Лабораторная работа №1

Системы счисления

Теоретическая часть

В повседневной практике для представления чисел люди пользуются почти исключительно десятичной системой счисления. Лишь в редких случаях встречаются остатки других систем - римский счет, двенадцатиричная система (часы), шестидесятиричная (минуты).

Однако система изображения чисел, которая веками складывалась применительно к ручному труду, не позволяет получить наиболее эффективные методы выполнения вычислений. По этой причине в вычислительной технике применяются другие системы счисления и чаще всего - двоичная.

Введем несколько определений.

Система счисления - совокупность символов и правил для обозначения чисел.

Разделяют системы счисления позиционные и непозиционные. Непозиционная система счисления задается перечислением изображаемых в ней значений. Позиционная система счисления характеризуется основанием и тем, что числа, как правило, представляются несколькими разрядами (являются многоразрядными), а вес любого разряда определяется его позицией в числе.

Основание позиционной системы счисления определяет количество различных цифр (символов), допустимое в системе счисления. Это же число определяет, во сколько раз вес цифры данного разряда меньше веса цифры соседнего старшего разряда.

Так, в десятичной системе счисления, основание которой равно 10, различают 10 арабских цифр - 0, 1, 2, ..., 9. Следовательно, при ее использовании для записи числа, не превышающего девяти, достаточно одной цифры, и такое число записывается как одноразрядное. А в случае записи числа, большего девяти, оно представляется как многоразрядное. При этом вес каждого более старшего (расположенного слева от текущего) разряда в десять (основание системы счисления) раз больше текущего.

Так, например, число 359 - трехразрядное, и в нем 9 - цифра разряда единиц, 5 - цифра разряда десятков, 3 - цифра разряда сотен (в 10 раз превышает вес разряда десятков). При этом значение трехразрядного числа 359 получается суммированием трех слагаемых : 3 сотни + 5 десятков + 9 единиц.

Так, вес самого младшего разряда целых чисел равен 1, поскольку номер разряда равен 0, а любое число, в том числе и число 10, возведенное в нулевую степень, дает в результате единицу. Вес следующего слева разряда равен 10 в степени 1, т.е. равен десяти, и т.д.

Это же правило справедливо и для записи дробных чисел. При этом разрядам справа от разряда единиц, имеющего номер 0, присваиваются отрицательные значения: -1, -2, и т.д., а их веса получаются также при возведении основания 10 в соответствующую степень. Так, например, вес третьего разряда в дробной части числа 42,9724 будет равен 10 в степени (-3), т.е. равен одной тысячной.

Указанное правило можно проиллюстрировать следующим образом:

Число	7	5	0	6	8	2	5	9
Номер разряда	4	3	2	1	0	-1	-2	-3
Вес разряда	10000	1000	100	10	1	0,1	0,01	0,001

Как видно из примера, в позиционной системе счисления достаточно знать значение основания системы счисления, символы, изображающие отдельные цифры, и указанное правило, чтобы представить любое число.

В вычислительной технике широко применяют двоичную, восьмеричную и шестнадцатиричную систему счисления.

Двоичная система счисления имеет основание 2, и, следовательно, две разных цифры - 0 и 1; восьмеричная - восемь разных цифр - 0, 1, 2, 3, 4, 5, 6, 7, а шестнадцатиричная - шестнадцать цифр - десять арабских цифр от 0 до 9 и еще шесть символов -

A (цифра, изображающая десять), D (цифра тринадцать),

B (цифра одиннадцать), E (цифра четырнадцать),

C (цифра двенадцать), F (цифра пятнадцать).

Проще всего сопоставить запись одних и тех же чисел в этих системах счисления можно с использованием таблицы 3, приведенной на следующей странице.

Мы уже говорили о том, что современные цифровые ЭВМ все используют в качестве основной двоичную систему счисления. К ее достоинствам относится:

- простота выполнения арифметических и логических операций, что влечет за собой простоту устройств, реализующих эти операции;
- возможность использования аппарата алгебры логики (булевой алгебры) для анализа и синтеза операционных устройств ЭВМ.

К неудобствам двоичной системы счисления относится необходимость перевода чисел из десятичной в двоичную и наоборот, а также то, что запись числа в двоичной системе громоздка (требует большего числа разрядов, чем привычная для человека десятичная). По этой и ряду других причин, кроме двоичной применяются восьмеричная и шестнадцатиричная системы счисления.

Таблица 3.1 Система счисления

10	**2**	**8**	**16**
0	0	0	0
1	1	1	1
2	0 1	2	2
3	1 1	3	3
4	1 0 0	4	4
5	1 0 1	5	5
6	1 1 0	6	6
7	1 1 1	7	7
8	1 0 0 0	1 0	8
9	1 0 0 1	1 1	9
10	1 0 1 0	1 2	A
11	1 0 1 1	1 3	B
12	1 1 0 0	1 4	C
13	1 1 0 1	1 5	D
14	1 1 1 0	1 6	E
15	1 1 1 1	1 7	F
16	1 0 0 0 0	2 0	1 0

Совместное использование указанных систем обусловлено двумя причинами:

- в восьмеричной и шестнадцатиричной системах любое число записывается более компактно, нежели двоичное;
- простотой преобразования из двоичной в восьмеричную (шестнадцатиричную) систему счисления и наоборот.

Приведем правила перевода чисел из двоичной системы в восьмеричную (шестнадцатиричную) и наоборот.

3.2. Правило перевода “8с/с → 2с/с”

При переводе многоразрядного числа каждую цифру исходного восьмеричного числа представить всегда точно тремя двоичными цифрами, взятыми из приведенной выше таблицы. При этом, если для записи соответствующей восьмеричной цифры в виде двоичной требуется менее трех двоичных цифр, двоичный эквивалент дополняется слева нулями (незначащие нули не исказят значения числа). Таким образом, например, при записи четырехразрядного восьмеричного числа должно получиться двенадцатиразрядное двоичное. После окончания такого преобразования можно отбросить старшие для всего числа незначащие двоичные цифры.

Отметим, что три цифры принято называть ***триадой***. Поэтому можно сказать, что при описываемом переводе каждая восьмеричная цифра заменяется соответствующей ей триадой двоичных цифр.

Если исходное число дробное, т.е. имеет целую и дробную часть, то в двоичном числе запятая ставится между триадами, представляющими соответствующие цифры исходного числа.

Пример.

Преобразуем восьмеричное число 371,62.

Для этого запишем для каждой цифры соответствующую триаду:

3 → 011

7 → 111

1 → 001

6 → 110

2 → 010

Теперь можно записать число в двоичной форме (для наглядности между триадами поместим пробелы):

371,62 → 011 111 001 , 110 010

И, наконец, запишем полученное двоичное число так, как это принято в математике, без незначащих нулей, а также отбросив правые нули в дробной части числа:

371,62 → 11111001,11001

3.3. Правило перевода “2с/с →8с/с”

При переводе многоразрядного двоичного числа в восьмеричную форму поступают следующим образом: Исходное число разбивают на триады. При этом для целой части числа разбиение проводят от местонахождения запятой влево, а для дробной части - от этого же места вправо. Затем самая левая группа при необходимости дополняется незначащими нулями до образования триады, а самая правая группа только в дробной части дополняется нулями справа также до образования полной триады. После этого каждая триада заменяется соответствующей восьмеричной цифрой. Местоположение запятой сохраняется по тем же правилам, что и в правиле П1.

Пример.

Представить двоичное число 1101100,01111101 в форме восьмеричного.

Разобьем исходное число на группы по три цифры, приняв в качестве точки отсчета местоположение запятой (для наглядности между триадами поместим пробелы):

1 101 100 , 011 111 01

Теперь дополним до трех цифр нулями самую левую группу слева и самую правую группу справа:

001 101 100 , 011 111 010

И, наконец, заменим каждую триаду соответствующей восьмеричной цифрой:

001 101 100 , 011 111 100 → 154,372

3.4. Правило перевода “16с/с→2с/с”

При переводе многоразрядного шестнадцатиричного числа в двоичную форму каждую цифру исходного числа заменяют группой точно из четырех двоичных цифр (заменяют ***тетрадой*** двоичных цифр). Местоположение запятой сохраняется по тем же правилам, что и в правиле П1. В окончательной записи можно отбросить самые левые (незначащие) нули и самые правые нули дробной части.

Пример. Преобразовать шестнадцатиричное число “6С,7D” в двоичную форму.

Для этого запишем для каждой цифры соответствующую тетраду:

6 → 0110

C → 1100

7 → 0111

D → 1101

Теперь можно записать число в двоичной форме (для наглядности между тетрадами поместим пробелы):

6C,7D → 0110 1100 , 0111 1101

И, наконец, запишем полученное двоичное число так, как это принято в математике, без незначащих нулей:

6C,7D → 1101100,01111101

3.5. Правило перевода “2с/с →16с/с”

При переводе многоразрядного двоичного числа в шестнадцатиричную форму поступают следующим образом. Исходное число разбивают на тетрады. При этом для целой части числа разбиение проводят от местонахождения запятой влево, а для дробной части от этого же места вправо. Затем самая левая группа при необходимости дополняется незначащими нулями до образования тетрады, а самая правая группа только в дробной части дополняется нулями справа также до образования полной тетрады. После этого каждая тетрада заменяется соответствующей шестнадцатиричной цифрой. Местоположение запятой сохраняется по тем же правилам, что и в правиле П1.

Пример. Представить двоичное число 1101100,01111101 в форме шестнадцатиричного.

Разобьем исходное число на группы по четыре цифры, приняв в качестве точки отсчета местоположение запятой (для наглядности между тетрадами поместим пробелы):

110 1100 , 0111 1101

Теперь дополним до четырех цифр нулями слева самую левую группу:

0110 1100 , 0111 1101

И, наконец, заменим каждую тетраду соответствующей шестнадцатиричной цифрой:

0110 1100 , 0111 1101 -> 6C,7D.

Шестнадцатиричная и восьмеричная системы счисления используются для более компактной и удобной записи двоичных чисел.

Так, известность шестнадцатиричной системе принесло то, что с ее использованием удобно представлять программы в кодах большинства современных ЭВМ.

3.6. Перевод чисел из одной системы счисления в другую

Поскольку в практической деятельности люди привыкли оперировать десятичной системой счисления, а в ЭВМ числа представляются в двоичной, необходимо научиться преобразовывать числа из одной системы счисления в другую. Рассмотренные выше правила перевода из двоичной системы счисления в восьмеричную и шестнадцатиричную и наоборот носят частный характер и не могут быть распространены на другие системы. Здесь же мы рассмотрим общие правила перевода, справедливые для любой пары систем счисления, хотя и более громоздкие и трудоемкие по сравнению с рассмотренными выше.

Правила перевода целых и дробных чисел не совпадают, поэтому приведем три правила перевода чисел из системы счисления с основанием R в систему счисления с основанием Q.

3.7. Перевод целых чисел

Для перевода целого числа N, представленного в системе счисления (с/с) с основанием R, в с/с с основанием Q необходимо данное число делить на основание Q по правилам с/с с основанием R до получения целого остатка, меньшего Q. Полученное частное снова необходимо делить на основание Q до получения нового целого остатка, меньшего Q, и т.д., до тех пор, пока последнее частное будет меньше Q. Число N в с/с с основанием Q представится в виде не упорядоченной последовательности остатков деления в порядке, обратном их получению (иными словами, старшую цифру числа N дает последнее частное).

Пример. Преобразовать десятичное число 67 в двоичную форму.

Основание исходной системы счисления R=107. Основание новой системы счисления Q=2.

Согласно приведенному правилу надо исходное число 67 делить на основание новой системы (на 2) по правилам десятичной системы счисления (исходная с/с).

Поскольку процесс деления на 2 очень прост, воспользуемся следующим приемом: в левом столбце будем писать текущие частные, а в правом - текущие остатки от их деления на 2 (это может быть либо 0, либо 1):

67	1	При делении 67 на 2 получается частное 33 и остаток 1;
33	1	при делении 33 - частное 16 и остаток 1 и т.д.
16	0	
8	0	
4	0	
2	0	
1	1	←Старшая цифра числа.
0		

Теперь можно записать число 67 в новой системе счисления. Оно равно 1000011.

3.8. Перевод правильной дроби

Перевод правильной дроби, представленной в с/с с основанием R, в с/с с основанием Q заключается в последовательном умножении этой дроби на основание Q по правилам системы счисления с основанием R, причем перемножают только дробные части. Дробь N в с/с с основанием Q представляется в виде упорядоченной последовательности целых частей произведений в порядке их получения. (Иными словами, старший разряд является первой цифрой произведения). Количество последовательных произведений определяет количество цифр в полученном числе.

Для многих чисел указанный процесс умножения потенциально никогда не кончается. Поэтому он продолжается до тех пор, пока не будет получено необходимое число цифр дробной части. При переводе числа с целью представления ее в “машинной” форме можно точно указать требуемое количество цифр. (Это будет рассматриваться позже, в разделе 1.5).

Пример. Перевести в двоичную систему счисления десятичную дробь

0,7243.

Основание исходной системы счисления R=10. Основание новой системы счисления Q=2.

Согласно приведенного правила исходное число 0,7243 надо умножать на основание новой системы (на 2) по правилам десятичной системы счисления (исходная с/с). Выполним серию умножений до получения, например, шести цифр в двоичном числе:

Искомые цифры дроби:

0,7243 * 2 = **1**,4486	1	-> старшая цифра
0,4486 * 2 = **0**,8972	0	
0,8942 * 2 = **1**,7944	1	
0,7944 * 2 = **1**,5888	1	
0,5888 * 2 = **1**,1776	1	
0,1776 * 2 = **0**,3552	0	
0,3552 * 2 = **0**,7104	0	

Искомое представление число 0,7243 в двоичной системе счисления -> 0,101110.

Обратите внимание, что ***для получения шести цифр дроби выполнено семь умножений***

Это связано с необходимостью выполнить округление, чтобы представить дробь заданной длины более точно.

Из последнего примера, конечная дробь в одной системе счисления может стать бесконечной в другой. Это утверждение справедливо для всех случаев, когда одна система счисления не может быть получена возведением в целую степень основания другой.

Примеры.

- Десятичная дробь **0,2** представляется бесконечной дробью **0,33333...** в шестнадцатиричной системе счисления (основания с/с 10 и 16).
- Шестнадцатиричная дробь **0,B1** представляется конечной дробью **0,10110001** в двоичной системе счисления (основания с/с 16 и 2).

3.9. Перевод неправильной дроби

Перевод неправильной дроби из одной системы счисления в другую осуществляется отдельно для целой и дробной части по правилам, изложенным выше.

3.10. Арифметика цифровых вычислительных машин

Как уже говорилось выше, практически все современные цифровые ЭВМ в качестве основной используют двоичную систему счисления. А все арифметические операции над двоичными числами можно свести к двум элементарным - сложению и сдвигу двоичных кодов, изображающих числа. Это позволит технически реализовать четыре действия арифметики в одном устройстве, называемом арифметико-логическом (АЛУ), используя одни и те же электрические схемы.

Представление чисел со знаками

При выполнении арифметических операций в ЭВМ применяют прямой, обратный и дополнительный коды.

Как уже говорилось выше, кодом называют такую запись числа, которая отличается от естественной и общепринятой. Так вот, в математике естественной формой записи числа является запись, при которой непосредственно перед старшей значащей цифрой числа помещается знак плюс(+) или минус(-), а длина записи определяется величиной числа (иначе, количество символов, использованных для записи разных чисел, как правило, не совпадает). В ЭВМ это не так. Одной из важнейших характеристик любой ЭВМ является длина слова в ней. Длина слова определяется количеством двоичных разрядов слова.

Поэтому в ЭВМ, ***вне зависимости от величины числа, его код всегда имеет фиксированное количество двоичных цифр.***

Кроме этого, в двоичном алфавите нет никаких символов, кроме цифр 0 и 1, и необходимы новые правила для указания знака числа. Суть этих правил сводится к тому, что ***знак плюс изображается цифрой 0, знак минус - цифрой 1, а цифра, изображающая знак всегда записывается самой первой в записи числа.***

Обратите внимание, что ***код числа всегда содержит изображение его знака***, в отличие от математической записи, которая позволяет опускать знак плюс при изображении положительного числа.

Так, код 011101, согласно этим правилам, изображает положительное (самая левая цифра - 0) двоичное число 11101.

Для того, чтобы более просто, и, следовательно, более экономично реализовать устройство АЛУ применяют несколько разных кодов чисел. Это связано с тем, что разные операции в ЭВМ более просто реализуются в разных кодах.

При выполнении арифметических операций в ЭВМ применяют ***прямой, обратный*** и ***дополнительный*** коды чисел.

Прямой код двоичного числа - это само двоичное число, в котором все цифры, изображающие его значение, записываются как в математической записи, а знак числа записывается двоичной цифрой.

При этом никакого символа, отделяющего эту цифру от старшей цифры, используемой при изображении его величины, не допускается. В таких случаях говорят о том, что назначение цифры в коде определяется его позицией.

Примеры.

Изображаемое число		Код	
• +1101	(+13)	0000 1101	(В примерах коды)
• +1011101	(+93)	0101 1101	(изображаются)
• 1101	(-13)	1000 1101	(восемью цифрами)

Итак, прямой код почти не отличается от принятого в математике: для выявления абсолютной величины (модуля) числа, надо отбросить цифру, обозначающую его знак.

Однако применительно к операциям сложения и вычитания такой код неудобен: правила счета для положительных и отрицательных чисел различаются. Чтобы прояснить это обстоятельство, представим что длина кода (слова) равна 5 двоичным разрядам и запишем несколько чисел в нем:

Число	-2	-1	0	+1	+2
Код	10010	10001	00000	00001	00010

Как видно из примера, при использовании прямого кода при переходе значения число через ноль, происходит скачкообразное изменение кода! Поэтому построение устройства, в котором должны выполняться такие действия арифметики, как сложение чисел с разными знаками и вычитание, становится сложной задачей.

Прямой код используется при хранении чисел в памяти ЭВМ, а также при выполнении операций умножения и деления.

Чтобы построить более простые схемы АЛУ предложены и активно применяются обратный и дополнительный коды.

Обратный код положительного числа совпадает с прямым, а при записи отрицательного числа все его цифры, кроме цифры, изображающей знак числа, заменяются на противоположные (0 заменяется на 1, а 1 - на 0).

Примеры записи.

Изображаемое число Код

- +1101 (+13) 0000 1101 (В примерах коды)
- +1011101 (+93) 0101 1101 (изображаются)
- 1101 (-13) 1111 0010 (восемью цифрами)

Сопоставление этой записи с прямым кодом показывает, что непосредственно восстановить абсолютную величину (модуль) отрицательного числа непросто. Однако, в этом коде как к положительным, так и к отрицательным числам можно применять одни и те же правила, а операцию А-В можно заменить операцией сложения чисел А и “минус В”.

Посмотрим, как представляется последовательные числа при переходе через ноль:

Число	-2	-1	0	+1	+2
Код	11101	11110	00000	00001	00010

Из примера видно, что переход через ноль также не выглядит естественным. Отмеченная особенность влечет за собой и следующее - в обратном коде ноль изображают две различающиеся комбинации: 00000 (+0) и 11111 (-0), что усложняет аппаратную реализацию операций.

Для восстановления прямого кода отрицательного числа из обратного кода надо все цифры, кроме цифры, изображающей знак числа, заменить на противоположные

Дополнительный код положительного числа совпадает с прямым, а код отрицательного числа образуется как результат увеличения на 1 его обратного кода.

Иными словами, процесс построения дополнительного кода отрицательного числа можно разбить на два этапа - построить обратный код, а затем из него построить дополнительный.

Проиллюстрируем это на примере.

Число → - 101101

Прямой код → 1101101

Обратный код → 1010010

+1

Дополнительный → 1010011

Примеры записи.

Изображаемое число	Код	
• +1101 (+13)	0000 1101	(В примерах коды)
• +1011101 (+93)	0101 1101	(изображаются)
• 1101 (-13)	1111 0011	(восемью цифрами)

В дополнительном коде, в отличие от обратного, ноль изображается только одной комбинацией, и кроме этого, достаточно естественно получается переход через ноль, если иметь в виду, что любое число, большее другого на 1, получается при прибавлении к этому другому 1 по правилам сложения. Применительно к дополнительному коду это именно так, если принять к сведению, что разрядность слова фиксирована, и единица переноса из старшего разряда теряется, поскольку ее некуда записать:

2 → 11101 + 1 = 11110

1 → 11110 + 1 = 11111

0 → 11111 + 1 = (1)00000 (перенос отбрасывается)

+1 → 00000 + 1 = 00001

+2 → 00001 + 1 = 00010

Для восстановления прямого кода числа из дополнительного нужно полностью повторить (и именно в том же порядке!) действия, которые использовались при переводе из прямого в дополнительный код: сначала все цифры, кроме цифры, изображающей знак, заменить на противоположные, а затем прибавить 1.

Основным достоинством дополнительного кода является то, что в нем единообразно реализуются операции сложения чисел разных знаков (алгебраическое сложение), а операцию вычитания можно свести к операции сложения заменой знака вычитаемого на обратный. Вспомнив, что в памяти ЭВМ числа хранятся в прямом коде, станет ясно, что замена знака вычитаемого может быть выполнена чрезвычайно просто (заменой знака числа в прямом коде на обратный). Именно по указанной причине дополнительный код применяется чаще обратного.

Сложение и вычитание чисел

Сложение и вычитание чисел в обратном и дополнительном кодах выполняется с использованием обычного правила арифметического сложения многоразрядных чисел. Общей для этих кодов особенностью (и очень удобной особенностью) является лишь то, что при поразрядном сложении чисел разряды, изображающие знаки чисел рассматриваются как равноправные разряды двоичного числа, которые складываются друг с другом и с единицей переноса из предыдущего разряда числа по обычным правилам арифметики. Различия же обратного и дополнительного кодов связаны с тем, что делается с единицей переноса из старшего разряда (изображающего, как неоднократно говорилось, знак числа).

При сложении чисел в дополнительном коде единица переноса из старшего разряда игнорируется (теряется), а в обратном коде эту единицу надо прибавить к младшему разряду результата.

Пример 1. Сложить числа +12 и -5.

а) В обратном коде

Десятичная форма	→	+12	-5
Двоичная форма	→	+1100	-101
Прямой код	→	00001100	10000101
Обратный код	→	00001100	11111010

Выполним сложение в столбик:

```
   0 0 0 0 1 1 0 0
   1 1 1 1 1 0 1 0
 ===============
(1) 0 0 0 0 0 1 1 0
                   +  1          (Добавление 1 переноса)
 ==============

 0 0 0 0 0 1 1 1
```

Итак, результат в обратном коде = 00000111.

Поскольку знаковый разряд равен 0, результат положительный, и, следовательно, запись кода числа совпадает с записью прямого кода. Теперь можно восстановить алгебраическую запись результата. Он равен +111 (незначащие нули отброшены), или в десятичной форме +7.

Проверка (+12-5=+7) показывает, что результат верный.

а) В дополнительном коде

Десятичная форма	→	+12	-5
Двоичная форма	→	+1100	-101
Прямой код	→	00001100	10000101
Обратный код	→	00001100	11111010
			+1
Дополнительный код	→	00001100	11111011

Выполним сложение в столбик:

0 0 0 0 1 1 0 0

1 1 1 1 1 0 1 1

============

(1) 0 0 0 0 0 1 1 1

(Перенос игнорируется)

Итак, результат в дополнительном коде = 00000111.

Поскольку знаковый разряд равен 0, результат положительный, и, следовательно, запись кода числа совпадает с записью прямого кода. Теперь можно восстановить алгебраическую запись результата. Он равен +111 (незначащие нули отброшены), или в десятичной форме +7.

Проверка (+12-5=+7) показывает, что результат верный.

Умножение и деление двоичных чисел производится в ЭВМ в прямом коде, а знаки их используются лишь для определения знака результата. Также как и в математике, умножение сводится к операциям сложения и сдвига. Деление выполняется за счет комбинирования сдвигов, вычитаний (в этот момент могут использоваться обратный или дополнительный коды) и сложений.

Задание

Необходимо написать программу выполняющую преобразование систем счисления в соответствии с вариантом задания. Программа должна осуществлять контроль введенных данных на допустимость значений.

Варианты:

1. Преобразовать введённое пользователем целое число из 8сс в 16сс. и в обратную сторону.
2. Преобразовать введенное пользователем целое число из 5сс в 17сс. и в обратную сторону
3. Преобразовать введенное пользователем целое число из 21сс в 10сс. и в обратную сторону
4. Преобразовать введенное пользователем целое число из 2сс в 10сс. и в обратную сторону
5. Преобразовать введенное пользователем целое число из 19сс в 3сс. и в обратную сторону
6. Преобразовать введенное пользователем целое число из 9сс в 10сс. и в обратную сторону
7. Преобразовать введенное пользователем целое число из 3сс в 2сс. и в обратную сторону
8. Преобразовать введенную пользователем десятичную дробь из 13сс в 10сс. и в обратную сторону
9. Преобразовать введенную пользователем десятичную дробь из 2сс в 10сс. и в обратную сторону
10. Преобразовать введенную пользователем десятичную дробь из 16сс в 10сс. и в обратную сторону
11. Преобразовать введенную пользователем десятичную дробь из 8сс в 9сс. и в обратную сторону
12. Преобразовать введенную пользователем десятичную дробь из 7сс в 6сс. и в обратную сторону
13. Преобразовать введенную пользователем десятичную дробь из 20сс в 10сс. и в обратную сторону
14. Преобразовать введенную пользователем десятичную дробь из 6сс в 12сс. и в обратную сторону
15. Преобразовать введенную пользователем десятичную дробь из 21 сс в 10сс. и в обратную сторону

Контрольные вопросы

1. Что такое позиционная система счисления?
2. Как можно записать общую формулу для представления чисел в двоичной системе счисления?
3. Запишите общую формулу для представления чисел шестнадцатеричной системы счисления.
4. Как переводить целое десятичное число в двоичную систему счисления?
5. Что такое тетрада?
6. Какие типы данных языка C# могут хранить переменные длиной в две тетрады, четыре тетрады?
7. Как переводить целое шестнадцатеричное число в двоичную систему счисления?
8. Правило перевода целого десятичного числа в шестнадцатеричную систему счисления.
9. Как преобразовать двоичные и шестнадцатеричные дробные числа в десятичный вид?
10. Чем ограничивается длина дробной части числа при переводе из десятичной системы в двоичную?
11. Как связана длина дробной части десятичного числа и дробная часть шестнадцатеричного числа?