

Конспект новой темы(можно взять с презентации).

[Ссылка на лекцию – Представление целых чисел в памяти компьютера](#)

Домашнее задание(в конце лекции)

Вся информация, обрабатываемая компьютерами, хранится в них в двоичном виде. Каким же образом осуществляется это хранение?

Информация, вводимая в компьютер и возникающая в ходе его работы, хранится в его памяти. Память компьютера можно представить, как длинную страницу, состоящую из отдельных строк. Каждая такая строка называется **ячейкой памяти**.

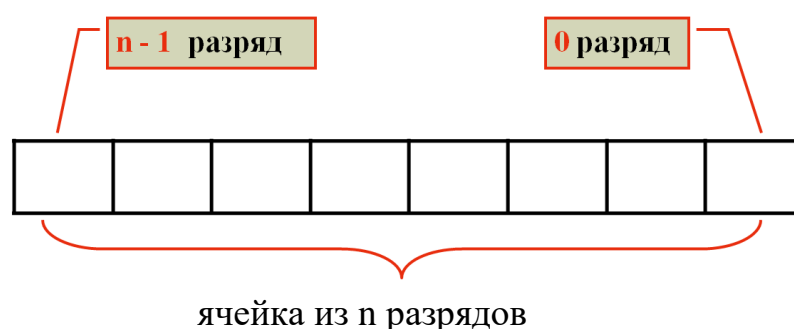
Ячейка – это часть памяти компьютера, вмещающая в себя информацию, доступную для обработки **отдельной командой** процессора. Содержимое ячейки памяти называется **машинным словом**.

Ячейка памяти состоит из некоторого числа однородных элементов. Каждый элемент способен находиться в одном из двух состояний и служит для изображения одного из разрядов числа. Именно поэтому каждый элемент ячейки называют **разрядом**.

Нумерацию разрядов в ячейке принято вести справа налево, самый правый разряд имеет порядковый номер 0. Это младший разряд ячейки памяти, старший разряд имеет порядковый номер (n-1) в n-разрядной ячейке памяти.

Содержимым любого разряда может быть либо 0, либо 1.

Содержимое ячейки памяти называется **машинным словом**. Ячейка памяти разделяется на разряды, в каждом из которых хранится разряд числа.



Например, самые современные персональные компьютеры являются 64-разрядным, то есть машинное слово и соответственно, ячейка памяти, состоит из 64 разрядов или **битов**.

Бит — минимальная единица измерения информации. Каждый бит может принимать значение 0 или 1. **Битом** также называют **разряд** ячейки памяти ЭВМ.

Стандартный размер наименьшей ячейки памяти равен восьми битам, то есть восьми двоичным разрядам. Совокупность из 8 битов является основной единицей представления данных – байт.

Байт (от английского byte – слог) – часть машинного слова, состоящая из 8 бит, обрабатываемая в ЭВМ как одно целое. На экране – ячейка памяти, состоящая из 8 разрядов – это байт. Младший разряд имеет порядковый номер 0, старший разряд – порядковый номер 7.

8 бит = 1 байт

Главные правила представления данных в компьютере.

- Правило 1. *Данные (и программы) в памяти компьютера хранятся в двоичном виде, т. е. в виде цепочек единиц и нулей.*
- Правило 2. *Представление данных в компьютере дискретно.*
- Правило 3. *Множество представимых в памяти компьютера величин ограничено и конечно.*
- Правило 4. *В памяти компьютера числа хранятся в двоичной системе счисления.*

Типы числовых величин

Целые (формат с фиксированной запятой)

Вещественные (формат с плавающей запятой)

- Представление в памяти компьютера целых чисел без знака и со знаком.
- Представление в памяти компьютера вещественных чисел.

Для представления чисел в памяти компьютера используются два формата: *формат с фиксированной точкой* и *формат с плавающей точкой*. В формате с фиксированной точкой представляются только целые числа, в формате с плавающей точкой – вещественные числа (целые и дробные).

В подавляющем большинстве задач, решаемых с помощью ЭВМ, многие действия сводятся к операциям над целыми числами. Сюда относятся задачи экономического характера, при решении которых данными служат количества акций, сотрудников, деталей, транспортных средств и т.д. Целые числа используются для обозначения даты и времени, и для нумерации различных объектов: элементов массивов, записей в базах данных, машинных адресов и т.д.

Целые числа могут представляться в компьютере со знаком или без знака.

Целые числа без знака обычно занимают в памяти один или два байта и принимают в однобайтовом формате значения от 00000000_2 до 11111111_2 , а в двухбайтовом формате - от $00000000\ 00000000_2$ до $11111111\ 11111111_2$.

Целые числа со знаком обычно занимают в памяти компьютера один, два или четыре байта, при этом самый левый (старший) разряд содержит информацию о знаке числа. Знак "плюс" кодируется нулем, а "минус" - единицей.

В компьютерной технике применяются три формы записи (кодирования) целых чисел со знаком: *прямой код*, *обратный код*, *дополнительный код*.

Прямой код – это представление числа в двоичной системе счисления, при этом первый разряд отводится под знак числа. Если число положительное, то в первом разряде находится 0, если число отрицательное, в первом разряде указывается единица.

Таким образом, используя прямой код, в 16 – ти разрядной ячейке можно записать 16 – ти разрядное число в двоичной системе счисления. Например:

Положительное десятичное число 24 представляется, как

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

Знак числа «+»

Отрицательное десятичное число – 24 представляется, как

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

Знак числа «-»

На самом деле прямой код используется почти исключительно для положительных чисел.

Обратный код для положительного числа в двоичной системе счисления совпадает с прямым кодом. Для отрицательного числа все цифры числа заменяются на противоположные (1 на 0, 0 на 1), а в знаковый разряд заносится единица.

Для отрицательных чисел используется так называемый дополнительный код. Это связано с удобством выполнения операций над числами вычислительной техникой.

Дополнительный код используют в основном для представления в компьютере отрицательных чисел. Такой код делает арифметические операции более удобными для выполнения их вычислительной техникой.

В дополнительном коде, также как и прямом, первый разряд отводится для представления знака числа. Прямой и дополнительный код для положительных чисел совпадает. Поскольку прямой код используется почти исключительно для представления положительных чисел, а дополнительный – для отрицательных, то почти всегда, если в первом разряде 1, то мы имеем дело с дополнительным кодом. (Ноль обозначает положительное число, а единица – отрицательное).

Алгоритм получения дополнительного кода для отрицательного числа:

1. Найти прямой код числа (перевести число в двоичную систему счисления число без знака)
2. Получить обратный код. Поменять каждый ноль на единицу, а единицу на ноль (инвертировать число)
3. К обратному коду прибавить 1

Пример: Найдем дополнительный код десятичного числа – 47.

1. Найдем двоичную запись числа 47 (прямой код).

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

2. Инвертируем это число (обратный код).

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

3. Прибавим 1 к обратному коду и получим запись этого числа в оперативной памяти.

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

Почему же используется дополнительный код для представления отрицательного числа?

Так проще выполнять математические операции. Например, у нас два числа, представленных в прямом коде. Одно число положительное, другое – отрицательное и эти числа нужно сложить. Однако просто сложить их нельзя. Сначала компьютер должен определить, что это за числа. Выяснив, что одно число отрицательное, ему следует заменить операцию сложения операцией вычитания. Потом, машина должна определить, какое число больше по модулю, чтобы выяснить знак результата и определиться с тем, что из чего вычитать. В итоге, получается сложный алгоритм. Куда проще складывать числа, если отрицательные преобразованы в дополнительный код.

4. Решение примеров

Задание 1. Записать внутреннее представление следующих десятичных чисел, используя 8 -разрядную ячейку:

64_{10}

8 разрядное представление:

0	1	0	0	0	0	0	0
---	---	---	---	---	---	---	---

-120_{10}

8 разрядное представление:

Прямой код	0	1	1	1	1	0	0	0
Обратный код	1	0	0	0	0	1	1	1
Дополнительный код	1	0	0	0	1	0	0	0

Задание 2. Как запишутся в оперативной памяти компьютера следующие десятичные числа в 16-ти разрядной сетке

57_{10}

16 разрядное представление:

0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	1
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

200_{10}

16 разрядное представление:

0	0	0	0	0	0	0	0	1	1	0	0	1	0	0	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

-117_{10}

16-разрядное представление:

Прямой код	0	0	0	0	0	0	0	0	0	1	1	1	0	1	0	1
Обратный код	1	1	1	1	1	1	1	1	1	0	0	0	1	0	1	0
Дополнительный код	1	1	1	1	1	1	1	1	1	0	0	0	1	0	1	1

-200_{10}

16-разрядное представление:

Прямой код	0	0	0	0	0	0	0	0	1	1	0	0	1	0	0	0
Обратный код	1	1	1	1	1	1	1	1	0	0	1	1	0	1	1	1
Дополнительный код	1	1	1	1	1	1	1	1	0	0	1	1	1	0	0	0

5. Домашняя работа

Задание 1. Получить внутреннее представление целого числа 123_{10}
в 8-разрядной ячейке памяти компьютера.

Задание 2. Получить внутреннее представление целого числа
- 123_{10} в 8-разрядной ячейке памяти компьютера.

Задание 3. Получить внутреннее представление целого числа - 17_{10}
в 16 – ти разрядной ячейке памяти компьютера.