

### Strim\_13

IP-адрес — это уникальный идентификатор устройства в сети Интернета. Он состоит из четырёх частей, называемых байтами. Каждый байт — это отдельная часть, которая составляет 8 бит. В сумме 32 бита дают 32-битный IP-адрес.

Байты — это отдельные части, которые обычно представлены как отдельные десятичные числа через точку.

Например:

10.34.188.35

8.8.8.8 – адрес DNS сервера

Каждое отделенное точкой число в IP-адресе обозначает 8 бит, записанных в двоичном виде:

192.168.0.1 = 11000000. 10101000.00000000.00000001

Если число в двоичном виде занимает меньше 8 позиций, слева дописываются до нужного количества ноли.

В задачах для перевода чисел IP – адреса в двоичный вид будем использовать f-строку, которая автоматически подставляет какое-то значение, если оно записано в фигурных скобках. Например:

```
f'{36:b}'
```

Результат:

100100

И кроме того, чтобы получить 8 цифр, дополним эту запись таким образом:

```
f'{36:08b}'
```

Результат:

00100100

Для получения полной двоичной записи IP-адреса (например, для IP 36.125.194.81) будем использовать в Python следующую строку:

```
print(f'{36:08b}', f'{125:08b}', f'{194:08b}', f'{81:08b}')
```

## Результат:

00100100 01111101 11000010 01010001

Таким образом мы получили 4 байта IP-адреса.

Байты IP адреса

192.168.0.1 = 11000000.10101000.00000000.00000001

Для администрирования IP-адреса разделяют на группы - сети (подсети). IP-сеть – это группа IP-адресов, определенных по особому принципу. В каждой IP-сети, некоторая начальная часть адресов должна совпадать. Т.е., если у IP-адресов одинаковая совпадающая часть, то они находятся внутри одной сети. Возникает вопрос, а какая часть адресов должна совпадать? Размер этой части определяется маской сети. Маска сети — это шаблон, который используется для определения границ сети. В этом шаблоне сначала идут только единицы, которые определяют адрес сети (постоянную часть IP-адреса), потом с какого-то момента только нули — её переменная часть. Маска может быть записана как IP-адрес, или количеством единиц. С помощью маски можно определить какая часть адресов IP-сети будет одинаковая, а какая будет отличаться.

Пример записи сети с маской:

нулевой адрес (адрес сети)      маска сети

Например, 192.168.0.0/255.255.255.240      или 192.168.0.0/28

Во втором варианте записи число 28 обозначает количество единиц, которое будет дополнено до 32 четырьмя нулями.

С помощью маски задается количество адресов в сети, так в этом примере в маске 28 единиц, 4 нуля, соответственно, мы получаем 24 адресов, 16 штук.

Маска, состоящая из 24 единиц и 8 нулей, даст количество возможных комбинаций адресов 28, т.е. 256 адресов, 16 единиц – 216, 0 единиц вся глобальная сеть. Т.е., чем больше в маске нулей, тем больше сеть.

Для решения задач нами будет использоваться свойство, заключающееся в том, что результат поразрядной конъюнкции маски сети с любым IP-адресом равен адресу сети. Т.е. адрес сети получается в результате поразрядной конъюнкции маски и IP-адреса.

Адрес сети = Маска & IP адрес

В любой сети самый первый адрес – адрес сети. Последний, в котором все единицы в переменной части, называется широковещательным. Это два особых адреса, которые используются для работы самой сети. И они не присваиваются отдельным узлам сети.

Адреса		Двоичная запись	Двоичная запись
255.255.255.240	11111111	11111111	11111111
192.168.0.0	11000000	10101000	00000000
192.168.0.1	11000000	10101000	00000001
192.168.0.2	11000000	10101000	00000010
192.168.0.3	11000000	10101000	00000011
192.168.0.4	11000000	10101000	00000100
192.168.0.5	11000000	10101000	00000101
192.168.0.6	11000000	10101000	00000110
192.168.0.7	11000000	10101000	00000111
192.168.0.8	11000000	10101000	00001000
192.168.0.9	11000000	10101000	00001001
192.168.0.10	11000000	10101000	00001010
192.168.0.11	11000000	10101000	00001011
192.168.0.12	11000000	10101000	00001100
192.168.0.13	11000000	10101000	00001101
192.168.0.14	11000000	10101000	00001110
192.168.0.15	11000000	10101000	00001111

16 адресов – это и есть список адресов сети с заданной маской.

Ссылка на видео-разбор с таймингом: [https://vk.com/video-205546952\\_456241205?t=1m45s](https://vk.com/video-205546952_456241205?t=1m45s)

## Задача № 1 (245)

По заданным IP-адресу узла сети и маске определите адрес сети:

IP-адрес: 217.8.244.3

Маска: 255.255.252.0

При записи ответа выберите из приведенных в таблице чисел 4 фрагмента четыре элемента IP-адреса и запишите в нужном порядке соответствующие им буквы без точек.

A	B	C	D	E	F	G	H
0	3	8	217	224	244	252	255

Ссылка на видео-разбор с таймингом: [https://vk.com/video-205546952\\_456241205?t=37m20s](https://vk.com/video-205546952_456241205?t=37m20s)

### Решение

Будем использовать свойство  $IP \& mask = Net$ , адрес сети равен поразрядной конъюнкции маски и IP-адреса.

Получим двоичную запись IP-адреса и маски сети:

```
print(f'{217:08b}', f'{8:08b}', f'{244:08b}', f'{3:08b}')
```

Результат:

11011001 00001000 11110100 00000011

```
print(f'{255:08b}', f'{255:08b}', f'{252:08b}', f'{0:08b}')
```

Результат:

11111111 11111111 11111100 00000000

Выполним поразрядное умножение полученных адресов:

```
IP   11011001 00001000 11110100 00000011
Mask 11111111 11111111 11111100 00000000
Net  11011001 00001000 11110100 00000000
```

Переведем полученную двоичную запись в десятичный вид:

11011001 00001000 11110100 00000000 = 217.8.244.0

Запишем соответствующие числам буквы без точек DCFA

Ответ: DCFA

## Задача №2 (250)

Для узла с IP-адресом 111.81.200.27 адрес сети равен 111.81.192.0. Чему равно наибольшее возможное значение третьего слева байта маски? Ответ запишите в виде десятичного числа.

Ссылка на видео-разбор с таймингом: [https://vk.com/video-205546952\\_456241205?t=45m45s](https://vk.com/video-205546952_456241205?t=45m45s)

### Решение

Получим двоичную запись IP-адреса и маски сети:

```
print(f'{111:08b}', f'{81:08b}', f'{200:08b}', f'{27:08b}')
```

### Результат:

01101111 01010001 11001000 00011011

Получим двоичную запись IP-сети:

```
print(f'{111:08b}', f'{81:08b}', f'{192:08b}', f'{0:08b}')
```

### Результат:

01101111 01010001 11000000 00000000

Подберём маску так, чтобы результат поразрядной конъюнкции IP-адреса и маски соответствовал адресу сети. Там, где на месте единицы в IP-адресе в адресе сети стоит единица, в маске сети также должна быть единица. Где на месте единицы в IP-адресе стоит ноль в адресе сети, в маске сети должен располагаться 0. Вспомним, что маска – это шаблон, в котором сначала идут все единицы, а потом идут все нули, учитывая эти свойства запишем те значения битов маски, которые являются очевидными исходя из этих её свойств.

IP	01101111	01010001	11001000	00011011
mask	11111111	11111111	11??0000	00000000
Net	01101111	01010001	11000000	00000000

Так как в задаче требуется подобрать маску с наибольшим возможным значением третьего слева байта маски, на месте битов с определяемым нами значением следует записать 1. Таким образом маска подсети будет иметь вид:

mask 11111111 11111111 11110000 00000000

Переведем в десятичный вид значение третьего слева байта маски:

```
int('11110000',2)
```

результат:

240

Ответ: 240

### Задача № 3 (10154)

В терминологии сетей TCP/IP маска сети – это двоичное число, меньшее 232; в маске сначала (в старших разрядах) стоят единицы, а затем с некоторого места нули. Маска определяет, какая часть IP-адреса узла сети относится к адресу сети, а какая – к адресу самого узла в этой сети. Обычно маска записывается по тем же правилам, что и IP-адрес – в виде четырёх байт, причём каждый байт записывается в виде десятичного числа. Адрес сети получается в результате применения поразрядной конъюнкции к заданному IP-адресу узла и маске.

Для узла с IP-адресом 148.195.140.28 адрес сети равен 148.195.140.0. Найдите наименьшее возможное количество единиц в двоичной записи маски подсети.

Ссылка на видео-разбор с таймингом: [https://vk.com/video-205546952\\_456241205?t=54m50s](https://vk.com/video-205546952_456241205?t=54m50s)

#### Решение

В Python переведем IP-адрес узла сети в двоичную форму:

```
print(f'{148:08b}', f'{195:08b}', f'{140:08b}', f'{28:08b}')
```

получаем 4 байта IP-адреса узла:

10010100 11000011 10001100 00011100

Выполним те же действия для IP адреса сети:

```
print(f'{1148:08b}', f'{195:08b}', f'{140:08b}', f'{0:08b}')
```

результат:

10010100 11000011 10001100 00000000

Подберём маску так, чтобы результат поразрядной конъюнкции IP-адреса и маски соответствовал адресу сети  $IP \& mask = Net$ :

IP	10010100	11000011	10001100	00011100
mask	11111111	11111111	111111??	???00000
Net	10010100	11000011	10001100	00000000

Там, где в IP - в адресе сети стоит единица, в маске, также, должна быть единица. В позициях нулей значения маски могли бы любыми, но т.к. в маске должны сначала следовать все единицы, заполним ими область вплоть до последней единицы в области адреса IP-сети. Далее рассмотрим IP – адрес узла сети. В той области, где в последнем байте располагаются единицы запишем в маске нули, т.к. нам нужно в результате поразрядной конъюнкции с битами в этой же области маски получить значения нулей в адресе сети. Т.к. после нулей уже не может быть единиц, допишем до конца маски оставшиеся нули, и рассмотрим возможные значения оставшихся неопределённых нами битов. Т.к. результат поразрядной конъюнкции независимо от их значения определён нулями в этой области IP-адреса сети, они могут иметь любые значения, но в задаче требуется подобрать маску с наименьшим возможным количеством единиц в двоичной записи, значит, в этих битах следует записать нули. Таким образом маска подсети будет иметь вид:

mask 11111111 11111111 11111100 00000000

Количество единиц такой маски равно 22.

Ответ: 22

#### Задача № 4 (10160)

В терминологии сетей TCP/IP маска сети – это двоичное число, меньшее 232; в маске сначала (в старших разрядах) стоят единицы, а затем с некоторого места нули. Маска определяет, какая часть IP-адреса узла сети относится к адресу сети, а какая – к адресу самого узла в этой сети. Обычно маска записывается по тем же правилам, что и IP-адрес – в виде четырёх байт, причём каждый байт

записывается в виде десятичного числа. Адрес сети получается в результате применения поразрядной конъюнкции к заданному IP-адресу узла и маске. Для узла с IP-адресом 76.155.48.2 адрес сети равен 76.155.48.0. Для скольких различных значений маски это возможно?

Ссылка на видео-разбор с таймингом: [https://vk.com/video-205546952\\_456241205?t=1h3m20s](https://vk.com/video-205546952_456241205?t=1h3m20s)

### Решение

В Python переведем IP-адрес узла в двоичную форму:

```
print(f'{76:08b}', f'{155:08b}', f'{48:08b}', f'{2:08b}')
```

получаем 4 байта IP-адреса узла:

01001100 10011011 00110000 00000010

Выполним те же действия для адреса сети:

```
print(f'{76:08b}', f'{155:08b}', f'{48:08b}', f'{0:08b}')
```

результат:

01001100 10011011 00110000 00000000

Подберём возможные варианты масок:

IP	01001100	10011011	00110000	00000010
mask	11111111	11111111	1111???	?????00
Net	01001100	10011011	00110000	00000000

Нужно подобрать возможные значения для битов в третьем и четвертом байте маски. Возможны следующие варианты:

11111111	11111100
11111111	11111000
11111111	11110000
11111111	11100000
11111111	11000000
11111111	10000000
11111111	00000000
11111110	00000000
11111100	00000000
11111000	00000000
11110000	00000000



Таких вариантов 11 штук.

Ответ: 11

### Задача №5 (10166)

В терминологии сетей TCP/IP маска сети – это двоичное число, меньшее 232; в маске сначала (в старших разрядах) стоят единицы, а затем с некоторого места нули. Маска определяет, какая часть IP-адреса узла сети относится к адресу сети, а какая – к адресу самого узла в этой сети. Обычно маска записывается по тем же правилам, что и IP-адрес – в виде четырёх байт, причём каждый байт записывается в виде десятичного числа. Адрес сети получается в результате применения поразрядной конъюнкции к заданному IP-адресу узла и маске.

Для некоторой подсети используется маска 255.255.254.0. Сколько различных адресов компьютеров теоретически допускает эта маска, если два адреса (адрес сети и широковещательный) не используют.

Ссылка на видео-разбор с таймингом: [https://vk.com/video-205546952\\_4562412051t=1h12m](https://vk.com/video-205546952_4562412051t=1h12m)

#### Решение

Переведём маску в двоичную форму:

```
print(f'{255:08b}', f'{255:08b}', f'{254:08b}', f'{0:08b}')
```

#### Результат:

11111111 11111111 11111110 00000000

С помощью маски можно определить, какая часть в адресах IP-сети будет одинаковая, а какая будет меняться. Т.е. для решения задачи следует определить количество комбинаций битов в области номера узла. Переменная часть по заданной в этой задаче маске имеет длину 9 бит, следовательно мы можем получить 29, т.е., 512 IP-адресов в сети. Т.к. два адреса (адрес сети и широковещательный) не используют могут быть получены 510 адресов.

Ответ: 510

### Задание №6 (10172)

В терминологии сетей TCP/IP маска сети – это двоичное число, меньшее 232; в маске сначала (в старших разрядах) стоят единицы, а затем с некоторого места нули. Маска определяет, какая часть IP-адреса узла сети относится к адресу сети, а какая – к адресу самого узла в этой сети. Обычно маска записывается по тем же правилам, что и IP-адрес – в виде четырёх байт, причём каждый байт записывается в виде десятичного числа. Адрес сети получается в результате применения поразрядной конъюнкции к заданному IP-адресу узла и маске.

Для узла с IP-адресом 175.122.80.13 адрес подсети равен 175.122.80.0. Сколько существует различных возможных значений маски, если известно, что в этой сети не менее 60 узлов. Ответ запишите в виде десятичного числа.

Ссылка на видео-разбор с таймингом: [https://vk.com/video-205546952\\_4562412051t=1h17m20s](https://vk.com/video-205546952_4562412051t=1h17m20s)

### Решение

Т.к. по условию задачи в сети не менее 60 узлов в ее маске должно быть не менее 6 нулей, т.к.  $2^6 = 64$  – минимальная степень двойки, которой будет достаточно, чтобы все адреса поместились

Переведём IP-адрес узла в двоичную форму:

```
print(f'{175:08b}', f'{122:08b}', f'{80:08b}', f'{13:08b}')
```

получаем 4 байта IP-адреса узла:

10101111 01111010 01010000 00001101

Выполним те же действия для адреса сети:

```
print(f'{175:08b}', f'{122:08b}', f'{80:08b}', f'{0:08b}')
```

результат:

10101111 01111010 01010000 00000000

Подберём возможные варианты масок, учитывая, что в маске должно быть не менее 6 нулей

00000000

	Мин. Единиц				Макс. единиц			
IP	10101111	01111010	01010000	00001101				
mask	11111111	11111111	1111	????	??	000000		
Net	10101111	01111010	01010000	00000000				

Следовательно, в маске может

быть от 20 до 26 единиц, т.е. может быть 7 разных вариантов масок.

Ответ: 7

## Задание № 7(36)

Если маска подсети 255.255.240.0 и IP –132.126.150.18 чему равен номер компьютера в сети?

Ссылка на видео-разбор с таймингом: [https://vk.com/video-205546952\\_4562412051t=1h25m55s](https://vk.com/video-205546952_4562412051t=1h25m55s)

### Решение

Получим двоичный вид адреса компьютера в сети

```
print(f'{132:08b}', f'{126:08b}', f'{150:08b}', f'{18:08b}')
```

10000100 01111110 10010110 00010010

Двоичный вид маски:

11110110 11111110 11110000 00000000

	Постоянная часть			Переменная часть				
IP	10000100	01111110	1001	0110	0001	0010	10	
mask	11110110	11111110	1111	0000	0000	0000	0000	

Порядковый номер компьютера

0110 00010010 число в десятичном виде 1554

Ответ: 1554

## Задание №8 (260)

Два узла, находящиеся в одной сети, имеют IP-адреса 211.115.61.154 и 211.115.59.137. Укажите наибольшее возможное значение третьего слева байта маски сети. Ответ запишите в виде десятичного числа.

Ссылка на видео-разбор с таймингом: [https://vk.com/video-205546952\\_4562412051t=1h32m30s](https://vk.com/video-205546952_4562412051t=1h32m30s)

### Решение

Переведём IP-адреса в двоичную форму. Адрес первого узла:

```
print(f'{211:08b}', f'{115:08b}', f'{61:08b}', f'{154:08b}')
```

11010011 01110011 00111101 10011010

Адрес второго узла:

```
print(f'{211:08b}', f'{115:08b}', f'{59:08b}', f'{137:08b}')
```

11010011 01110011 00111011 10001001

Надо так подобрать маску, чтобы там, где у неё находятся единицы, эти IP-адреса совпадали.

одинаковая постоянная часть

IP 1	11010011	01110011	00111101	10011010
IP 2	11010011	01110011	00111011	10001001
mask	11111111	11111111	11110000	00000000
net 1	11010011	01110011	00111000	00000000

Такой маской будет являться маска с третьим байтом такого вида 11111000 это наибольшее возможное значение. Найдем соответствующее ему десятичное число:

```
int('11111000',2)
```

Ответ: 248

## Задание №9

В терминологии сетей TCP/IP маской сети называют двоичное число, которое показывает, какая часть IP-адреса узла сети относится к адресу сети, а какая – к адресу узла в этой сети. Адрес сети получается в результате применения поразрядной конъюнкции к заданному адресу узла и его маске.

Сеть задана IP-адресом 112.160.0.0 и сетевой маской 255.240.0.0.

Сколько в этой сети IP-адресов, для которых количество единиц в двоичной записи IP-адреса кратно 5?

В ответе укажите только число.

Ссылка на видео-разбор с таймингом: [https://vk.com/video-205546952\\_4562412051t=1h41m30s](https://vk.com/video-205546952_4562412051t=1h41m30s)

*Решение*

Переведём IP-адрес и маску сети в двоичную форму.

IP-адрес сети:

```
print(f'{112:08b}', f'{160:08b}', f'{0:08b}', f'{0:08b}')
```

01110000 10100000 00000000 00000000

маска:

```
print(f'{255:08b}', f'{240:08b}', f'{0:08b}', f'{0:08b}')
```

11111111 11110000 00000000 00000000

IP	01110000	10100000	00000000	00000000
mask	11111111	11110000	00000000	00000000

постоянная часть

переменная часть IP адреса -20 битов

По условию задачи количество единиц суммарно во всех байтах должно быть кратно 5. При этом в постоянной части IP-адреса находится 5 единиц, следовательно, в переменной части по условию задачи могут находиться 0, 5, 10, 15, 20 единиц. Далее нам нужно сосчитать все варианты комбинаций записи 20 битов, в которых будет такое количество единиц.

0 единиц – 1 адрес

20 единиц – 1 адрес

Далее для подсчёта воспользуемся комбинаторными формулами подсчета перестановок:

$$\begin{aligned} 5 \text{ единиц} &- \frac{20 \cdot 19 \cdot 18 \cdot 17 \cdot 16}{5!} = 15504 \\ 10 \text{ единиц} &- \frac{20 \cdot 19 \cdot 18 \cdot 17 \cdot 16 \cdot 15 \cdot 14 \cdot 13 \cdot 12 \cdot 11}{10!} = 184756 \\ 15 \text{ единиц} &- 15504 \end{aligned}$$

Окончательный ответ:

$15504 \cdot 2 + 184756 + 2 = 215766$  вариантов IP-адресов сети, для которых количество единиц в двоичной записи IP-адреса кратно 5.

Ответ: 215766

## Задание №10

В терминологии сетей TCP/IP маской сети называют двоичное число, которое показывает, какая часть IP-адреса узла сети относится к адресу сети, а какая – к

адресу узла в этой сети. Адрес сети получается в результате применения поразрядной конъюнкции к заданному адресу узла и маске сети.

Сеть задана IP-адресом 192.168.32.160 и маской сети 255.255.255.240.

Сколько в этой сети IP-адресов, для которых количество нулей в двоичной записи IP-адреса больше 21?

В ответе укажите только число.

Ссылка на видео-разбор с таймингом: [https://vk.com/video-205546952\\_4562412051t=1h53m30s](https://vk.com/video-205546952_4562412051t=1h53m30s)

*Решение*

Переведём IP-адрес и маску сети в двоичную форму.

IP-адрес сети:

```
print(f'{192:08b}', f'{168:08b}', f'{32:08b}', f'{160:08b}')
```

00001010 00110000 01100000 00000000

маска:

```
print(f'{255:08b}', f'{255:08b}', f'{255:08b}', f'{240:08b}')
```

11111111 11111111 11111111 11110000

Полученный адрес сети и маска описывают определенный набор IP-адресов.

В постоянной части IP адреса 8 единиц и 20 нулей.

Т.к. по условию задачи количество нулей в двоичной записи IP-адреса должно быть больше 21, в переменной части должно быть 2, 3 или 4 нуля. Для получения ответа нам необходимо сосчитать возможные комбинации таких вариантов.

Варианты:	сумма нулей
-----------	-------------

- |    |      |   |
|----|------|---|
| 1. | 0000 | 4 |
| 2. | 0001 | 3 |
| 3. | 0010 | 3 |
| 4. | 0011 | 2 |
| 5. | 0100 | 3 |
| 6. | 0101 | 2 |

- |     |      |   |
|-----|------|---|
| 7.  | 0110 | 2 |
| 8.  | 1000 | 3 |
| 9.  | 1001 | 2 |
| 10. | 1010 | 2 |
| 11. | 1100 | 2 |
| 12. | 1011 |   |
| 13. | 0111 |   |
| 14. | 1111 |   |
| 15. | 1101 |   |
| 16. | 1110 |   |

общее количество – 11.

Ответ : 11