Белорусский государственный технологический университет

Факультет информационных технологий

Кафедра программной инженерии

 Лабораторная работа 11

По дисциплине «Основы алгоритмизации и программирования»

На тему «Битовые операции»

Выполнил:

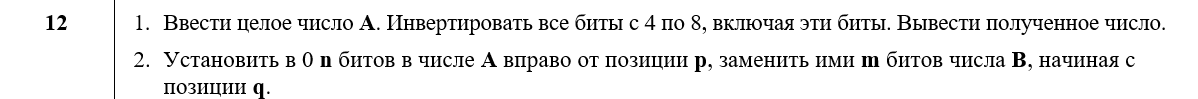
Студент 1 курса 6 группы

Романов Игорь Вячеславович

Преподаватель: асс. Андронова М.В.

2023, Минск

Вариант 12 (основной)



1.

#include <iostream>

using namespace std;

int main() {

setlocale(LC\_CTYPE, "Russian");

unsigned int A, mask = 1 << 7; //ввод числа A и маски для него

cout << "Введите целое число: ";

cin >> A;

unsigned int A\_inverted = ~A, maskB = 1 << 4, A\_copy = A; //ввод вспомогательных переменых A\_inverted и А\_copy и маски для переменной A\_inverted

cout << "Двоичная запись числа А: "; //вывод двоичной записи числа А

for (int i = 1; i <= 8; i++) {

putchar(mask & A ? '1' : '0');

if (i % 2 == 0) {

putchar(' '); //вывод пробела после каждых двух битов

}

A <<= 1; //сдвиг А на один бит влево с каждой итерацией

}

cout << "\nЧисло А после инверсии : "; //вывода битов числа A с 1 по 3

for (int i = 1; i <= 3; i++) {

putchar(mask & A\_copy ? '1' : '0');

if (i % 2 == 0) {

putchar(' '); //вывод пробела после каждых двух битов

}

A\_copy <<= 1; //сдвиг переменной А\_copy на 1 бит влево с каждой итерацией

}

for (int i = 4; i <= 8; i++) { // вывод инвертированных битов с 4 по 8

putchar(maskB & A\_inverted ? '1' : '0');

if (i % 2 == 0) {

putchar(' '); //вывод пробелов после каждых двух битов

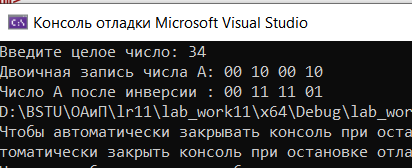
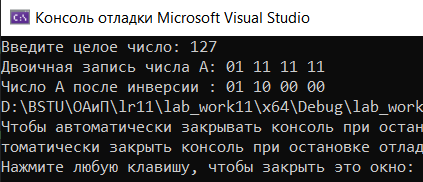
}

A\_inverted <<= 1; //сдвиг A\_inverted на 1 бит влево с каждой итерацией

}

return 0;

}



2.

#include <iostream>

using namespace std;

int main()

{

setlocale(LC\_CTYPE, "Russian");

unsigned int A, mask; //ввод числа А и маски для него

int n, p; //ввод количества обнуляемых битов n и позиции начала обнуления p

cout << "Введите число А:";

cin >> A;

char tmp[33];

\_itoa\_s(A, tmp, 2); //перевод числа A в двоичную систему

cout << "Число A в двоичнном коде:" << tmp << endl; //вывод двоичного представления числа А

cout << "С какой позиции начать обнулять биты?:"; //ввод позиции, с которой будут обнуляться биты

cin >> p;

cout << "Сколько битов обнулить?:"; //ввод количества обнуляемых битов

cin >> n;

mask = (1 << (n + 1)) - 1; //создание маски для числа A

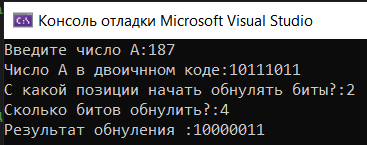
A &= ~(mask << p); //применение маски к числу A

\_itoa\_s(A, tmp, 2); //перевод числа А в двоичную систему

cout << "Результат обнуления :" << tmp << endl; //вывод результата на экран

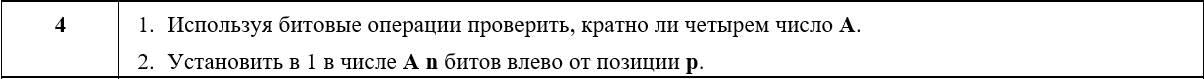
return 0;

}



Доп.задания

Вариант 4



1.

#include <iostream>

using namespace std;

int main()

{

setlocale(LC\_CTYPE, "Russian");

int A; char tmp[33]; //ввод числа A

cout << "Введите число ";

cin >> A;

\_itoa\_s(A, tmp, 2); //перевод числа A в двоичную систему

cout << "Число в двоичном виде = " << tmp << endl; //представление числа A в двоичном виде

if ((A & 3) == 0) //проверка кратности числа на 4 (результат логического умножения на 3 должен быть равен 0)

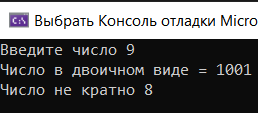
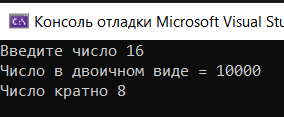
cout << "Число кратно 8" << endl;

else

cout << "Число не кратно 8" << endl;

return 0;

}



2.

#include <iostream>

using namespace std;

int main()

{

setlocale(LC\_CTYPE, "Russian");

unsigned int A, A\_copy, mask, maskA; //ввод числа А, вспомогательной переменной А\_copy, маски для выведения 16-ти битового представления числа А

int n, p; //ввод количествава преобразуемых битов и позиции начала преобразования

cout << "Ввод числа А: ";

cin >> A;

A\_copy = A;

cout << "Двоичная запись числа А: ";

mask = 1 << 15;

for (int i = 1; i <= 16; i++) { //представление числа А в двоичной системе

putchar(mask & A ? '1' : '0');

if (i % 4 == 0) { //вставка пробела после каждого 4-ого символа

putchar(' ');

}

A <<= 1;

}

cout << "\nС какой позиции превратить биты в 1? - ";

cin >> p;

cout << "Сколько битов превратить в 1? - ";

cin >> n;

maskA = ((1 << n) - 1) << (16 - p); //определение маски для преобразования заданных битов

cout << "Результат: "; //вывод результата на экран

A\_copy = A\_copy | maskA; //применение маски к переменной А\_copy

for (int i = 1; i <= 16; i++) { //цикл для вывода результата

putchar(mask & A\_copy ? '1' : '0');

if (i % 4 == 0) { //вставка пробела после каждого 4-ого символа

putchar(' ');//вывод пробела после каждых 4 битов

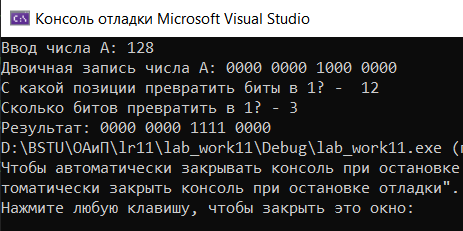
}

A\_copy <<= 1; //сдвиг влево А на 1 каждую итерацию

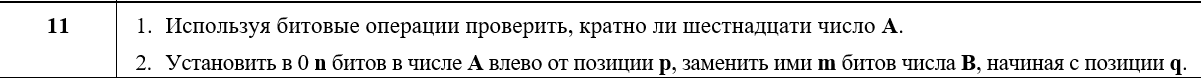
}

return 0;

}



Вариант 11



1.

#include <iostream>

using namespace std;

int main()

{

setlocale(LC\_CTYPE, "Russian");

int A; char tmp[33]; //ввод числа A

cout << "Введите число ";

cin >> A;

\_itoa\_s(A, tmp, 2); //перевод числа A в двоичную систему

cout << "Число в двоичном виде = " << tmp << endl; //представление числа A в двоичном виде

if ((A & 15) == 0) //проверка кратности числа на 16 (результат логического умножения на 15 должен быть равен 0)

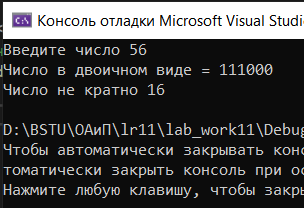
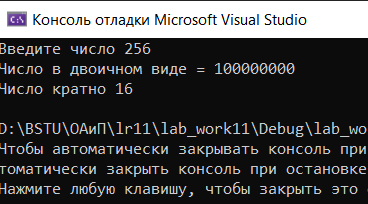
cout << "Число кратно 16" << endl;

else

cout << "Число не кратно 16" << endl;

return 0;

}



2.

#include <iostream>

using namespace std;

void main()

{

setlocale(LC\_CTYPE, "Russian");

unsigned int A, A\_copy, mask, maskA; //ввод числа А, вспомогательной переменной А\_copy и маски ддя числа A

int n, p; //ввод числа обнуляемых битов и позиции начала обнуления

cout << "Введите число А: ";

cin >> A;

A\_copy = A;

mask = 1 << 15; //определение маски для вывода преобразованного числа А

cout << "Двоичное представление числа А: ";

for (int i = 1; i <= 16; i++) { //выводчисла А в двоичной системе

putchar(mask & A ? '1' : '0');

if (i % 4 == 0) { //вставка пробела после каждого 4-ого символа числа A

putchar(' ');

}

A <<= 1; //сдвиг влево А\_copy на 1 каждую итерацию

}

cout << "\nС какой позиции начинать обнулять биты?: ";

cin >> p;

cout << "Сколько битов обнулить?: ";

cin >> n;

maskA = ~((1 << n) - 1 << 16 - p); //задание маски для обнуления битов

A\_copy = A\_copy & maskA; //применение маски к переменной А\_copy

for (int i = 1; i <= 16; i++) { //вывод результата на экран

putchar(mask & A\_copy ? '1' : '0');

if (i % 4 == 0) { //вставка пробела после каждого 4-ого символа числа A

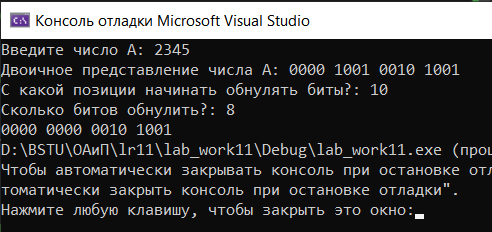
putchar(' ');

}

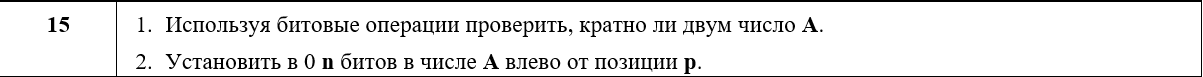
A\_copy <<= 1; //сдвиг влево А\_copy на 1 каждую итерацию

}

}



Вариант 15



1.

#include <iostream>

using namespace std;

int main()

{

setlocale(LC\_CTYPE, "Russian");

int A; char tmp[33]; //ввод числа A

cout << "Введите число ";

cin >> A;

\_itoa\_s(A, tmp, 2); //перевод числа A в двоичную систему

cout << "Число в двоичном виде = " << tmp << endl; //представление числа A в двоичном виде

if ((A & 1) == 0) //проверка кратности числа на 2 (результат логического умножения на 1 должен быть равен 0)

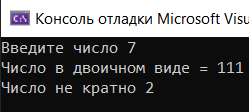
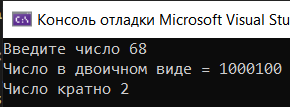
cout << "Число кратно 2" << endl;

else

cout << "Число не кратно 2" << endl;

return 0;

}



2.

#include <iostream>

using namespace std;

int main() {

setlocale(LC\_CTYPE, "Russian");

unsigned int A; // ввод числа A и маски для него

int p, n, mask; // ввод начальной позиции обнуления и количества заменяемых битов

char tmp[33];

cout << "Введите число A: "; cin >> A; // ввод числа A

\_itoa\_s(A, tmp, 2); //перевод числа A в двоичную систему счисления

cout << "Число A в двоичном коде: " << tmp << endl;

cout << "Введите количество битов для обнуления: "; cin >> n;

cout << "Введите начальную позицию: "; cin >> p;

mask = ((1 << n) - 1) << (p - n + 1); //создание маски

A &= ~mask; //применение маски

\_itoa\_s(A, tmp, 2); //перевод числа A в двоичную систему счисления

cout << "Результат обнуления: " << tmp << endl; //вывод результата обнуления

return 0;

}

