

1. **Записать число -3.5 в коротком/длинном/расширенном представлениях (в 16-чном виде).**

Сначала переведём данное число в двоичную систему счисления:

$3.5_{10} = 11.1_2$

Затем в *нормализованное* представление (сдвинем точку к самой левой (старшей) 1):

$11.1_2 = 1.11_2E1_2$

Вспомним, что:

- 1. знак итогового числа всегда определяется старшим битом;
- 2. порядок кодируется в *смещённом* виде, т.е.
  - 1 записывается как 01...10,
  - 0 записывается как 01...11,
  - 1 записывается как 10...00,
  - 2 записывается как 10...01 и т.д.;
- 3. мантисса записывается без старшего разряда, т.е. без “1.” (кроме расширенного представления).

Собирая всё это в единое целое и подставляя конкретное количество бит под порядок и мантиссу для конкретного представления, записываем:

- **Короткое** (float, 4 байта): порядок **8** бит, мантисса **23** бита:  

1 | 100 0000 0 | 110 0000 0000 0000 0000 0000<sub>2</sub> = **c0600000**<sub>16</sub>  
знак            порядок            мантисса
- **Длинное** (double, 8 байт): порядок **11** бит, мантисса **52** бита:  

1 | 100 0000 0000 | 1100 0000 ... 0000<sub>2</sub> = **c00c000000000000**<sub>16</sub>  
знак            порядок            мантисса
- **Расширенное** (long double, 10 байт): порядок **15** бит, мантисса **64** бита:  

1 | 100 0000 0000 0000 | 1110 0000 ... 0000<sub>2</sub> = **c000e000000000000000**<sub>16</sub>  
знак            порядок            мантисса

ОТВЕТЫ:  
**c0600000**  
**c00c000000000000**  
**c000e000000000000000**

2. **Дана последовательность байт: 00 00 A0 40**

**Какое вещественное число лежит в этих ячейках памяти?**

В архитектуре x86 используется порядок байт *little-endian*, т.е. обратный порядок, поэтому после чтения данных из памяти в регистр процессора получится:

$40A00000_{16}$

Т.к. данные занимают 4 байта и сказано, что это вещественное число, то это тип float, т.е. **короткое** представление. Для него один бит (старший) отводится на знак числа, 8 бит – на порядок (который кодируется в *смещённом* виде, где 01...1 = 0; 10...0 = 1; 10...1 = 2 и т.д.), остальные 23 бита на мантиссу. Запишем данное число в двоичной системе счисления:

0 | 100 0000 1 | 010 0000 0000 0000 0000 0000<sub>2</sub>  
знак            порядок            мантисса

Получаем  $+1.01_2E10_2 = 101_2 = 5_{10}$

Ответ: **5**

**Disclaimer!** Все мы знаем, что размеры типов в Си (в т.ч. float, double, long double) не фиксируются Стандартом, поэтому по умолчанию подразумеваются значения для компиляторов GNU GCC x86\_64.

Комментарии

Имя:

Комментарий:

Отправить