**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

**Кафедра ИС**

отчет

**по лабораторной работе №1**

**по дисциплине «Программирование»**

Тема: "Одномерные статические массивы"

|  |  |  |
| --- | --- | --- |
| Студент гр. 4372 |  | Климаш И.В |
| Преподаватель |  | Глущенко A.Г |

Санкт-Петербург

2024

**Цель работы.**

Цель данной лабораторной работы — изучить принципы работы с битовыми операциями, научиться изменять и анализировать битовые представления чисел различных типов данных (int, float, double), а также узнать, как данные хранятся в памяти компьютера.

**Основные теоретические положения.**

В данной лабораторной работе исследуются основные принципы работы с битами, такие как установка и сброс битов с использованием побитовых операций. Битовые операции используются для оптимизации работы с данными, особенно когда необходимо работать с отдельными битами, а не с целыми числами.

Основные побитовые операции:

* И (&): возвращает 1, если оба операнда содержат 1.
* ИЛИ (|): возвращает 1, если хотя бы один из операндов содержит 1.
* НЕ (~): инвертирует все биты операнда.
* Сдвиг влево/вправо (<<, >>): сдвигает все биты числа влево или вправо на указанное количество позиций.

При работе с числами типа float и double используется союз union, чтобы получить доступ к битовому представлению числа, не изменяя его сам тип.

Экспериментальные результаты:

1. **Размеры типов данных**  
   В первом задании определены размеры различных типов данных в байтах:
   * int: 4 байта
   * short int: 2 байта
   * long int: 8 байт
   * float: 4 байта
   * double: 8 байт
   * long double: 16 байт
   * char: 1 байт
   * bool: 1 байт
2. **Битовое представление целого числа**  
   Введено число типа int. Программа выводит его двоичное представление и позволяет изменить определённый бит. Например:
   * Введенное число: 5
   * Двоичное представление: 00000000 00000000 00000000 00000101
   * После изменения бита 0 на значение 0, число изменилось на 4:
   * Новое двоичное представление: 00000000 00000000 00000000 00000100
3. **Битовые операции над числом типа float**

С помощью союза union программа преобразует float в целое число, чтобы вывести его двоичное представление. После изменения определённого бита, можно наблюдать, как меняется исходное число:

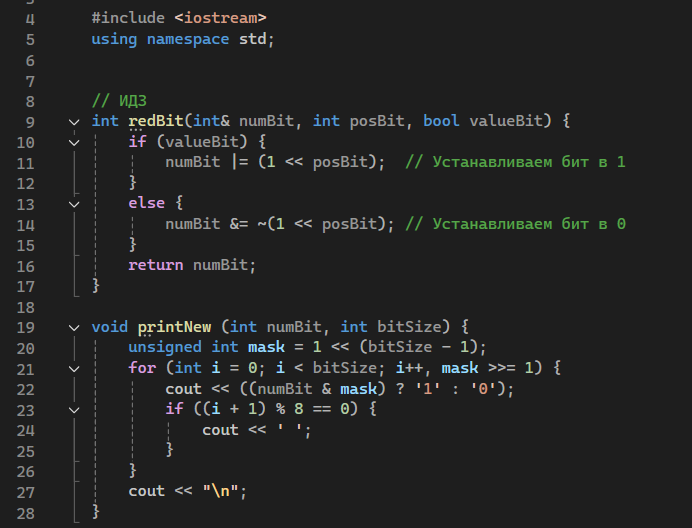
* + Введенное число: 5.5
  + Двоичное представление: 01000001 10110000 00000000 00000000
  + После изменения одного из битов значение изменилось.

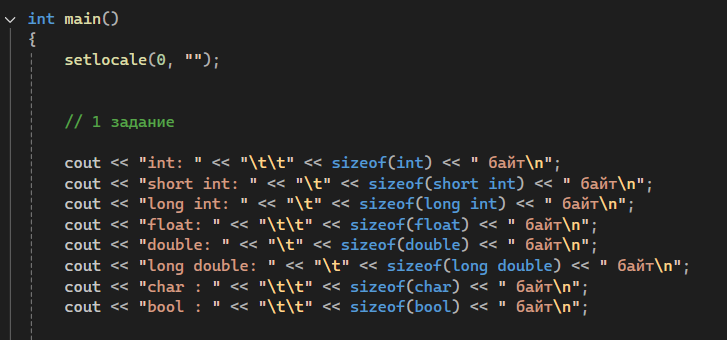
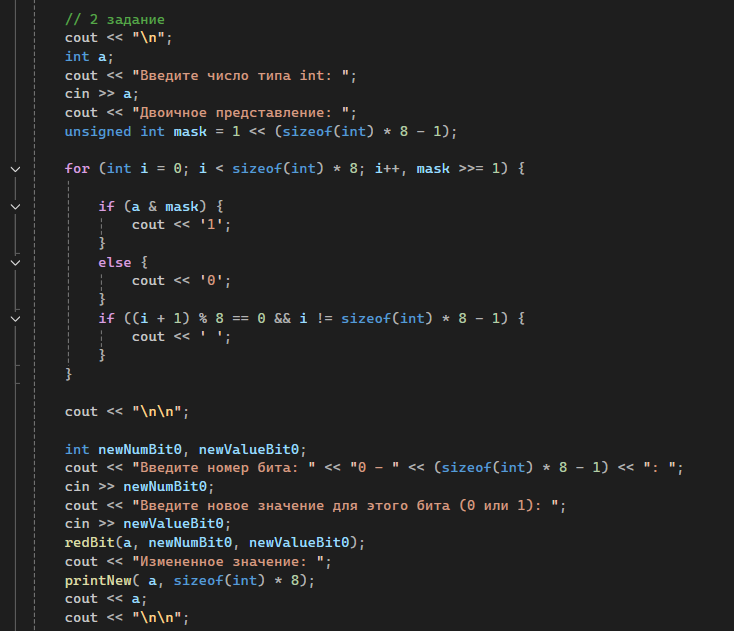
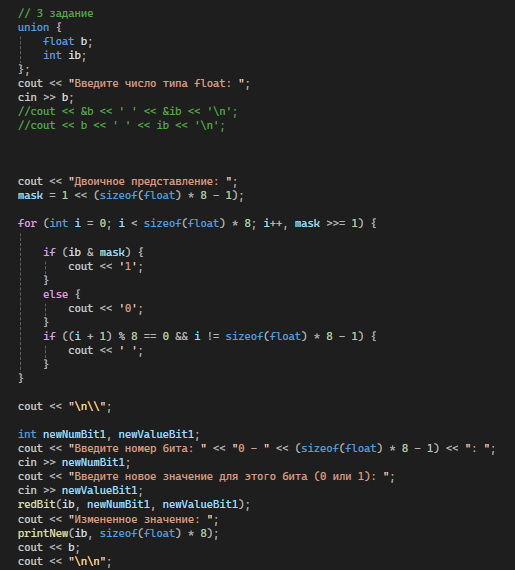
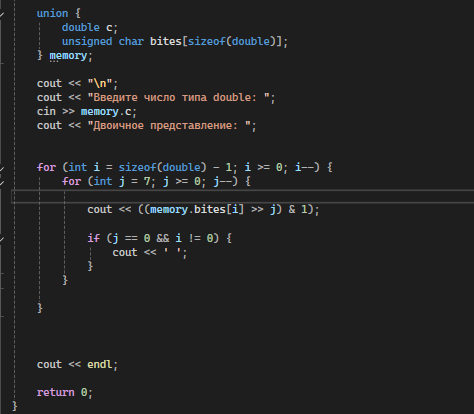
1. **Представление числа типа double в памяти:**

Программа выводит двоичное представление числа типа double, используя массив байт. Пример для числа 10.25:

* + Двоичное представление: 01000000 00101001 00000000 00000000 00000000 00000000 00000000 00000000

**Экспериментальные резултаты.**



**Обработка результатов эксперимента**

Для каждого типа данных (int, float, double) были выведены их двоичные представления, а также была предоставлена возможность изменять отдельные биты. Изменения были отражены как в двоичных, так и в десятичных представлениях данных.

**Выводы**

Работа с битовыми операциями позволяет эффективно управлять отдельными битами числа, что важно при оптимизации программ и работе с низкоуровневыми задачами. Полученные результаты полностью соответствуют теоретическим положениям и показали, как изменения отдельных битов могут повлиять на значение переменных.

**Протокол**

1. Запустить Visual Studio.
2. Написать код, исходя из данных заданий.
3. Написать комментарии и проверить на корректность код.
4. Сохранить код и загрузить на github.