**МИНОБРНАУКИ РОССИИ**

**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**

**ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

**"ЛЭТИ" ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**

**Кафедра информационных систем и технологий**

**ОТЧЁТ**

**по практической работе №1**

**по дисциплине "Программирование"**

**Тема:** **типы данных в компьютерной памяти**

Студент гр. 0324 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Гордиенко Т.Е.

Преподаватель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Глущенко А.Г.

Санкт-Петербург

2020

**Цель работы**: разработать алгоритм и написать программу, которая позволяет:

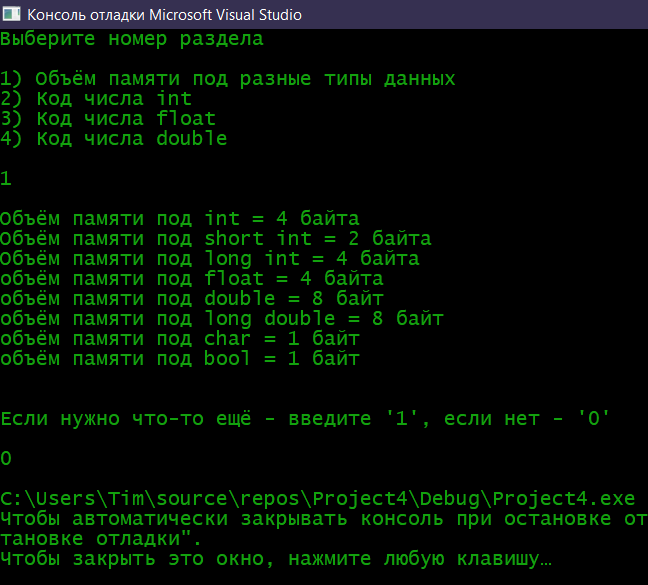
1)Вывести, сколько памяти (в байтах) на вашем компьютере отводится под различные типы данных со спецификациями и без.

2)Вывести на экран двоичное представление в памяти (все разряды) целого числа.

3)Вывести на экран двоичное представление в памяти (все разряды) типа float.

4)Вывести на экран двоичное представление в памяти (все разряды) типа double.

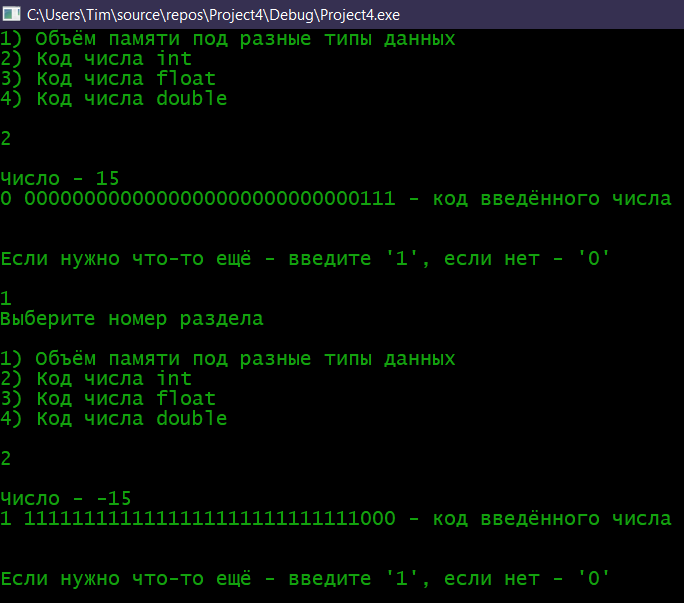
**Выполнение работы.**

 Для каждой задачи я сделал отдельную функцию. Функции будут вызваны только тогда, когда пользователь выберет определённый раздел. После завершения работы, над выбранным разделом, у пользователя будет выбор - либо закончить работу программы, либо продолжить с тем же или иным разделом, как показано на скрин.1. В данном случае, пользователем был выбран 1-й раздел, и программа вывела объём памяти, выделяемый компьютером под разные типы данных и предложила продолжить работу либо закончить. После чего, пользователем было выбрано завершение работы.

Скрин.1

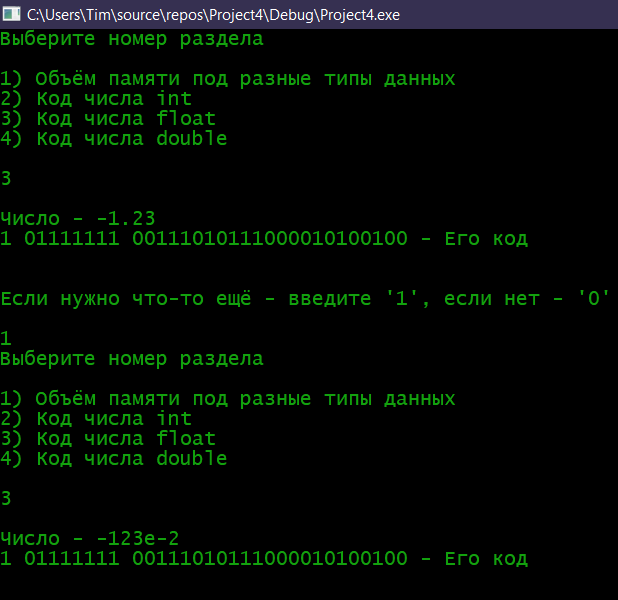
1) Для вывода размера разных типов данных, я воспользовался оператором "sizeof()", который возвращает цело число байт, занимаемых определённым элементом или типом данных.

2)Для определения кода целого числа, была использована поразрядная конъюнкция(&). В качестве маски мы будем использовать переменную "mask" равную минимально возможному int (т.к. его двоичный код будет состоять из 1 и 31/15 нулей в зависимости от процессора). Введённое число записывается в переменную "num". Т.к. первый бит отвечает за знак числа, он будет отделён пробелом. После работы над каждым битом введённого числа, сдвигаем его на 1 разряд влево. Так на скрин.2 показан результат работы для числа 15 и -15.

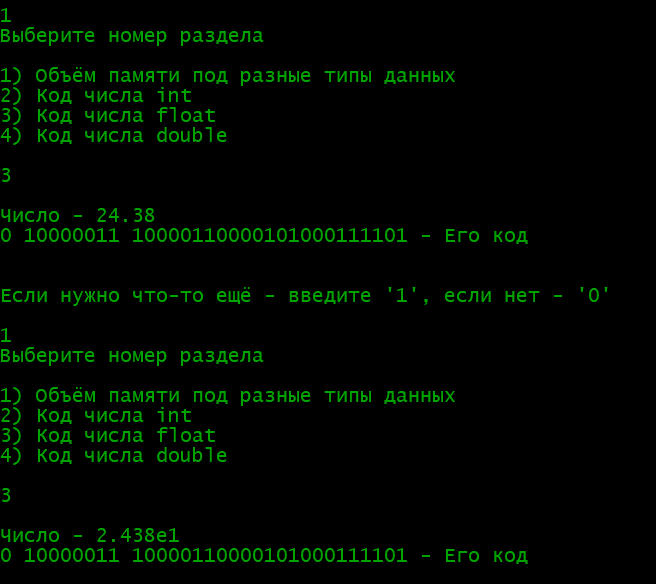


Скрин.2

3)Для float воспользовались тем же способом, что и для int, только с небольшим изменением. Т.к. поразрядная конъюнкция не применима к float, было создано объединение из float и int. Введённое число будет записано в переменную "num". А т.к. num объединено с tool (тип long int). то в tool будет "помещён" тот же код, что и в tool без потерь, т.к. под float и long int отводится одинаковый объём памяти. Так на Скрин.3 и Скрин.4 показан результат работы для -1.23; -123e-2; 24.38; 2.438e1.

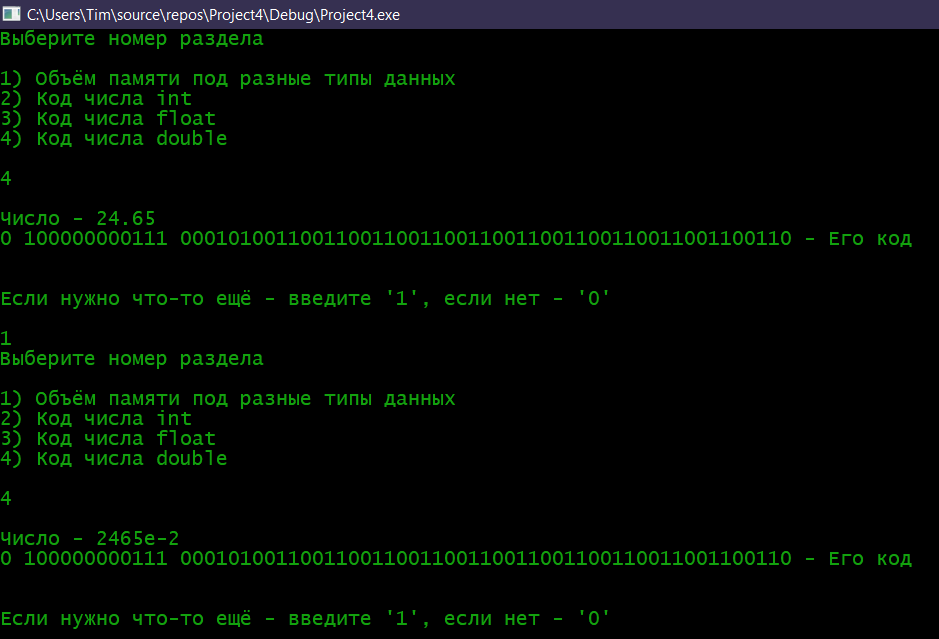


Скрин.3



Скрин.4

4) Для double поступаем таким же образом как для float, только вместо long int tool будем использовать long int tool[2] (массив из двух long int) т.к. double занимает 8 байт, а long int - 4. Так на Скрин.5 приведён результат работы для 24.65 и 2465е-2.



Скрин.5