

DZ_131_1

Задача № 1 (10149)

(Уровень: Базовый)

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.

Например, если IP-адрес узла равен 131.32.255.131, а маска равна 255.255.240.0, то адрес сети равен 131.32.240.0.

Для узла с IP-адресом 220.128.112.142 адрес сети равен 220.128.96.0. Чему равен третий слева байт маски? Ответ запишите в виде десятичного числа.

Ссылка на видео-разбор с таймингом: https://vk.com/video-205546952_456241206?t=0h0m0s

Решение

Нам дан IP-адрес, и мы знаем, что его поразрядная конъюнкция с маской должна равняться адресу сети. Это основное свойство, на которое опирается решение задачи. Переведем IP-адрес и адрес сети, их третьи байты в двоичный 8-битный вид. Выполним это действие в Python, используя f-строку:

Для IP-адрес

```
f'{112:08b}'
```

результат:

01110000

Для сети

```
f'{96:08b}'
```

результат:

01100000

Подберём маску так, чтобы результат поразрядной конъюнкции, была IP-адреса и маски соответствовал адресу сети.

IP 01110000

&

mask 11110000

Net 01100000

Переведём полученную маску в десятичный вид:

```
int('11100000',2)
```

результат:

224

Ответ: 224

Telegram: @fast_ege

DZ_131_2

Задача №2 (10153)

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 115.12.69.38 адрес сети равен 115.12.64.0. Найдите наименьшее возможное количество единиц в двоичной записи маски подсети.

Ссылка на видео-разбор с таймингом: https://vk.com/video-205546952_456241206?t=0h3m45s

Решение

В Python переведём IP-адрес в двоичную форму:

```
print(*[f'{x:08b}' for x in (115,12,69,38)])
```

получаем 4 байта IP-адреса:

01110011 00001100 01000101 00100110

Выполним те же действия для адреса сети:

```
print(*[f'{x:08b}' for x in (115,12,64,0)])
```

результат:

01110011 00001100 01000000 00000000

Подберём вариант маски:

IP	01110011	00001100	01000101	00100110
mask	11111111	11111111	11???000	00000000
Net	01110011	00001100	01000000	00000000

Там, где на месте единицы в IP-адреса 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 быть 0.

Так как в задаче требуется подобрать маску с наименьшим возможным количеством единиц в двоичной записи, на месте битов с определяемым нами значением следует записать нули. Таким образом маска подсети будет иметь вид:

Количество единиц такой маски равно 18.

Ответ: 18

Telegram: @fast_ege

DZ_131_3

Задача № 3 (10158)

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 204.108.112.142 адрес сети равен 204.108.64.0. Найдите наибольшее возможное количество нулей в двоичной записи маски подсети

Ссылка на видео-разбор с таймингом: https://vk.com/video-205546952_456241206?t=0h7m5s

Решение

В Python переведём IP-адрес в двоичную форму:

```
print(*[f'{x:08b}' for x in (204,108,112,142)])
```

получаем 4 байта IP-адреса:

11001100 01101100 01110000 10001110

Выполним те же действия для адреса сети:

```
print(*[f'{x:08b}' for x in (204,108,64,0)])
```

результат:

11001100 01101100 01000000 00000000

Подберём вариант маски:

IP	11001100	01101100	01110000	10001110
mask	11111111	11111111	11000000	00000000
Net	11001100	01101100	01000000	00000000

IP	11010000	11001111	11100110	01000001
mask	11111111	11111111	111?000	00000000
Net	11010000	11001111	11100000	00000000

Там, где на месте единицы в IP-адреса 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 быть 0.

В данном случае это единственный вариант маски.

Ответ: 14

Telegram: @fast_ege

DZ_131_4

Задача № 4 (10159)

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 208.207.230.65 адрес сети равен 208.207.224.0. Для скольких различных значений маски это возможно?

Ссылка на видео-разбор с таймингом: https://vk.com/video-205546952_456241206?t=0h9m20s

Решение

В Python переведём IP-адрес в двоичную форму:

```
print(*[f'{x:08b}' for x in (208,207,230,65)])
```

получаем 4 байта IP-адреса:

11010000 11001111 11100110 01000001

Выполним те же действия для адреса сети:

```
print(*[f'{x:08b}' for x in (208,207,224,0)])
```

результат:

11010000 11001111 11100000 00000000

Подберём возможные варианты масок:

IP	11010000	11001111	11100110	01000001
mask	11111111	11111111	111??000	00000000
Net	11010000	11001111	11100000	00000000

Нужно подобрать возможные значения для двух битов третьего байта маски. Возможны следующие варианты:

11100000
11110000
11111000

Ответ: 3

Telegram: @fast_ege

Задание №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_456241206?t=0h11m30s

Решение

Найдем двоичный вид маски:

```
print(*[f'{x:08b}' for x in (255,255,254,0)])
```

результат:

11111111 11111111 11111110 00000000

Как мы знаем, маска делит любые адреса на две части. Первая и постоянная часть маски IP-адресов, принадлежащих одной сети, состоит из единиц до первого нуля в высшем разряде, и у всех IP-адресов, принадлежащих одной сети будет одинаковой. Вторая переменная часть маски отсчитывается от этого первого нуля в высшем разряде и различна у каждого IP-адреса. Возможное количество адресов в сети определяется частью с нулями. Чем их больше, тем больше адресов может быть в сети.

Следовательно, переменная часть IP-адресов в сети с такой маской состоит из 9 битов. Количество различных IP-адресов в такой сети $2^9 = 512$, но так как по условию два адреса (адрес сети и широковещательный) не используют, всего возможно $512 - 2 = 510$ адресов.

Ответ: 510

Telegram: @fast_ege

Задание №6(10169)

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.

Два узла, находящиеся в разных подсетях, имеют IP-адреса 157.127.182.76 и 157.127.190.80. В масках обеих подсетей одинаковое количество единиц.

Укажите наименьшее возможное количество единиц в масках этих подсетей.

Ссылка на видео-разбор с таймингом: https://vk.com/video-205546952_456241206?t=0h13m30s

Решение

Т.к. по условию задачи узлы находятся в разных сетях, постоянная часть IP-адресов у них должна быть различной (соответствующая единицам в маске).

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

```
print(*[f'{x:08b}' for x in (157,127,182,76)])
```

```
10011101 01111111 10110110 01001100
```

Адрес второго узла:

```
print(*[f'{x:08b}' for x in (157,127,190,80)])
```

```
10011101 01111111 10111110 01010000
```

IP 1	10011101	01111111	10110110	01001100
IP 2	10011101	01111111	10111110	01010000
mask	11111111	11111111	1111?	

Обратим внимание на то, что начало адресов этих двух узлов совпадают до 4 бита в третьем байте (отсчёт битов ведётся справа налево и начинается с 0), следовательно, до этого бита в масках обоих адресов однозначно будут находиться единицы (20 шт.). Если в маске оставшиеся биты заполнить нулями:

IP 1	10011101	01111111	10110110	01001100
IP 2	10011101	01111111	10111110	01010000
mask	11111111	11111111	11110000	00000000
net	10011101	01111111	10110000	00000000

то третий байт адреса для этих узлов будет одинаковым и IP-адреса будут принадлежать одной сети.

Если в маске в пятом бите третьего байта будет единица:

```
IP 1    10011101 01111111 10110110 01001100
IP 2    10011101 01111111 10111110 01010000
mask    11111111 11111111 11111000 00000000
net 1    10011101 01111111 10110000 00000000
net 2    10011101 01111111 10111000 00000000
```

то значения в третьего байта в адресах сетей у данных IP-адресов будут разные, таким образом, наименьшее возможное количество единиц в масках этих подсетей будет равно 21.

Ответ: 21

Telegram: @fast_ege

DZ_131_7

Задание №7 (10162)

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.

Два узла, находящиеся в одной сети, имеют IP-адреса 112.117.107.70 и 112.117.121.80. 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_456241206?t=0h16m40s

Решение

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

```
print(*[f'{x:08b}' for x in (112,117,107,70)])
```

```
01110000 01110101 01101011 01000110
```

Адрес второго узла:

```
print(*[f'{x:08b}' for x in (112,117,121,80)])
```

```
01110000 01110101 01111001 01010000
```


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

```
IP 1    01110000 01110101 01101011 01000110
IP 2    01110000 01110101 01111001 01010000
```

```
mask    11111111 11111111 11100000 00000000
net 1    10011101 01111111 10100000 00000000
```

Такой маской будет являться маска с третьим байтом такого вида 11100000 это наибольшее возможное значение. Найдём соответствующее ему десятичное число:

```
int('11100000',2)
```

Ответ: 224

Telegram: @fast_ege

DZ_131_8

Задание №8(10783)

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.

Например, если IP-адрес узла равен 131.32.255.131, а маска равна 255.255.240.0, то адрес сети равен 131.32.240.0.

Два узла, находящиеся в одной сети, имеют IP-адреса 121.171.5.70 и 121.171.5.107. Укажите наименьшее возможное количество адресов в этой сети.

Ссылка на видео-разбор с таймингом: https://vk.com/video-205546952_456241206?t=0h19m15s

Решение

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

```
print(*[f'{x:08b}' for x in (121,171,5,70)])
```

01111001 10101011 00000101 01000110

Адрес второго узла:

```
print(*[f'{x:08b}' for x in (121,171,5,107)])
```

01111001 10101011 00000101 01101011

Т.к. узлы находятся в одной сети, маску следует подобрать так, чтобы постоянная часть адресов, где у маски находятся единицы, совпадала.

IP 1	01111001	10101011	00000101	01000110
IP 2	01111001	10101011	00000101	01101011
mask	11111111	11111111	11111111	11000000

Если вместо хотя бы одного следом идущим за единицами взять 0, адреса сети будут различаться и, следовательно, IP-адреса будут принадлежать разным сетям. Таким образом, в маске должно быть не менее 6 нулей, количество адресов в сети $2^6 = 64$

Ответ: 64

Telegram: @fast_ege

DZ_131_9

Задание №9 (10171)

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 115.53.128.88 адрес подсети равен 115.53.128.0. Сколько существует различных возможных значений третьего слева байта маски, если известно, что в этой сети не менее 1000 узлов? Ответ запишите в виде десятичного числа.

Ссылка на видео-разбор с таймингом: https://vk.com/video-205546952_456241206?t=0h21m25s

Решение

Условие наличия в заданной сети не менее 1000 узлов другими словами обозначает, что в маске должно быть не менее 10 нулей, т.к. 10 нулей обеспечат возможность существования 1024 адресов.

Переведём IP-адрес и адрес подсети в двоичную форму.

IP-адрес:

```
print(*[f'{x:08b}' for x in (115,53,128,88)])
```

01110011 00110101 10000000 01011000

Адрес сети:

```
print(*[f'{x:08b}' for x in (115,53,128,0)])
```

01110011 00110101 10000000 00000000

Подберем маску, исходя из заданного условия наличия в ней не менее 10 нулей:

IP	10011101	01111111	10111110	01010000
mask	11111111	11111111	1?????00	00000000
net	10011101	01111111	10110000	00000000

Таким образом, возможные варианты третьего байта маски могут быть

10000000
11000000
11100000
11110000
11111100
11111100

Их количество равно 6

Ответ : 6

Telegram: @fast_ege

DZ_131_10

Задача № 10 (11662)

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 123.222.111.192 и маской сети 255.255.255.248. Сколько в этой сети IP-адресов, для которых сумма единиц в двоичной записи четвёртого байта IP-адреса не делится без остатка на 3? В ответе укажите только число.

Ссылка на видео-разбор с таймингом: https://vk.com/video-205546952_456241206?t=0h24m35s

Решение

Переведём IP-адрес и маску сети в двоичную форму.

IP-адрес сети:

```
print(*[f'{x:08b}' for x in (123,222,111,192)])
```

01111011 11011110 01101111 11000000

маска:

```
print(*[f'{x:08b}' for x in (255,255,255,248)])
```

11111111 11111111 11111111 11111000

Рассмотрим четвертые байты IP-адреса сети и маски:

11000000

11111000

И определим все возможные варианты всех IP-адресов сети:

Варианты:	сумма единиц
-----------	--------------

11000000	2
----------	---

11000001	3
----------	---

11000010	3
----------	---

11000011	4
----------	---

11000100	3
----------	---

11000101	4
----------	---

11000110	4
----------	---

11000111	5
----------	---

Таким образом, количество сумм единиц в двоичной записи четвёртого байта IP-адреса не делящихся без остатка на 3 равно 5

Ответ: 5

Telegram: @fast_ege

Задача № 11 (10580)

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 10.48.96.0 и маской сети 255.255.240.0. Сколько в этой сети IP-адресов, для которых количество единиц в двоичной записи IP-адреса больше, чем количество нулей?

В ответе укажите только число.

Ссылка на видео-разбор с таймингом: https://vk.com/video-205546952_456241206?t=0h27m25s

Решение

Переведём IP-адрес и маску сети в двоичную форму.

IP-адрес сети:

```
print(*[f'{x:08b}' for x in (10,48,96,0)])
```

00001010 00110000 01100000 00000000

маска:

```
print(*[f'{x:08b}' for x in (255, 255, 240, 0)])
```

11111111 11111111 11110000 00000000

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

IP	00001010	00110000	01100000	00000000
mask	11111111	11111111	11110000	00000000

постоянная часть

переменная часть IP адреса -12 битов

В постоянной части IP адреса 6 единиц и 14 нулей.

Т.к. по условию задачи количество единиц в двоичной записи IP-адреса должно превышать количество нулей, переменная часть IP адреса должна содержать на 9 единиц больше, чем нулей.

Возможные варианты:

6 единиц (постоянная часть) + 12 единиц (все биты в переменной части)

Общее количество единиц $18 > 14$ общее количества нулей

6 единиц (постоянная часть) + 11 единиц (11 битов в переменной части и один 0)

Общее количество единиц $17 > 15$ ($14 + 1$) общего количества нулей

6 единиц (постоянная часть) + 10 единиц (10 битов в переменной части и два 0)

Общее количество единиц $16 = 16$ ($14 + 2$) общего количества нулей – по условию задачи этот вариант уже не подходит.

Следовательно, для подсчета количества IP-адресов сети, для которых количество единиц в двоичной записи IP-адреса больше, чем количество нулей необходимо сосчитать возможные комбинации этих двух вариантов.

Случай, когда все 12 битов заняты единицами – дает единственный вариант, а в случае, когда среди 12 битов может быть один ноль – 12 вариантов. Их общее количество – 13.

Ответ:13

Telegram: @fast_ege

DZ_131_12

Задача № 12 (16379)

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 112.208.0.0 и сетевой маской 255.255.128.0.

Сколько в этой сети IP-адресов, для которых количество единиц в двоичной записи IP-адреса кратно 11? В ответе укажите только число.

Ссылка на видео-разбор с таймингом: https://vk.com/video-205546952_456241206?t=0h32m30s

Решение

Переведём IP-адрес и маску сети в двоичную форму.

IP-адрес сети:

```
print(*[f'{x:08b}' for x in (112,208,0,0)])
```

01110000 11010000 00000000 00000000

маска:

```
print(*[f'{x:08b}' for x in (255, 255, 128, 0)])
```

11111111 11111111 10000000 00000000

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

IP 01110000 11010000 00000000 00000000

mask 11111111 11111111 10000000 00000000

постоянная часть

переменная часть IP адреса -15 битов

По условию задачи количество единиц суммарно во всех байтах должно быть кратно 11. При этом в постоянной части IP-адреса находится 6 единиц, следовательно, подходит единственный вариант, когда в переменной части будет 5 единиц. Далее нам нужно сосчитать все варианты комбинаций записи 15 битов, пять из которых являются единицами. Воспользуемся комбинаторной формулой:

$$\frac{15 * 14 * 13 * 12 * 11}{5!}$$

Получаем ответ 3003 вариантов IP-адресов сети, для которых количество единиц в двоичной записи IP-адреса кратно 11.

Ответ: 3003

Telegram: @fast_ege