Лениногорский филиал федерального государственного бюджетного образовательного учреждения высшего образования «Казанский национальный исследовательский технический университет им. А.Н. Туполева-КАИ»

Кафедра Естественнонаучных и гуманитарных дисциплин

Методические указания к лабораторным работам
по дисциплине
Основы информатики и программирования

направления подготовки:

09.03.02 Информационные системы и технологии

Лабораторная работа № 1. Основы позиционных систем счисления. Переводы целых чисел в различных системах счисления.

1. ЦЕЛЬ РАБОТЫ

Целью работы является ознакомление студентов с позиционными системами счисления, с компьютерным представлением чисел, алгоритмами перевода целых чисел из одной системы счисления в другую и получение практических навыков перевода чисел.

2. ТЕОРЕТИЧЕСКИЕ ПОЛОЖЕНИЯ

Для обозначения количественных характеристик объектов и явлений используются последовательности символов. Например, для указания года используется последовательность символов "1999". Символы '1 и '9' выбираются из набора символов, в который входят также символы '0', '2', '3',' 4', '5', '6', '7' и '8'.

Набор символов, правил счета и записи чисел в виде последовательности символов из этого набора образуют *систему счисления.* Набор символов системы счисления называется *алфавитом,* а сами символы - *цифрами.*

Все системы счисления можно разделить на две группы:

1. Непозиционные (аддитивные) системы счисления.

2. Позиционные системы счисления.

Непозиционные системы счисления – системы записи чисел, в которых каждой цифре в записи числа соответствует величина, не зависящая от местонахождения этой цифры в записи числа.

Наиболее известной из аддитивных систем счисления является римская система счисления. В ней для обозначения чисел используются буквы латинского алфавита: I, V, X, L, C, D и M.

Число в римской системе счисления обозначается набором стоящих подряд знаков. Значение числа определяется следующим образом:

1) суммируются значения идущих подряд нескольких одинаковых знаков (группа первого вида);

2) вычитаются значения двух знаков, если слева от большего знака стоит меньший, то есть от значения большего знака отнимается значение меньшего (группа второго вида).

Позиционные системы счисления – системы записи чисел, в которых значение цифры в записи числа зависит от ее позиции или местонахождения в числе.

Совокупность различных цифр, используемых в позиционной системе счисления для записи чисел, называется алфавитом системы счисления.

Например, 35 и XXXV - это две различные записи одного и того же числа в арабской и римской системах счисления.

Число q, равное количеству различных цифр в алфавите позиционной системы счисления, называется *основанием системы счисления.*

В алфавите арабской системы счисления q равно десяти, так как алфавит включает в себя десять различных чисел: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9. В соответствии со значением основания арабскую систему счисления называют *десятичной системой счисления.*

Число N_q в позиционной системе счисления с основанием q и алфавитом A в многочленной записи выглядит следующим образом:

$N_q=a_n a_{n-1}...a_1 a_o a_{-1} a_{-2}...a_{-m}=a_n q^n + a_{n-1} q^{n-1} +...+a_1 q^1+ a_0 q^0+ ... +a_{-1} q^{-1}+a_{-2} q^{-2}+ ...+a_{-m} q^m$,

где a_n , a_{n-1},..., a_1, a_o, a_{-1}, a_{-2},..., a_{-m} - цифры из алфавита *A; n. n - 1,..., 1, 0, -1,-2, ...,-m - номера разрядов.*

Разряды с номерами, которые больше или равны 0, образуют целую часть числа. Разряды с номерами, меньшими 0, образуют дробную часть числа. В записи числа эти разряды отделяются разделительной (дробной) точкой или запятой.

Если дробная часть отсутствует, то число называют целым и опускают разделительную точку в записи числа. Если отсутствует целая часть, то число называют правильной дробью и перед разделительной точкой записывают 0 (25 – целое число, 0.14 – правильная дробь).

Например, в арабской системе счисления значение числа 345.678 определяется выражением:

$345.678 = 3 \bullet 10^2 + 4 \bullet 10^1 + 5 \bullet 10^0 + 6 \bullet 10^{-1} + 7 \bullet 10^{-2} + 8 \bullet 10^{-3}$.

Наиболее просто с технической точки зрения реализуются устройства, имеющие два устойчивых состояния (электромагнитное реле замкнуто или разомкнуто, магнитный материал намагничен или размагничен, электронная схема имеет на выходе высокое или низкое напряжение и т.д.). В связи с этим для представления чисел в ЭВМ применяется двоичная система счисления.

Двоичная система счисления имеет алфавит, состоящий только из двух цифр: 0 и 1. Основанием двоичной системы счисления является число два.

Кроме двоичной системы счисления при вводе и выводе чисел используются также десятичная, восьмеричная и шестнадцатеричная системы счисления. Запись чисел в этих системах короче и удобнее записи чисел в двоичной системе счисления.

Восьмеричная система счисления имеет алфавит из восьми цифр: 0, 1, 2, 3, 4, 5, 6, 7. Основанием восьмеричной системы является число восемь.

Шестнадцатеричная система счисления состоит из шестнадцати различных цифр: 0, 1,2, 3, 4, 5, 6, 7, 8, 9, *a, b, c, d, e, f.* Основанием шестнадцатеричной системы счисления является число шестнадцать.

Для перевода целых чисел из десятичной в систему счисления с основанием q исходное число необходимо разделить на основание системы счисления q. При этом будет получено частное (целое число) и остаток от деления (целое число). На следующем шаге алгоритма необходимо полученное частное также разделить на основание системы счисления. Будет получено новое частное и остаток. Деление очередного частного производится до тех пор, пока очередное частное не окажется строго меньше основания системы счисления q. Цифре старшего разряда будет соответствовать частное последнего деления. Цифре следующего разряда – остаток последнего деления. Цифре следующего разряда – остаток предпоследнего деления и т. д., цифре младшего разряда будет соответствовать остаток первого деления.

Пример 1. Перевести число 45 из 10-ой системы счисления в 2-ую систему счисления.

1) 45:2 = 22 (1);

2) 22:2 = 11 (0);
3) 11:2 = 5 (1);
4) 5:2 = 2 (1);
5) 2:2 = 1 (0).

Запишем число в двоичной системе счисления: $45_{10} = 101101_2$.

Ответ: $45_{10} = 101101_2$.

Пример 2. Перевести число 1226 из 10-ой системы счисления в восьмеричную систему счисления.

1) 1226:8 = 153 (2);
2) 153:8 = 19 (1);
3) 19:8 = 2 (3);

Запишем число в восьмеричной системе счисления: $1226_{10} = 2312_8$.

Ответ: $1226_{10} = 2312_8$.

Пример 3. Перевести число 476 из 10-ой системы счисления в шестнадцатеричную систему счисления.

1) 476:16 = 29 (12 = C);
2) 29:16 = 1 (13 = D).

$476_{10} = 1DC_{16}$.

Ответ: $476_{10} = 1DC_{16}$.

Для перевода чисел из системы счисления с основанием q в десятичную систему счисления необходимо число представить в многочленной форме, которая представляет собой сумму m + n + 1 слагаемых. Каждое слагаемое ставится в соответствие разряду исходного числа и представляет собой произведение двух сомножителей. Первый сомножитель - десятичное число, соответствующее по весу цифре разряда исходного числа. Второй сомножитель - это степень, основанием которого является основание системы счисления, а показателем степени - номер разряда.

$$N_{(10)} = a_n^* q^n + a_{n-1}^* q^{n-1} + \ldots + a_1^* q^{n-1} + a_0^* q^0 + a_{-1}^* q^{-1} + a_{-m}^* q^{-m}$$

Пример 4. Перевести число 3E5A1 из шестнадцатеричной системы счисления в десятичную систему счисления:

$3^4E^35^2A^11^0{}_{16} = 3^1 \cdot 16^4 + E^1 \cdot 16^3 + 5^1 \cdot 16^2 + A^1 \cdot 16^1 + 1^1 \cdot 16^0 =$

$= 3^1 \cdot 16^4 + 14^1 \cdot 16^3 + 5^1 \cdot 16^2 + 10^1 \cdot 16^1 + 1^1 \cdot 16^0 = 255393_{10}$

Запишем число в десятичной системе счисления:

Пример 5. Перевести число 1101111011 из двоичной системы счисления в десятичную систему счисления.

$1^91^80^71^61^51^41^30^21^11^0{}_2 = 1 \cdot 2^9 + 1 \cdot 2^8 + 0 \cdot 2^7 + 1 \cdot 2^6 + 1 \cdot 2^5 + 1 \cdot 2^4 + 1 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0 = 512 + 256 + 0 + 64 + 32 + 16 + 8 + 0 + 2 + 1 = 981_{10..}$

Запишем число в десятичной системе счисления: $1101111011_2 = 891_{10}$

Ответ: $1101111011_2 = 891_{10}$

Пример 6. Перевести число 1573 из восьмеричной системы счисления в десятичную систему счисления.

$1^35^27^13^0{}_8 = 1 \cdot 8^3 + 5 \cdot 8^2 + 7 \cdot 8^1 + 3 \cdot 8^0 = 512 + 320 + 56 + 3 = 891_{10.}$

Запишем число в десятичной системе счисления: $1573_8 = 891_{10}$.

Ответ: $1573_8 = 891_{10}$.

Для перевода чисел из двоичной в восьмеричную и шестнадцатеричную систему счисления необходимо разбить исходное число на триады (тетрады). Триада (тетрада) представляет собой последовательность трех (четырех) соседних двоичных цифр взятых из записи исходного числа.

Разбиение исходного числа производится с конца на начало при движении влево. Крайняя левая группа, если она не укомплектована двоичными цифрами, дополняется нулями: слева. Далее необходимо каждой триаде (тетраде) поставить в соответствие цифру восьмеричной (шестнадцатеричной) системы счисления. Записывается число. Порядок цифр восьмеричных (шестнадцатеричных) цифр в записи искомого числа такой же, что и порядок соответствующих триад (тетрад) в записи исходного числа.

Для перевода из восьмеричной, шестнадцатеричной систем счисления в двоичную систему счисления необходимо каждой цифре исходного числа поставить в соответствие триаду (тетраду) двоичных цифр. Записывается искомое число. Искомое число будет состоять из последовательности триад (тетрад). Порядок триад такой же, как и порядок соответствующих цифр в записи исходного числа.

Для перевода чисел из двоичной системы счисления в восьмеричную и обратно необходимо использовать таблицу 1.

Таблица 1

Число в 10-ой сс	Число в 8-ой сс	Число в 2-ой сс
0	0	000
1	1	001
2	2	010
3	3	011
4	4	100
5	5	101
6	6	110
7	7	111

Для перевода чисел из двоичной системы счисления в шестнадцатеричную и обратно необходимо использовать таблицу 2.

Таблица 2

Число в 10-ой сс	Число в 16-ой сс	Число в 2-ой сс
0	0	0000
1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111
8	8	1000
9	9	1001
10	A	1010
11	B	1011
12	C	1100
13	D	1101
14	E	1110
15	F	1111

Пример 7. Выполнить перевод числа 101101_2 из двоичной системы счисления в восьмеричную систему счисления.

Разобьем исходную запись числа на триады двоичных разрядов:
101101 → 101 101.
Поставим в соответствие каждой триаде восьмеричную цифру:
101_2 → 5_8;
101_2 → 5_8.
Запишем число: $101101_2 = 55_8$.
Ответ: $101101_2 = 55_8$.

Пример 8. Выполнить перевод числа 101101_2 из двоичной системы счисления в шестнадцатеричную систему счисления.
Разобьем исходную запись числа на тетрады двоичных разрядов:
101101 → 0010 1101
Поставим в соответствие каждой тетраде шестнадцатеричную цифру:
0010_2 → 2_{16};
1101_2 → d_{16}
Запишем искомое число: $101101_2 = 2d_{16}$.
Ответ: $101101_2 = 2d_{16}$.

Пример 9. Выполнить перевод числа 2312_8 из восьмеричной системы счисления в двоичную систему счисления.
Каждой цифре в записи числа поставим в соответствие триаду двоичных цифр:
2_8 → 010_2;
3_8 → 011_2;
1_8 → 001_2;
2_8 → 010_2.
Запишем искомое число: $2312_8 = 10011001010_2$.
Ответ: 2312_8 =.10011001010_2..

Пример 10. Выполнить перевод числа $1dc_{16}$ из шестнадцатеричной системы счисления в двоичную систему счисления:
Каждой цифре в записи числа поставим в соответствие тетраду двоичных цифр:
1_{16} → 0001_2;
d_{16} → 1101_2;
c_{16} → 1100_2.
Запишем искомое число: $1dc_{16} = 111011100_2$
Ответ:. $1dc_{16} = 111011100_2$

3. ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ
1. Ознакомиться с основными теоретическими положениями.
2. Пройти тестирование в программе «Системы счисления» (см. Приложение 2).

Лабораторная работа № 2. Перевод дробных чисел в различных системах счислениях.

1. ЦЕЛЬ РАБОТЫ
Целью работы является ознакомление обучающихся с алгоритмами перевода дробных чисел из одной системы счисления в другую и получение практиче-

ских навыков перевода дробных чисел из одной системы счисления в другую систему счисления.

2. ТЕОРЕТИЧЕСКИЕ ПОЛОЖЕНИЯ

Набор символов, правил счета и записи чисел в виде последовательности символов из этого набора образуют *систему счисления.* Набор символов системы счисления называется *алфавитом,* а сами символы - *цифрами.*

Число N_q в позиционной системе счисления с основанием q и алфавитом A в многочленной записи выглядит следующим образом:

$$N_q=a_na_{n-1}...a_1a_0a_{-1}a_{-2}...a_{-m}=a_nq^n+a_{n-1}q^{n-1}+…+a_1q^1+a_0q^0+…+a_{-1}q^{-1}+a_{-2}q^{-2}+...+a_{-m}q^m,$$

где a_n , a_{n-1},..., a_1, a_o, a_{-1}, a_{-2},..., a_{-m} - цифры из алфавита *A; п. п - 1,..., 1, 0, -1,-2, ...,-т - номера разрядов.*

Разряды с номерами, которые больше или равны 0, образуют целую часть числа. Разряды с номерами, меньшими 0, образуют дробную часть числа. В записи числа эти разряды отделяются разделительной (дробной) точкой или запятой.

Если дробная часть отсутствует, то число называют целым и опускают разделительную точку в записи числа. Если отсутствует целая часть, то число называют правильной дробью и перед разделительной точкой записывают 0 (25 – целое число, 0.14 – правильная дробь).

Например, в арабской системе счисления значение числа 345.678 определяется выражением:

$$345.678 = 3 • 10^2 + 4 • 10^1 + 5 • 10^0 + 6 • 10^{-1} + 7 • 10^{-2} + 8 • 10^{-3}.$$

Для перевода правильной десятичной дроби из десятичной системы счисления в систему счисления с основанием q правильную дробь необходимо умножить на основание системы счисления q. При этом будет получена целая и дробная часть произведения. На следующем шаге алгоритма необходимо дробную часть произведения умножить на основание системы счисления q. При этом будет получена также целая и дробная часть произведения. Дробные части произведений далее умножаются на основание системы счисления q.

Этот процесс завершается в трёх случаях:

1. Дробная часть произведения оказывается равной нулю. В этом случае перевод исходного десятичного числа в систему счисления с основанием q точный.
2. Дробная часть произведения оказывается равной одной из дробных частей произведений, найденных ранее. В этом случае искомое число представляет собой периодическую дробь (перевод приближенный).
3. Задана точность перевода, определяемая количеством разрядов в дробной части числа. В этом случае считается, что все разряды дробной части искомого числа определены, когда количество найденных произведений равно точности перевода.

Запишем исходное число. Цифре разряда с номером -1 соответствует целая часть первого произведения. Цифре разряда с номером -2 соответствует целая часть второго произведения, и т. д.

При вводе дробных чисел перевод в двоичную систему счисления может быть произведен приближенно, при обратном переводе из двоичной в десятичную систему счисления результат, если не производить округления, будет меньше исходного числа.

Пример 1. Перевести из десятичной системы счисления в двоичную, восьмеричную и шестнадцатеричную системы счисления и обратно из полученных представлений в десятичную систему счисления. Перевод производить с точностью: для шестнадцатеричной системы счисления до 2-х знаков, для восьмеричной системы счисления – до 3 знаков, для десятичной системы счисления – до 5 знаков. Сравнить результаты, полученные после перевода в десятичную систему счисления с исходным числом. Определить относительную ошибку перевода.

1. Выполнить перевод числа 17.97 из десятичной системы счисления в двоичную систему счисления.

1.1. Переводим целую часть числа:

1) 17 : 2 = 8 (1), 8 >= 2;
2) 8 : 2 = 4 (0), 4 >= 2;
3) 4 : 2 = 2 (0), 2 >= 2;
4) 2 : 2 = 1 (0), 1< 2 – конец перевода.

Итак, $17_{10} = 10001_2$

1.2. Переводим дробную часть числа:

1) $0.97 \cdot 2 = 1.94$ (1);
2) $0.94 \cdot 2 = 1.88$ (1);
3) $0.88 \cdot 2 = 1.76$ (1);
4) $0.76 \cdot 2 = 1.52$ (1);
5) $0.52 \cdot 2 = 1.04$ (1).

Итак, $0.97_{10} = 0.11111_2$

Таким образом, $17.97_{10} = 10001.11111_2$

2. Выполним перевод числа 17.97 из десятичной системы счисления в восьмеричную систему счисления.

2.1. Переводим целую часть числа:

1) 17 : 8 = 2 (1), 2 < 8 – конец перевода.

Итак, $17_{10} = 21_8$

2.2. Переводим дробную часть числа:

1) $0.97 \cdot 8 = 7.76$ (7);
2) $0.76 \cdot 8 = 6.08$ (6);
3) $0.08 \cdot 8 = 0.64$ (0);

Итак, $0.97_{10} = 0.760_8$

Таким образом, $17.97_{10} = 21.760_8$

3. Выполним перевод числа 17.97 из десятичной системы счисления в шестнадцатеричную систему счисления.

3.1. Переводим целую часть числа:

1) 17 : 16 = 1 (1), 1 < 16 – конец перевода.

Итак, $17_{10} = 11_{16}$

3.2. Переводим целую часть числа:

1) $0.97 \cdot 16 = 15.52$ (15=F);
2) $0.52 \cdot 16 = 8.32$ (8);

Итак, $0.97_{10} = F8_{16}$

Таким образом, $17.97_{10} = 11.F8_{16}$

4. Выполним перевод числа 1001.11111 из двоичной системы счисления в десятичную систему счисления:

$1^4 0^3 0^2 0^1 1^0 . 1^{-1} 1^{-2} 1^{-3} 1^{-4} 1^{-5}{}_2 = 1 \cdot 2^4 + 0 \cdot 2^3 + 0 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 + 1 \cdot 2^{-1} + 1 \cdot 2^{-2} + 1 \cdot 2^{-3} +$

$+ 1 \cdot 2^{-4} + 1 \cdot 2^{-5} = 16 + 0 + 0 + 0 + 1 + 0.5 + 0.25 + 0.125 + 0.0625 + 0.03125 =$
$= 17.96875_{10}$.

Запишем искомое число: $10001.11111_2 = 17.96875_{10,}$

Имеем: $17,97 \neq 17,96875$

5. Вычислим относительную ошибку E:

$$\varepsilon = \frac{(N_{10} - N_{10})}{N_{10}} * 100\% = \frac{(17.97 - 17.96875)}{17.97} * 100 = 0.06\%$$

6. Выполним перевод числа 21.760 из восьмеричной системы счисления в десятичную систему счисления.

$2^1 1^0 . 7^{-1} 6^{-2} 0^{-3}{}_8 = 2 \cdot 8^1 + 1 \cdot 8^0 + 6 \cdot 8^{-2} + 0 \cdot 8^{-3} = 16 + 1 + 0.875 + 0.09375 + 0 =$
$= 17.96875_{10}$.

Запишем искомое число: $21.760_8 = 17.96875_{10}$

Имеем: $17.97 \neq 17.96875$

7. Вычислим относительную ошибку E:

$$\varepsilon = \frac{(N_{10} - N_{10})}{N_{10}} * 100\% = \frac{(17.97 - 17.96875)}{17.97} * 100\% = 0.0069\%$$

8. Выполним перевод числа 11.F8 из шестнадцатеричной системы счисления в десятичную систему счисления:

$1^1 1^0 . F^{-1} 8^{-2}{}_{16} = 1 \cdot 16^1 + 1 \cdot 16^0 + 15 \cdot 16^{-1} + 8 \cdot 16^{-2} = 16 + 1 + 0.9375 + 0.03125 =$
$= 17.96875_{10}$

Запишем искомое число: $11.f851e_{16} = 17.96875_{10}$

Имеем: $17.97 \neq 17.96875$

9. Вычислим относительную ошибку E:

$$\varepsilon = \frac{(N_{10} - N_{10})}{N_{10}} * 100\% = \frac{(17.97 - 17.96875)}{17.97} * 100\% = 0.0069\%$$

Для перевода чисел из двоичной в восьмеричную и шестнадцатеричную систему счисления необходимо разбить исходное число на триады (тетрады). Триада (тетрада) представляет собой последовательность трех (четырех) соседних двоичных цифр взятых из записи исходного числа.

Разбиение исходного числа производится от разделительной точки. Целая часть числа разбивается при движении от разделительной точки влево. Дробная часть числа разбивается при движении от разделительной точки вправо. Крайняя левая группа, если она не укомплектована двоичными цифрами, дополняется нулями: слева. Крайняя правая группа, если она не укомплектована двоичными цифрами, дополняется нулями: справа. Далее необходимо каждой триаде (тетраде) поставить в соответствие цифру восьмеричной (шестнадцатеричной) системы счисления. Запишем число. Порядок цифр восьмеричных (шестнадцатеричных) цифр в записи искомого числа такой же, что и порядок соответствующих триад (тетрад) в записи исходного числа.

Для перевода из восьмеричной, шестнадцатеричной систем счисления в двоичную систему счисления необходимо каждой цифре исходного числа поставить в соответствие триаду (тетраду) двоичных цифр. Запишем искомое число. Искомое число будет состоять из последовательности триад (тетрад). Порядок триад такой же, как и порядок соответствующих цифр в записи исходного числа.

Пример 2. Выполнить перевод числа 23.12_8 из восьмеричной системы счисления в двоичную систему счисления.

Каждой цифре в записи числа поставим в соответствие триаду двоичных цифр:

$2_8 \rightarrow 010_2$;

$3_8 \rightarrow 011_2$;

$1_8 \rightarrow 001_2$;

$2_8 \rightarrow 010_2$.

Запишем искомое число: $2312_8 = 010011.001010_2$.

Ответ: $23.12_8 = 10011.00101_2$..

Пример 3. Выполнить перевод числа $1d.c_{16}$ из шестнадцатеричной системы счисления в двоичную систему счисления.

Каждой цифре в записи числа поставим в соответствие тетраду двоичных цифр:

$1_{16} \rightarrow 0001_2$;

$d_{16} \rightarrow 1101_2$;

$c_{16} \rightarrow 1100_2$.

Запишем искомое число: $1d.c_{16} = 00011101.1100_2$

Ответ:. $1d.c_{16} = 11101.11_2$

Пример 4. Выполнить перевод числа 10110.11101_2 из двоичной системы счисления в восьмеричную систему счисления:

Разобьем исходную запись числа на триады двоичных разрядов:

$10110.11101_2 \rightarrow$ 010 110.111 010.

Поставим в соответствие каждой триаде восьмеричную цифру:

$010_2 \rightarrow 2_8$;

$110_2 \rightarrow 6_8$;

$111_2 \rightarrow 7_8$;

$010_2 \rightarrow 2_8$;

Запишем число: $10110.11101_2 = 26.72_8$.

Ответ: $10110.11101_2 = 26.72_8$.

Пример 5. Выполнить перевод числа 10110.11101_2 из двоичной системы счисления в шестнадцатеричную систему счисления:

Разобьем исходную запись числа на тетрады двоичных разрядов:

$10110.11101_2 \rightarrow$ 0001 0110.1110 1000.

Поставим в соответствие каждой тетраде шестнадцатеричную цифру:

$0001_2 \rightarrow 1_{16}$;

$0110_2 \rightarrow 6_{16}$;

$1110_2 \rightarrow E_{16}$;

$1000_2 \rightarrow 8_{16}$;

Запишем число: $10110.11101_2 = 16.E8_{16}$.

Ответ: $10110.11101_2 = 16.E8_{16}$.

3. ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

1. Ознакомиться с основными теоретическими положениями.
2. Пройти тестирование в программе

Лабораторная работа № 3. Представление чисел в памяти ЭВМ.

1. ЦЕЛЬ РАБОТЫ

Целью работы является ознакомление обучающихся с алгоритмами представления различных чисел (положительных, отрицательных, дробных) в памяти ЭВМ и получение практических навыков представления чисел в памяти ЭВМ.

2. ТЕОРЕТИЧЕСКИЕ ПОЛОЖЕНИЯ

1. Хранение в ЭВМ целых чисел.

Для хранения программ, исходных, промежуточных данных и результатов решения задач в ЭВМ используется специальное устройство - *оперативная память, или оперативное запоминающее устройство (ОЗУ).* ОЗУ состоит из ячеек (слов), каждая из которых имеет свой номер, или адрес. Каждая ячейка содержит (в зависимости от типа ЭВМ) одно и то же количество двоичных разрядов (битов). Основной единицей измерения емкости ячейки памяти является байт, в который включается восемь битов.

В зависимости от типа хранящейся в памяти информации ее элементы могут занимать один или несколько последовательных ячеек (байтов).

Рассмотрим структуры хранения в памяти ЭВМ следующих типов данных:

- целые числа без знака, принимающие значения от 0 до 65535;
- целые числа со знаком, принимающие значения от -32768 до 32767;
- дробные числа со знаком с плавающей точкой в коротком формате.

В ЭВМ для хранения целых чисел используется их представление в двоичной системе счисления. Поэтому перед записью в память ЭВМ число преобразуется в двоичную систему счисления.

Формат хранения целого двоичного числа без знака, принимающего значения от 0 до 65535, представлена рис. 1.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
b	b	b	b	b	b	b	b	b	b	b	b	b	b	b	b

Рис. 1.

На рис. 1 символ *b* обозначает двоичную цифру: 0 или 1, а целое число над этим символом – номер ее разряда. Цифры с номерами разрядов от 0 до 7 образуют младший байт числа, а цифры с номерами разрядов от 8 до 15— старший байт числа.

На рис. 2 изображено двоичное число 111111001 в рассмотренном формате:

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	1	1	1	1	1	1	0	0	1

Рис. 2.

Формат хранения целого двоичного числа со знаком, принимающего значения от -32768 до 32767, представлен на рис. 3. Этот формат аналогичен формату хранения двоичного числа без знака (см. рис. 1), за исключением того, что старший разряд отводится для хранения знака числа s. При хранении положительного числа знаковый разряд принимает значение 0, а при хранении отрицательного числа – 1.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
s	b	b	b	b	b	b	b	b	b	b	b	b	b	b	b

Рис. 3.

Отрицательные числа хранятся в рассматриваемом формате в дополнительном коде. Чтобы найти двоичное представление отрицательного числа в дополнительном коде с заданным количеством двоичных разрядов, необходимо выполнить следующие действия:

1. Перевести абсолютное значение исходного числа (положительную форму числа) в двоичную систему счисления.

2. Дополнить слева нулями полученное двоичное число до необходимой разрядности.

3. Найти обратный код полученного числа, заменив двоичные нули на двоичные единицы, а двоичные единицы – на двоичные нули.

4. Добавить к полученному двоичному числу двоичную единицу.

Пример 1. Определить структуру хранения числа -24 в формате целого двоичного числа со знаком (количество двоичных разрядов равно 16):

1. Переведем абсолютное значение исходного числа (положительную форму числа) в двоичную систему счисления:

$24_{10} = 11000_2$.

2. Дополним слева нулями полученное двоичное число 11000 до разрядности, равной шестнадцати:

$11000_2 =.0000000000011000_2$.

3. Найдем обратный код полученного числа 0000000000011000, заменив двоичные нули на двоичные единицы, а двоичные единицы — на двоичные нули:

0000000000011000 → 1111111111100111.

4. Добавим к полученному двоичному числу 1111111111100111 двоичную единицу (операцию сложения выполняем в двоичной системе счисления):

$$
\begin{array}{r}
+\begin{array}{r}1111111111100111\\ 0000000000000001\end{array}\\
\hline
11111111\ 11101000
\end{array}
$$

Таким образом, число -24 в памяти ЭВМ будет представлено кодом 1111111111101000 (рис. 4).

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	1	1	1	1	1	1	1	1	1	1	0	1	0	0	0

Рис. 4.

Для перевода числа, хранимого в формате целого двоичного числа со знаком, в десятичную систему счисления воспользуемся следующей процедурой:

1. Если знаковый разряд равен 0 (число положительное), то производится перевод 15-разрядного двоичного числа в десятичную систему счисления. Иначе -

перейти к п. 2.

2. Если знаковый разряд равен -1 (число отрицательное), то выполняются последовательно следующие действия:

1) восстановление положительной формы числа;

2) перевод положительной формы числа из двоичной системы счисления в десятичную систему счисления;

3) запись отрицательного числа в десятичной системе счисления.

Для восстановления положительной формы числа можно воспользоваться процедурой, аналогичной рассмотренной:

- находим обратный код от числа, представленного в дополнительном коде.
- добавляем к полученному двоичному числу двоичную единицу.

Пример 2. Необходимо восстановить десятичное представление числа (рис. 4):

1. Так как знаковый разряд числа не равен 0 (число не положительное), то переходим к п. 2.

2. Выполняем последовательно следующие действия:

2.1. Восстанавливаем положительную форму числа:

- найдем обратный код числа 1111111111101000. Он имеет вид

$N_{обр}$ = 0000000000010111;

- добавим к полученному числу единицу;

$$
\begin{array}{r}
+\begin{array}{r}0000000000010111\\ 0000000000000001\end{array}\\
\hline
0000000000011000
\end{array}
$$

Таким образом, в двоичной системе счисления .положительная форма числа имеет вид:

$N_{пол\ 2}$=0000000000011000=11000.

2.2. Переводим положительную форму числа из двоичной системы счисления в десятичную систему счисления:

$N_{пол\ 2}$ = 11000 = N_{10} = $1 \bullet 2^4 + 1 \bullet 2^3 + 0 \bullet 2^2 + 0 \bullet 2^1 + 0 \bullet 2^0 = 24_{10}$;

$N_{пол\ 10}$ =: 24_{10}.

2.3. Записываем отрицательное число в десятичной системе счисления:

$N_{10} = -24_{10}$.

2. Хранение в ЭВМ дробных чисел.

Для хранения дробных чисел в ЭВМ используется нормализованное двоичное число с плавающей точкой. Всякое число с плавающей точкой состоит из двух частей: мантиссы и порядка. Мантисса содержит значащие цифры, а с помощью порядка указывается положение двоичной точки. Обе части числа хранятся вместе в четырех (короткий формат) или в восьми (длинный формат) последовательных байтах.

Представление числа включает в себя:

- знак числа;
- значение порядка;
- значение мантиссы.

Рассмотрим хранение дробного числа в коротком формате (рис. 5).

Зн	Порядок								Мантисса																						
Первый байт								Второй байт								Третий байт								Четвертый байт							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
b	b	b	b	b	b	b	b	b	b	b	b	b	b	b	b	b	b	b	b	b	b	b	b	b	b	b	b	b	b	b	b

Рис. 5.

Знак числа (на рисунке он обозначен символом Зн) представлен одним битом и равен 0, если число - положительное, и равен 1, если число – отрицательное.

Для хранения порядка выделяется 8 битов (7 битов первого байта и один старший бит второго байта числа). Порядок числа представляется в двоичной системе счисления. С помощью восьми битов можно представлять числа от 0 до 255. Это означает, что значением порядка является целое число от -128 до 127. Для того чтобы не хранить знак порядка, последний представляется с избытком — смещенный порядок. Показателем избытка (смещение порядка) является число 127. Значение смещенного порядка образуется сложением значения действительного порядка с показателем избытка.

Мантисса представляется в двоичной системе счисления и занимает 23 двоичных разряда (короткий формат).

Точка всегда подразумевается перед самым левым битом. Соответственно этому значения цифр представляются в двоичных позициях мантиссы слева направо: 2^{-1}, 2^{-2}, ... 2^{-23}.

Числа хранятся в нормализованном виде и мантисса принимает значения на полусегменте [1,2).

Рассмотрим процесс нормализации числа. Первоначально порядок считается равным 0, а его код в представлении числа равен 127 (0+127).

Первый случай: число меньше 1. Процесс нормализации заключается в сдвиге разделительной точки числа вправо и в одновременном уменьшении значения порядка на число сдвигов до тех пор, пока мантисса не будет располагаться на полусегменте [1,2).

Второй случай: число больше или равно 2. Процесс нормализации заключается в сдвиге разделительной точки числа влево и в одновременном увеличении значения порядка на число сдвигов до тех пор, пока мантисса не будет располагаться на полусегменте [1,2).

Третий случай: число располагается на полусегменте [1,2). В этом случае нормализация не требуется.

Если число нормализовано, то старший бит мантиссы всегда равен 1 и поэтому в его хранении нет необходимости.

Отрицательные дробные числа не представляются в дополнительном коде. В этом случае знак числа равен 1.

В длинном формате представляются числа с повышенной точностью. При этом все число занимает 8 байтов, из которых для представления мантиссы используются 55 битов.

Процедура получения представления дробного числа, заданного в десятичной системе счисления, следующая:

1. Произвести перевод числа из десятичной системы счисления в двоичную систему счисления. Количество значащих разрядов в числе, включая целую и дробную часть, должно быть равно 25. Если число меньше 1, то при подсчете значащих разрядов левые нули до первого единичного разряда не учитываются.

2. Произвести округление числа. Для этого к полученному числу необходимо добавить единицу, по весу равную самому младшему разряду числа, а затем оставить в записи числа 24 значащих разряда.

3. Произвести нормализацию числа.

4. Отбросить старший разряд.

5. Добавить к порядку число 127 и записать значение порядка в двоичной системе счисления.

6. Сформировать знаковый разряд.

Пример 3. Представим в памяти ЭВМ десятичное число 10.6.

1. Переведем число из десятичной системы счисления в двоичную систему счисления:

$10.6_{10} = 1010.(1001)_2 = 1010.100110011001100110011_2$.

2. Округлим число:

$$
\begin{array}{r}
+\begin{array}{r} 1010.100110011001100110011_2 \\ 0.000000000000000000001_2 \end{array} \\
\hline
1010.10011001100110011010_2
\end{array}
$$

3. Нормализуем число:

$1010.10011001100110011010_2 = 1.0101001100110011001l010_2 \bullet 2^3{}_{10},$.

4. Отбросим старший разряд:

$1.01010011001100110011010 \rightarrow .01010011001100110011010$.

5. Определим двоичный код порядка:

$3_{10} + 127_{10} = 130_{10} = 10000010_2$.

6. Определим знаковый разряд: знаковый разряд положительного числа равен 0.

Представление указанного числа в памяти ЭВМ показано на рис. 6 (шестнадцатеричное представление числа в памяти: 4129999а).

Зн	Порядок							Мантисса																							
Первый байт								Второй байт								Третий байт								Четвертый байт							
7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0

<table>
<tr><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td></tr>
<tr><td colspan="4">4</td><td colspan="4">1</td><td colspan="4">2</td><td colspan="4">9</td><td colspan="4">9</td><td colspan="4">9</td><td colspan="4">9</td><td colspan="4">a</td></tr>
</table>

Рис. 6.

3. ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

1. Ознакомиться с основными теоретическими положениями.
2. Пройти тестирование в программе