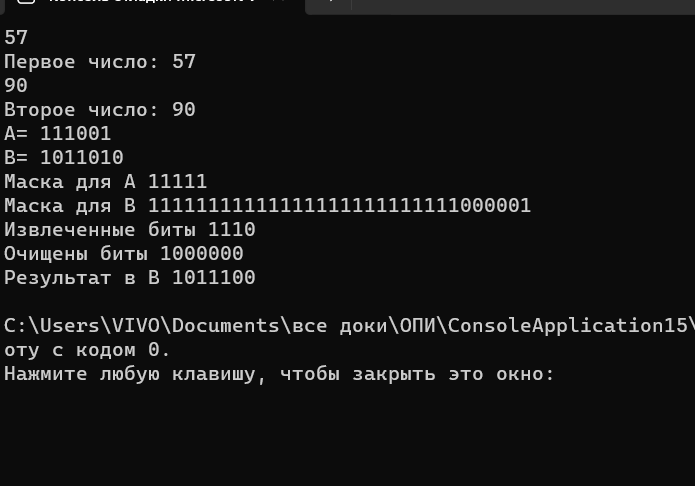
\_itoa\_s((B & maskB), tmp, 2);

cout << "Очищены биты " << tmp << endl;

\_itoa\_s(((A >> 2 & maskA)) << 1 | (B & maskB), tmp, 2);

cout << "Результат в B " << tmp << endl;

}



2.Установить в 0 **n** битов в числе **А** вправо от позиции **p**.

#include <windows.h>

#include <iostream>

int main()

{

using namespace std;

setlocale(LC\_CTYPE, "Russian");

int A, maskA = 1, n, p;

char tmp[33];

cout << "Введите A: " << endl;

cin >> A;

cout << "Введите позицию: " << endl;

cin >> p;

cout << "Введите количество битов: " << endl;

cin >> n;

\_itoa\_s(A, tmp, 2);//преобразуем значение переменной A в двоичное представление и сохраняем результат в массив tmp.

cout << "A= " << tmp << endl;

maskA = ~(maskA << n);// Сдвигает maskA на n позиций влево и выполняет побитовую инверсию(~) результата.

\_itoa\_s(maskA, tmp, 2);//Конвертирует maskA в строку с использованием системы счисления 2 (двоичная) и сохраняет результат в массиве tmp.

cout << "maskA= " << tmp << endl;

maskA = maskA << n;// Сдвигает maskA на n позиций влево и присваивает результат переменной maskA.

\_itoa\_s(maskA, tmp, 2);

cout << "maskA= " << tmp << endl;

maskA = ~(maskA);//Выполняет побитовую инверсию (~) переменной maskA и присваивает результат переменной maskA.

\_itoa\_s(maskA, tmp, 2);

cout << "maskA= " << tmp << endl;

maskA = (maskA << (p - 1));//Сдвигает maskA на (p - 1) позиций влево и присваивает результат переменной maskA.

\_itoa\_s(maskA, tmp, 2);

cout << "maskA= " << tmp << endl;

maskA = ~(maskA);

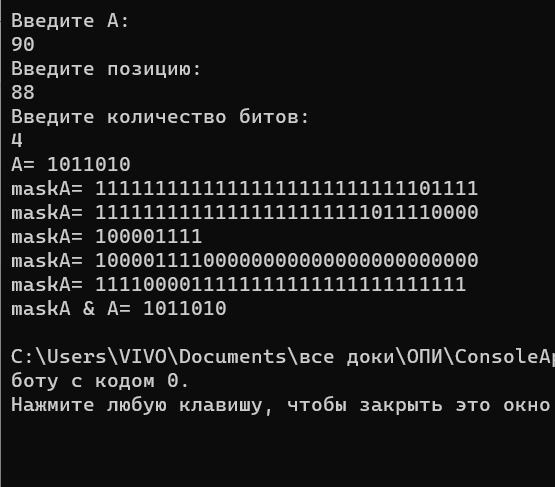
\_itoa\_s(maskA, tmp, 2);

cout << "maskA= " << tmp << endl;

\_itoa\_s(maskA & A, tmp, 2);//Выполняет побитовую операцию И (&) между maskA и A, затем конвертирует результат в строку

cout << "maskA & A= " << tmp << endl;

}



Вариант 2

1. Извлечь 5 битов числа **A**, начиная со второго и вставить их в число **B**, начиная с третьего бита.

#include <iostream>

using namespace std;

int main() {

setlocale(LC\_ALL, "rus");

int A, B;

cout << "Введите число A: ";

cin >> A;

cout << "Введите число B: ";

cin >> B;

int secBtis = (A >> 1) & ((1 << 5) - 1);//Эта строка извлекает 5 битов из числа A, начиная со второго бита.

int clearMask = ~(((1 << 5) - 1) << 2);//Эта строка создает маску для очистки 5 битов в числе B, начиная с третьего бита.

B &= clearMask;// Эта строка применяет маску к числу B, чтобы очистить указанные биты.

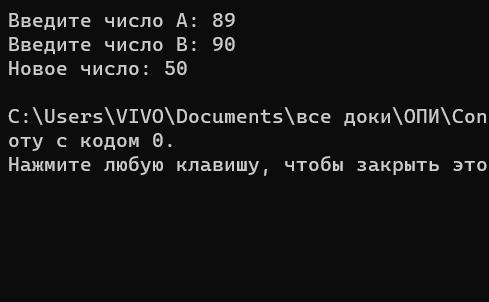
int iBits = secBtis << 2;//Эта строка сдвигает извлеченные биты из числа A на два бита влево.

B |= iBits;

cout << "Новое число: " << B << endl;

return 0;

}



2.Установить в 1в числе **А n** битов вправо от позиции **p**.

#include <iostream>

using namespace std;

int main() {

setlocale(LC\_ALL, "rus");

int A, p, n;

cout << "Введите число A: ";

cin >> A;

cout << "Введите позицию р: ";

cin >> p;

cout << "Введите количество битов n: ";

cin >> n;

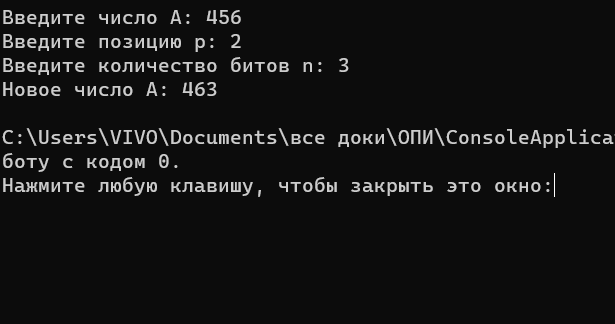
int mask = ((1 << n) - 1) << (p - n + 1);//создать маску для установки n битов в числе A, начиная с позиции p.

A |= mask;//применяет эту маску к числу А

cout << "Новое число A: " << A << endl;

return 0;

}



Вариант 4

1.Используя битовые операции проверить, кратно ли четырем число **А**.

#include <iostream>

using namespace std;

int main() {

setlocale(LC\_ALL, "rus");

int A;

cout << "Введите число А: ";

cin >> A;

if ((A & 3) == 0)//если A делится на 4 без остатка

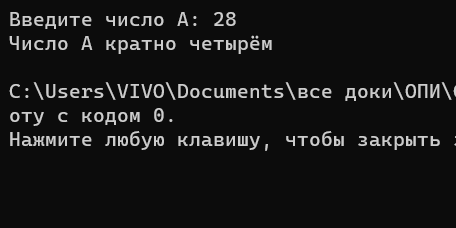
cout << "Число А кратно четырём" << endl;

else

cout << "Число А не кратно четырём" << endl;

return 0;

}



2.Установить в 1 в числе **А n** битов влево от позиции **p**.

#include <iostream>

using namespace std;

int main() {

setlocale(LC\_ALL, "rus");

int A, p, n;

cout << "Введите число А: ";

cin >> A;

cout << "Введите позицию p: ";

cin >> p;

cout << "Введите количество битов n: ";

cin >> n;

int mask = ((1 << n) - 1) << (p - n + 1);//маска для установки n битов влево от позиции p

A |= mask;

cout << "Новое число A: " << A << endl;

return 0;

}

