1) Изучите класс DoubleToBytes. Затем измените метод doubleToByteArray таким образом, чтобы ЛИБО использовать подход высокого уровня ИЛИ подход низкого уровня, НО без использования операторов сдвига. Какой бы подход вы ни выбрали, он должен оставить метод doubleToByteArray с точно такой же функциональностью.

2) При отправке многобайтовых типов данных обычно необходимо учитывать следующее. Архитектура ЦП либо little-endian, либо big-endian; некоторые современные процессоры позволяют выбирать через программное обеспечение. Предположим, что int со значением 0x12345678 хранится по адресу 0. Этот int будет храниться следующим образом:

Адрес 00 01 02 03

---------- - - - -

big-endian 12 34 56 78

little-endian 78 56 34 12

Стандарт протокола TCP / IP определяет, что все составляющие элемент байты должны отправляться в «сетевом порядке», который оказывается big-endian порядком байтов. Процессоры Intel Pentium - little-endian; многие другие процессоры - big-endian порядок байтов. Это означает, что на компьютере Intel ваше программное обеспечение обычно должно разбивать int на байты, а затем перевернуть байты перед их передачей.

Изучите класс DoubleToBytes. Затем ответьте на следующие два вопроса. Обратите внимание, что каждый ответ должен состоять только из одного предложения.

I. Ваш компьютер, вероятно, имеет процессор Intel. Но код Java в классе DoubleToBytes не должен был выполнять обращение байта. Почему нет?

II. Почему в методе byteArrayToDouble необходимо обнулить самые важные семь байтов long числа, в которое приведен каждый байт массива, перед тем, как сдвинуть его и связать его с другими для получения окончательного long результата?