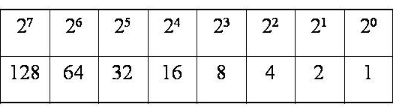
**IP-адрес** — это адрес, используемый узлом на сетевом уровне. Он имеет иерархическую структуру. Что это значит? Это значит, что каждая цифра в его написании несет определенный смысл. Примером будет номер обычного телефона — +74951234567. Первой цифрой идет +7. Это говорит о том, что номер принадлежит зоне РФ. Далее следует 495. Это код Москвы. Эти цифры закреплены за районной зоной. Как видите здесь наблюдается четкая иерархия. То есть по номеру можно понять какой стране, зоне он принадлежит. IP адреса придерживаются аналогично строгой иерархии. Контролирует их организация IANA (англ. Internet Assigned Numbers Authority). Если на русском, то это «Администрация адресного пространства Интернет». Заметьте, что слово «Интернет» с большой буквы. Мало кто придает этому значение. В англоязычной литературе термин «internet» используется для описания нескольких подключённых друг к другу сетей. А термин «Internet» для описания глобальной сети.

Не для кого, не секрет, что мы привыкли воспринимать числовую информацию в десятичном формате (в числах от 0-9). Однако все современные компьютеры воспринимают информацию в двоичном (0 и 1). Не важно при помощи тока или света передается информация. Вся она будет воспринята устройством как есть сигнал (1) или нет (0). Всего 2 значения. Поэтому был придуман алгоритм перевода из двоичной системы в десятичную, и обратно. IP адреса в десятичном формате выглядят как «193.233.44.12». Это и есть IP адрес в десятичной записи. Состоит он из 4-х чисел, называемых октетами и разделенных между собой точками. Каждое такое число (октет) может принимать значение от 0 до 255. То есть одно из 256 значений. Длина каждого октета равна 8 битам, а суммарная длина IPv4 = 32 битам. Теперь интересный вопрос. Каким образом этот адрес воспримет компьютер, и как будет с ним работать?

Можно конечно набить это в калькулятор, коих навалом в Интернете, и он переведет его в двоичный формат, но и переводить вручную должен уметь каждый. Особенно это касается тех, кто планирует сдать экзамен. У вас не будет под рукой ничего, кроме бумаги и маркера, и полагаться придется только на свои навыки. Поэтому пример, как это делать вручную. Строится таблица.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **128** | **64** | **32** | **16** | **8** | **4** | **2** | **1** |
| x | x | x | x | x | x | x | x |

Вместо «x» записывается либо 1, либо 0. Таблица разделена на 8 колонок, каждая из которых несет в себе 1 бит (8 колонок = 8 бит = 1 октет). Расположены они по старшинству слева направо. То есть первый (левый) бит — самый старший и имеет номер 128, а последний (правый) — самый младший и имеет номер 1. Теперь объясню, откуда эти числа взялись. Так как система двоичная, и длина октета равна 8-ми битам, то каждое число получается возведением числа 2 в степень от 0 до 7, и каждая из полученных цифр записывается в таблицу от большего к меньшему. То есть слева направо. От 2 в 7-ой степени до 2 в 0-ой степени. Таблица степеней 2-ки.



Теперь понятно, каким образом строится таблица. Давайте теперь разберем адрес «193.233.44.12» и посмотрим, как он выглядит в двоичном формате. Разберем каждый октет отдельно. Возьмем число 193 и посмотрим, из каких табличных комбинаций оно получается. 128 + 64 + 1 = 193.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **128** | **64** | **32** | **16** | **8** | **4** | **2** | **1** |
| 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 |

Те числа, которые участвовали в формировании комбинации получают 1, а все остальные получают 0.  
  
Берем первый октет 233. 128 + 64 + 32 + 8 + 1.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **128** | **64** | **32** | **16** | **8** | **4** | **2** | **1** |
| 1 | 1 | 1 | 0 | 1 | 0 | 0 | 1 |

Для 44 — это 32 + 8 + 4.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **128** | **64** | **32** | **16** | **8** | **4** | **2** | **1** |
| 0 | 0 | 1 | 0 | 1 | 1 | 0 | 0 |

И напоследок 12. 8 + 4.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **128** | **64** | **32** | **16** | **8** | **4** | **2** | **1** |
| 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |

Получается длинная битовая последовательность 11000001.11101001.00101100.00001100. Именно с данным видом работают сетевые устройства. Битовая последовательность обратима. Вы можете так же вставить каждый октет (по 8 символов) в таблицу и получить десятичную запись. Пусть это будет 11010101.10110100.11000001.00000011. Строю таблицу и заношу в нее первый блок.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **128** | **64** | **32** | **16** | **8** | **4** | **2** | **1** |
| 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 |

Получается 128 + 64 + 16 + 4 + 1 = 213.  
  
Вычисляю второй блок.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **128** | **64** | **32** | **16** | **8** | **4** | **2** | **1** |
| 1 | 0 | 1 | 1 | 0 | 1 | 0 | 0 |

Считаем 128 + 32 + 16 + 4 = 180.  
  
Третий блок.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **128** | **64** | **32** | **16** | **8** | **4** | **2** | **1** |
| 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 |

128 + 64 + 1 = 193.  
  
И напоследок четвертый.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **128** | **64** | **32** | **16** | **8** | **4** | **2** | **1** |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |

2 + 1 = 3

Собираем результаты вычислений и получаем адрес 213.180.193.3. Ничего тяжелого, чистая арифметика. Если тяжело и прям невыносимо трудно, то попрактикуйтесь.

**Задание (нужно решение, а не только ответы)**  
1) 10.124.56.220  
2) 113.72.101.11  
3) 173.143.32.194  
4) 200.69.139.217  
5) 88.212.236.76  
6) 01011101.10111011.01001000.00110000  
7) 01001000.10100011.00000100.10100001  
8) 00001111.11011001.11101000.11110101  
9) 01000101.00010100.00111011.01010000  
10) 00101011.11110011.10000010.00111101