Министерство образования Республики Беларусь

Учреждение образования

«Белорусский государственный университет

информатики и радиоэлектроники»

ФАКУЛЬТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И УПРАВЛЕНИЯ

Кафедра интеллектуальных информационных технологий

**Отчёт**

По дисциплине: Аппаратные основы интеллектуальных систем

На тему: Представление числовой информации и выполнение в ЭВМ арифметических операций над числами, представленными в разных формах и кодах.

Выполнил: Локтев К.А.

Группа 021702

Проверил: Захаров В.В.

Минск 2021

**Цель работы:**

Повторение и закрепление материала по представлению числовой информации в ЭВМ (системам счисления, формам представления и видам кодирования чисел), освоение навыков по правилам выполнения арифметических операций над числами, представленными в разных формах и кодах.

**Задача:**

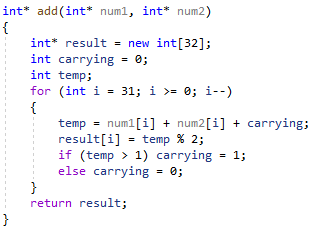
Составить и проверить программу, обеспечивающую выполнение следующих задач:

1. Перевод из десятичной системы счисления в двоичную и выполнение сложения/вычитания чисел 13 и 23 в прямом, дополнительном и обратном кодах всех вариантов слагаемых (+/+, +/-, -/+, -/-)
2. Выполнить умножение модулей чисел 13 и 23, определить знаки произведения для всех вариантов знаков сомножителей
3. Выполнить деление модуля числа 13 на модуль числа 23. Результат округлить до 5 разрядов. Определить знаки частного для всех вариантов знаков делимого и делителя.
4. Выполнить сложение чисел 13 и 23, представленных в форме с плавающей точкой.

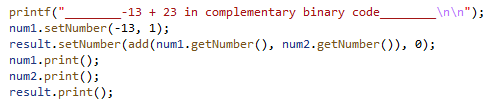
**Ход работы:**Перевод из десятичной системы счисления в двоичный прямой код реализован функцией, использующей стандартный подход с повторяющимся делением на 2 и возвращающей в обратном порядке все получившиеся остатки от деления.

Описан класс двоичных чисел, содержащий в себе поля для знака (0/1), кода (0/1/2 – прямой, дополнительный и обратный соответственно) и для собственно числа в двоичном виде. Также класс содержит метод для изменения хранящегося в нём числа, который принимает в качестве аргументов код числа (0/1/2) и либо число в десятичном виде, либо число в двоичном виде. Такой подход реализован в целях более удобной работы с классом, т.е. нет необходимости описывать функцию для перевода из двоичной СИ в десятичную. Перевод числа в код, соответствующий переданному в функцию числу от 0 до 2, производится этим же методом. Также класс содержит метод для побитового сдвига влево – это реализовано для более понятного и компактного кода функций, осуществляющих умножение и деление.

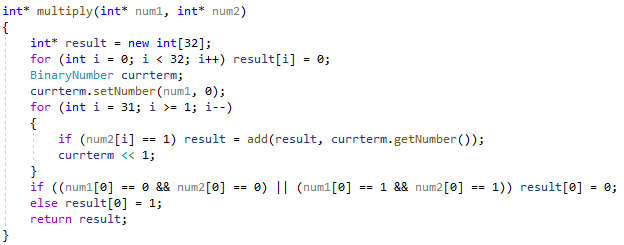
Функция, реализующая сложение двоичных чисел, использует стандартный подход с поразрядным сложением и переносом («запоминанием») перехода через разряд.



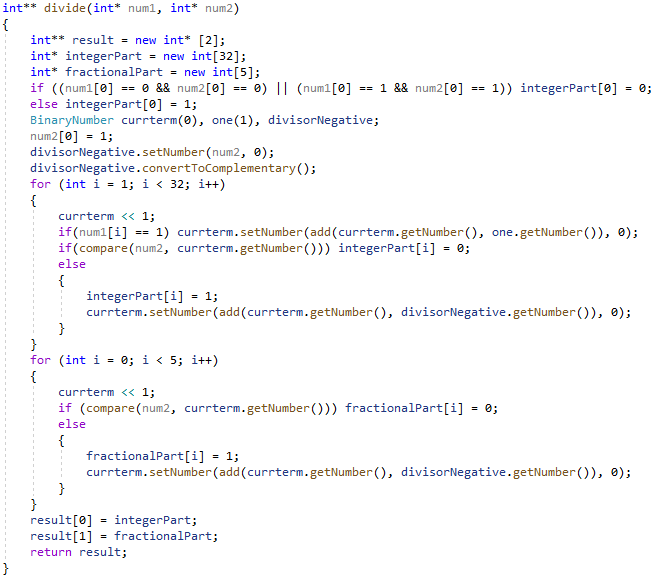
В случае, если одно или оба слагаемых отрицательны, перед вызовом функции сложения оно (они) переводятся в обратный код.



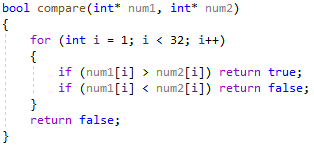
Функция, реализующая умножение двоичных чисел, реализована через стандартный подход с повторяющимся сложением и побитовым сдвигом второго множителя на каждой итерации. После окончания повторяющегося сложения знак результата определяется, как 0, если знаки множителей одинаковы, и 1, если они различны.



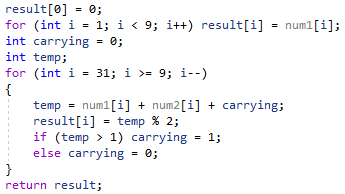
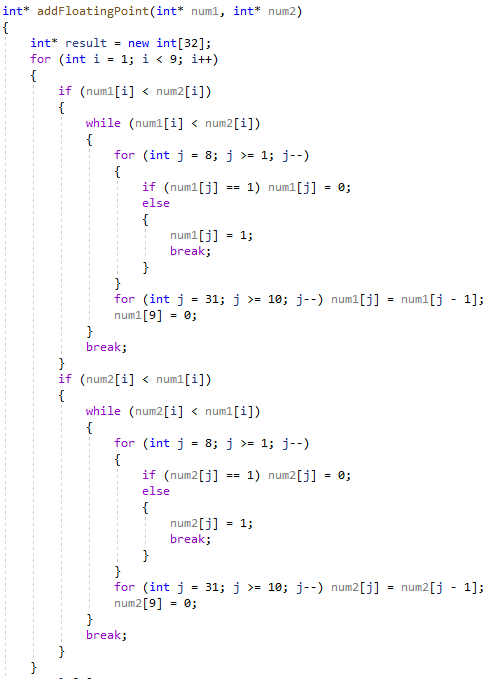
Функция, реализующая деление двоичных чисел, работает следующим образом. Для начала определяется знак частного – 0, если знаки делимого и делителя одинаковы, и 1, если они различны. Затем объявляется двоичное число, в которое поразрядно будет записываться делимое. В частное будут записываться нули до тех пор, пока текущее значение этого числа не будет превосходить делителя. Как только оно превзойдёт делитель, в текущий разряд частного запишется 1, и из этого числа будет вычтен делитель. Процесс повторяется до последнего разряда делимого. Полностью аналогичный процесс происходит и для дробной части частного.



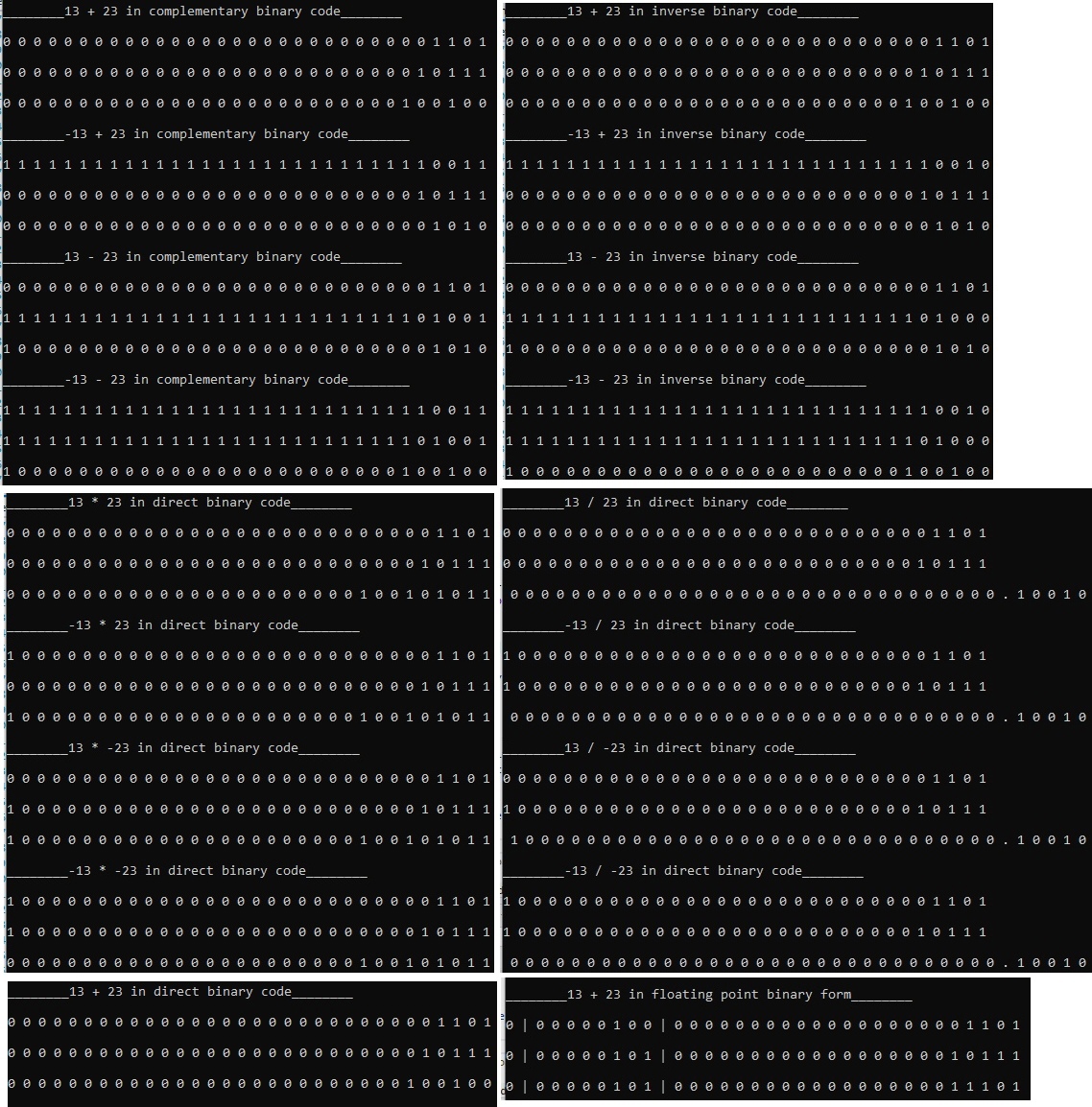
Для работы функции деления была введена вспомогательная функция, сравнивающая модули двух двоичных чисел. Эта функция поразрядно сравнивает двоичные числа и по сути своей является функцией «больше».



Функция, выполняющая сложение двоичных чисел в форме с плавающей точкой, изначально приводит числа к одному порядку, а затем складывает, используя стандартный алгоритм для сложения двоичных чисел.



Программой предусмотрена работа с произвольными двоичными числами, выполнение индивидуального задания предусмотрено как отдельный пункт меню. После выбора этого пункта программа выводит в консоль следующие значения:



**Выводы:**

На аппаратном уровне ЭВМ используют двоичную (бинарную) систему счисления – позиционную систему счисления с основанием 2. Данный выбор обусловлен непосредственным представление двоичной системы на электронных схемах (состояние 1 или 0 на транзисторе, лампе, реле и т.д.). На более высоких уровнях числа из двоичной системы могут быть переведены в любую другую и обратно.

Для хранения и представления двоичных чисел в памяти ЭВМ существуют разрядные сетки различных размерностей. Число всех возможных хранимых чисел определяется по формуле , где n – разрядность сетки. Для представления знака числа вводится дополнительный разряд. В общем случае, если его значение – 1, то число отрицательное, если 0 – положительное. Следует принимать во внимание, что разрядная сетка может переполнится. В таком случае применяются модифицированные коды – коды, в которых под знак числа отводится два разряда. Это позволяет легче идентифицировать переполнение сетки.

Для кодирования двоичных чисел применяются прямой, обратный и дополнительный коды. Прямой код удобен для выполнения операций с неотрицательными числами. Для проведения операций с отрицательными числами в прямом коде необходимо использовать дополнительные алгоритмы, что влияет на скорость вычисления и требует дополнительных ресурсов. Для операций с отрицательными числами удобно использовать обратный и дополнительный коды. Они позволяют свести операцию вычитания к операции сложения. Разница между обратным и дополнительными кодами заключается в том, что для представления нуля в обратном коде, как и в прямом, существует два варианта: т.н. отрицательный и положительный ноль (различие в знаковом разряде). Для положительных чисел прямой, обратный и дополнительный коды совпадают.

Все арифметические операции на аппаратном уровне можно свести к операции сложения и сдвига. Данное свойство позволяет значительно упростить схемы ЭВМ.