

**Министр науки и высшего образования Российской
Федерации**

**Федеральное государственное автономное
образовательное учреждение высшего образования**

**«Национальный исследовательский университет
ИТМО»**

**Факультет информационных технологий и
программирования**

Лабораторная работа №2

Представление чисел в различных системах счисления и битовые операции

Выполнила студентка группы № М31ХХ

Фамилия Имя Отчество

Подпись:

Проверил:

Повышев Владислав Вячеславович

Санкт-Петербург

Решение с комментариями:

1. Подключен заголовочный файл `<stdio.h>`, отвечающий за ввод и вывод данных.
2. Весь код находит в функции `main()`.
3. `int` присваивает `n` целое значение.
4. Строка `printf()` отвечает за вывод значения
5. Строка `scanf(...)` запоминает значения, введенные в консоль. Оператор `'&'` - оператор взятия адреса. Например, `&a` означает адрес переменной `'a'`.
6. Для перевода числа в 8-ричную систему счисления используется `'%o'`.
7. Для перевода числа в 16-ричную систему счисления используется `'%X'`.
8. Чтобы сдвинуть двоичный код справа на определенное количество бита, можно использовать оператор `'>>'`, а если наоборот, то `'<<'`
Например, число 10-ричной системы счисления 123, а количество сдвига 3.
Сначала нужно перевести из 10СС в 2СС, выполнить оператор, а затем перевести обратно в 10СС.
`a << 3`: Сдвиг влево на 3 позиции: $123 = 1111011$.
После сдвига влево на 3 позиции получим: $1011000 = 88$.
`a >> 3`: Сдвиг вправо на 3 позиции: $123 = 1111011$.
После сдвига вправо на 3 позиции получим: $1111 = 15$.
9. Для отрицания числа используется оператор `'~'`.
Оператора `'~'` выполняет побитовое инвертирование числа.
Нужно перевести число из 10-ричной СС в двоичную СС, а затем выполняется оператор отрицания. Число может выйти отрицательным.
Пусть число будет 123
 $123 = 1111011$
 $\sim a = -1111100 = -124$.
10. Для исключающие или используется оператора `'^'`.
Оператор `'^'` выполняет побитовую операцию, где каждый бит результата устанавливается в 1, если соответствующие биты разные.
Пусть `a = 5`, `b = 3`, переведем их в двоичную систему счисления.
`a = 110`, `b = 11`
 $a \wedge b = 110 \wedge 011 = 101$
Числа 101 (2СС) в 10-ричной системе счисления будет 6.

```

#include <stdio.h>

int main() {
    int n;
    // 10CC
    printf("1. Введите число в 10CC: ");
    scanf("%d", &n);
    printf("\n");
    // 8CC
    printf("2. Число в 8CC: ");
    printf("%o\n\n", n);
    // 16CC
    printf("3. Число в 16CC: ");
    printf("%X\n", n);
    // 16CC сдвиг на 4 бита
    printf("Число, сдвинутое вправо на 4 бита\n");
    printf("Результат: ");
    printf("%X\n\n", n >> 4);
    // 16CC
    printf("4. Отрицание\n");
    printf("16CC: ");
    printf("%X\n", n);
    printf("Результат: ");
    printf("%X\n\n", ~n);
    // XOR
    printf("5. XOR\n");
    int a;
    printf("Введите любое число в 16CC: ");
    scanf("%X", &a);
    printf("Результат XOR: %X\n", a ^ n);
    return 0;
}

```

STDIN:

123
7B

STDOUT:

1. Введите число в 10CC: 123

2. Число в 8CC: 173

3. Число в 16CC: 7B
Число, сдвинутое вправо на 4 бита
Результат: 7

4. Отрицание
16CC: 7B
Результат: FFFFFFFF84

5. XOR
Введите любое число в 16CC: 7B
Результат XOR: 0