**Классы**

Main – создается два объекта типа BinaryNumber - верхний и нижний.

Проверка на то,что верхнее число больше нижнего.

Запуск алгоритма расчета.

BinaryNumber – содержит число в двоичной и десятичной СС

Имеет static счетчик преобразований.

Содержат static методы с алгоритмом расчета.

**Алгоритм расчета**

За основу взято утверждение, что кратчайший путь преобразований лежит через максимально возможное количество преобразований n\*2. Оставшаяся разница должна быть покрыта преобразованиями n+2 и n+1 вставленные между умножениями. Работа алгоритма выполняется в двоичной СС. Переход к двоичной СС обусловлен видом операции n\*2 – он смещает число влево и добавляет ноль. Т.о. алгоритм делится на 2 части :

1. Привести нижнее число к части верхнего числа( часть длиной с нижнее ). Если оказывается, что часть верхнего числа меньше меньшего числа, то сравниваем с частью верхнего числа, увеличенного на один символ. Приведение осуществляется операциями n+2 и n+1 ( только для соблюдения кратности ). Это приведение быстрее осуществить в десятичной СС. Этот этап позволяет повторить начальную часть верхнего числа.
2. Цикл - Теперь мы каждый шаг умножаем нижнее число на 2. В конце нижнего числа появляется новый ноль, который мы сравниваем с цифрой под этим же порядковым номером у верхнего числа. Если ноль, мы ничего не делаем , если единица – осуществляем преобразование n+1. Т.о. мы постепенно повторяем бинарный код верхнего числа. Граница цикла – длина нижнего числа не больше длины верхнего, т.е. они становятся равны.

**Ввод , вывод**

Ввод чисел осуществляется через консоль. Есть проверка для верного соотношения большего и меньшего чисел(большее > меньшее). “Защита от дураков” не предусмотрена.

Вывод осуществляется так же через консоль. Выводится общее количество операций после первого этапа и полное количество операций после второго.