

DZ_132_1

Задача № 1 (10152)

В терминологии сетей TCP/IP маска сети – это двоичное число, меньшее 2^{32} ; 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 215.181.200.27 адрес сети равен 215.181.192.0. Чему равно наибольшее возможное значение третьего слева байта маски? Ответ запишите в виде десятичного числа.

Показать ответ

Ссылка на видео-разбор с таймингом: https://vk.com/video-205546952_456241208?t=0h0m0s

Решение

Для решения этой и последующих задач этого домашнего задания будем использовать модуль «ipaddress». Переберем разные варианты маски, как количество единиц от 0 до 32, и на основании этого сделаем сеть. Создадим объект сети «ip network». В качестве адреса в используем 215.181.200.27 и подставим маску, но т.к. это не адрес сети (это просто IP-адрес), мы укажем параметр «0», и выведем на экран полученные адреса сети. Используем метод «netmask», чтобы вывести значение маски в десятичном виде отдельным полем.

```
#модуль для создания адресов и сетей и работы с ними
from ipaddress import *
#перебор вариантов масок, как количества единиц
for mask in range(33):
#создадим объект сети
    net = ip_network(f'215.181.200.27/{mask}', 0)
```

```
#выведем адреса сети и значения масок в десятичном виде
print(net, net.netmask)
```

Результат работы программы:

```
0.0.0.0/0 0.0.0.0
128.0.0.0/1 128.0.0.0
192.0.0.0/2 192.0.0.0
192.0.0.0/3 224.0.0.0
...
215.180.0.0/14 255.252.0.0
215.180.0.0/15 255.254.0.0
215.181.0.0/16 255.255.0.0
215.181.128.0/17 255.255.128.0
215.181.192.0/18 255.255.192.0
215.181.192.0/19 255.255.224.0
215.181.192.0/20 255.255.240.0
215.181.200.0/21 255.255.248.0
215.181.200.0/22 255.255.252.0
215.181.200.0/23 255.255.254.0
...
```

Нам нужен адрес 215.181.192.0. Этот адрес есть в трёх масках. При этом наибольшее значение третьего байта маски – это 240.

Ответ: 240

Telegram: @fast__ege

DZ_132_2

Задача №2 (10155)

В терминологии сетей TCP/IP маска сети – это двоичное число, меньшее 2^{32} ; 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 142.198.113.106 адрес сети равен 142.198.112.0. Найдите наибольшее возможное количество единиц в двоичной записи маски подсети.

Ссылка на видео-разбор с таймингом: https://vk.com/video-205546952_456241208?t=0h3m0s

Решение

Решение этой задачи очень похоже на решение предыдущей. Отличием является то, что здесь не нужно выводить запись маски, а нужно определить маску с наибольшим количеством единиц в ней.

```
from ipaddress import *  
  
for mask in range(33):  
    net = ip_network(f'142.198.113.106/{mask}', 0)  
    print(net)
```

Результат работы программы:

```
0.0.0.0/0  
128.0.0.0/1  
...  
142.198.112.0/20  
142.198.112.0/21  
142.198.112.0/22  
142.198.112.0/23  
142.198.113.0/24  
...
```

Адресу 142.198.112.0 соответствуют 4 маски. Маска с наибольшим количеством единиц это 23

Ответ: 23

Telegram: @fast_ege

DZ_132_3

Задача № 3 (10157)

В терминологии сетей TCP/IP маска сети – это двоичное число, меньшее 2^{32} ; 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 241.185.253.57 адрес сети равен 241.185.252.0. Найдите наименьшее возможное количество нулей в двоичной записи маски подсети.

Ссылка на видео-разбор с таймингом: https://vk.com/video-205546952_456241208?t=0h4m25s

Решение

Т.к. в задаче наименьшее возможное количество нулей в двоичной записи маски, в выводе (32– mask), т.е. количество единиц в маске.

```
from ipaddress import *
for mask in range(33):
    net = ip_network(f'241.185.253.57/{mask}', 0)
    print(net, 32 - mask)
```

Результат работы программы:

```
0.0.0.0/0 32
...
241.185.248.0/21 11
241.185.252.0/22 10
241.185.252.0/23 9
241.185.253.0/24 8
241.185.253.0/25 7
241.185.253.0/26 6
...
```

Маску, при которых адрес сети 241.185.252.0 содержат нулей 10 и 9.
Наименьшим количеством является 9.

Ответ: 9

Telegram: @fast_ege

DZ_132_4

Задача № 4 (10167)

В терминологии сетей TCP/IP маска сети – это двоичное число, меньшее 2^{32} ; 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 108.133.75.91 адрес сети равен 108.133.75.64. Чему равно наибольшее количество возможных адресов в этой сети?

Ссылка на видео-разбор с таймингом: https://vk.com/video-205546952_456241208?t=0h6m25s

Решение

Количество возможных адресов в сети связано с количеством нулей в маске этой сети, и может быть вычислено, как 2 в степени количество нулей.

```
from ipaddress import *
for mask in range(33):
    net = ip_network(f'108.133.75.91/{mask}', 0)
    print(net, 2**(32-mask))
```

Результат работы программы:

```
...
108.133.75.0/24 256
108.133.75.0/25 128
108.133.75.64/26 64
108.133.75.64/27 32
108.133.75.80/28 16
108.133.75.88/29 8
108.133.75.88/30 4
108.133.75.90/31 2
108.133.75.91/32 1
```

Две маски дают указанный в задаче адрес сети 108.133.75.64, при этих масках 64, 32 адреса в сети соответственно. Т.к. требуется наибольшее количество возможных адресов ответом будет 64.

Ответ: 64

Telegram: @fast_ege

DZ_132_5

Задача №5 (10169)

В терминологии сетей TCP/IP маска сети – это двоичное число, меньшее 2^{32} ; 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_456241208?t=0h8m30s

Решение

Для решения будем переберем разные варианты масок для двух указанных подсетей. В качестве адреса одной из них будет использоваться первый IP-адрес, для второй используем второй IP-адрес.

```
from ipaddress import *

for mask in range(33):
    net1 = ip_network(f'157.127.182.76/{mask}', 0)
    net2 = ip_network(f'157.127.190.80/{mask}', 0)
    if net1 != net2:
        print(net1, net2)
```

Результат работы программы:

```
157.127.176.0/21 157.127.184.0/21
157.127.180.0/22 157.127.188.0/22
157.127.182.0/23 157.127.190.0/23
157.127.182.0/24 157.127.190.0/24
157.127.182.0/25 157.127.190.0/25
157.127.182.64/26 157.127.190.64/26
157.127.182.64/27 157.127.190.64/27
157.127.182.64/28 157.127.190.80/28
157.127.182.72/29 157.127.190.80/29
157.127.182.76/30 157.127.190.80/30
157.127.182.76/31 157.127.190.80/31
157.127.182.76/32 157.127.190.80/32
```

В результате работы программы получены пары различающихся адресов подсетей, но при этом с одинаковым количеством единиц. Наименьшее одинаковое количество единиц в обеих сетях равняется 21.

Ответ: 21

Telegram: @fast_ege

DZ_132_6

Задание №6 (10172)

В терминологии сетей TCP/IP маска сети – это двоичное число, меньшее 2^{32} ; 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_456241208?t=0h10m50s](https://vk.com/video-205546952_456241208?t=0h10m50s)

Решение

Выведем на экран адреса подсетей, а также количество возможных узлов в них, вспомним, что это число связано с количеством нулей в маске и вычисляется, как $2^{32 - \text{mask}}$

```
from ipaddress import *
for mask in range(33):
    net = ip_network(f'175.122.80.13/{mask}', 0)
    print(net, 2**(32-mask))
```

Результат работы программы:

```
0.0.0.0/0 4294967296
...
175.122.64.0/18 16384
```

175.122.64.0/19 8192
175.122.80.0/20 4096
175.122.80.0/21 2048
175.122.80.0/22 1024
175.122.80.0/23 512
175.122.80.0/24 256
175.122.80.0/25 128
175.122.80.0/26 64
175.122.80.0/27 32
175.122.80.0/28 16
175.122.80.8/29 8
175.122.80.12/30 4
175.122.80.12/31 2
175.122.80.13/32 1

Мы видим, что вот довольно много масок дают нужный нам адрес, но, по условию задачи в этой сети должно быть не менее 60 узлов, и таких вариантов здесь 7.

Ответ: 7

Telegram: @fast_ege

DZ_132_7

Задание № 7(17867)

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 172.16.168.0 и маской сети 255.255.248.0. Сколько в этой сети IP-адресов, для которых количество единиц в двоичной записи IP-адреса не кратно 5? В ответе укажите только число

Ссылка на видео-разбор с таймингом: https://vk.com/video-205546952_456241208?t=0h13m25s

Решение

Для ответа на вопрос задачи следует сосчитать количество адресов, в которых количество единиц двоичной записи, не кратно 5. Для этого создадим счетчик для подсчета таких значений

```
from ipaddress import *
```



```
net = ip_network('172.16.168.0/255.255.248.0')
k = 0
#получим двоичную запись для каждого значения ip-адреса сети
for ip in net:
    b = f'{ip:b}'
#проверим заданное условие
    if b.count('1')%5!=0:
        k += 1
print(k)
```

Результат работы программы:

1663

Ответ: 1663

Telegram: @fast_ege

DZ_132_8

Задание №8 (17710)

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 214.187.224.0 и сетевой маской 255.255.224.0. Сколько в этой сети IP-адресов, для которых количество единиц в двоичной записи IP-адреса не кратно 6, а сами адреса в двоичном виде заканчиваются на 1000? В ответе укажите только число.

Ссылка на видео-разбор с таймингом: https://vk.com/video-205546952_456241208?t=0h14m50s

Решение

```
from ipaddress import *
net = ip_network('214.187.224.0/255.255.224.0')
k = 0
#получим двоичную запись для каждого значения ip-адреса сети
for ip in net:
    b = f'{ip:b}'
#проверим заданное условие
    if b.count('1')%6!=0 and b[-4]+b[-3]+b[-2]+b[-1]=='1000':
```

```
k += 1
print(k)
```

Результат работы программы:

427

Ответ: 427

Telegram: @fast_ege

DZ_132_9

Задание №9(12947)

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, в которой содержится узел с IP-адресом 203.111.195.0, задана маской сети 255.255.240.0. Сколько в этой сети IP-адресов, в двоичной записи которых количество нулей кратно трём, а также содержатся три подряд идущие единицы и три подряд идущих нуля одновременно?

В ответе укажите только число.

Ссылка на видео-разбор с таймингом: https://vk.com/video-205546952_456241208?t=0h16m55s

Решение

```
from ipaddress import *
net = ip_network('203.111.195.0/255.255.240.0', 0)
k = 0
#получим двоичную запись для каждого значения ip-адреса сети
for ip in net:
    b = f'{ip:b}'
    #проверим заданное условие
    if b.count('0') % 3 == 0 and '111' in b and '000' in b:
        k += 1
print(k)
```

Результат работы программы:

1043

Ответ: 1043

DZ_132_10

Задание №10 (11835)

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, в которой содержится узел с IP-адресом 207.0.A.167, задана маской сети 255.255.255.192, где A — некоторое допустимое для записи IP-адреса число.

Определите количество значений A, для которых для всех IP-адресов этой сети в двоичной записи IP-адреса суммарное количество нулей в левых двух байтах больше суммарного количества нулей в правых двух байтах.

В ответе укажите только число.

Ссылка на видео-разбор с таймингом: https://vk.com/video-205546952_456241208?t=0h19m5s

Решение

По условию задачи переменная A — это часть IP-адреса, т.е. число от 0 до 255, перебрав все допустимые значения a, подставляя их в цикле в третий байт ip — адреса, определим подходящие под условие задачи значения. Для этого, используем срезы [:16], [16:] и метод «count», сосчитаем количество нулей, в первых двух байтах слева, 16 цифрах ip-адреса и также в двух правых байтах. Запишем эти вычисления в условие if all ()

```
from ipaddress import *
k = 0
#переберем все значения a
for a in range(256):
    # через f-строку подставлять в третий байт адреса значения a
    net = ip_network(f'207.0.{a}.167/255.255.255.192', 0)
    #проверим заданное условие, используя двоичную запись адреса
    if all(f'{ip:b}'[:16].count('0') > f'{ip:b}'[16:].count('0') for ip in net):
        k += 1
print(k)
```

Результат работы программы:

37

Ответ : 37

Telegram: @fast_ege

DZ_132_11

Задача № 11(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_456241208?t=0h23m0s

Решение

```
from ipaddress import *
net = ip_network('123.222.111.192/255.255.255.248')
k = 0
#получим двоичную запись для каждого значения ip-адреса сети
for ip in net:
    b = f'{ip:b}'
#проверим заданное условие для двоичной записи ip-адреса, срез от 24 цифры
    if b[24:].count('1')%3!=0:
        k += 1
print(k)
```

Результат работы программы:

5

Ответ: 5

Telegram: @fast_ege

DZ_132_12

Задача № 12(18138)

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 172.16.168.0 и маской сети 255.255.255.X.

Определите значение X, если известно, что в этой сети ровно 35 адресов, в которых количество нулей кратно 7. В ответе запишите число в 10 системе счисления.

Ссылка на видео-разбор с таймингом: https://vk.com/video-205546952_456241208?t=0h24m50s

Решение

В этой задаче требуется перебрать значения маски. Будем использовать перебор в виде количества единиц, от 24 цифры маски до 32

```
from ipaddress import *
# перебор маски
for mask in range(24,33):
    net = ip_network(f'172.16.168.0/{mask}')
    k = 0
#получим двоичную запись для каждого значения ip-адреса сети
    for ip in net:
        b = f'{ip:b}'
#проверим заданное условие
        if b.count('0')%7==0:
            k += 1
    if k==35:
#выведем на печать маску удовлетворяющую условиям
        print(net.netmask)
```

Результат работы программы:

255.255.255.128

Ответ: 128

Telegram: @fast__ege