

**Задание №13**  
**Решение задач из ЕГЭ**  
**на Питоне**

**Автор Стрельникова Л.В.**



**Решение задач тип 1 и 3**  
**Восстановление IP и URL адреса**

На сервере test.edu находится файл demo.net, доступ к которому осуществляется по протоколу http. Фрагменты адреса данного файла закодированы буквами А, Б ... Ж (см. таблицу).

Запишите последовательность этих букв, которая кодирует адрес указанного файла в Интернете.

**http:// test.edu / demo.net**

**ДВАЕГБЖ**

А	test
Б	demo
В	://
Г	/
Д	http
Е	.edu
Ж	.net

## Тип 13 № 7258(1 балл)

Идентификатор некоторого ресурса сети Интернет имеет следующий вид: <http://www.ftp.ru/index.html>

Какая часть этого идентификатора указывает на протокол, используемый для передачи ресурса? Выпишите нужную часть.

### Решение.

Протокол — часть адреса, стоящая до двоеточия.

1. **www** — world wide web.
2. **ftp** — протокол передачи информации.
3. **http** — протокол поиска информации в сети.
4. **https** — почтовый протокол

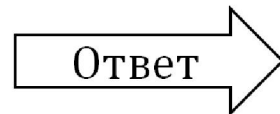
# Задача №1

1) Сотруднику фирмы продиктовали по телефону IP-адрес компьютера. Сотрудник записал этот адрес, но не поставил разделительные точки: **2153256182**  
Восстановите IP-адрес.

## Примечание

Нужно разделить на 4 группы чисел, каждое из которых от **0** до **255**.

**215.32.56.182**



## Задача №2 ( из ЕГЭ №1)

Петя записал IP-адрес на листке бумаги и положил его в карман куртки. Петина мама случайно постирала куртку вместе с запиской. После стирки Петя обнаружил в кармане обрывки с фрагментами IP-адреса. Эти фрагменты обозначены буквами **А, Б, В, Г**. Восстановите IP-адрес. В ответе укажите последовательность букв, обозначающих фрагменты, в порядке, соответствующем IP-адресу.

.64	3.13	3.133	20
<b>А</b>	<b>Б</b>	<b>В</b>	<b>Г</b>

**Ответ: 203.133.133.64 – ГБВА**

Ответ

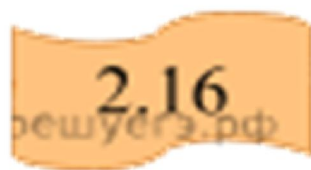
Числа между  
точками должны  
быть < 255!

## Задача №3 ( из ЕГЭ №3)

На месте преступления были обнаружены четыре обрывка бумаги. Следствие установило, что на них записаны фрагменты одного IP-адреса. Криминалисты обозначили эти фрагменты буквами А, Б, В и Г. Восстановите IP-адрес. В ответе укажите последовательность букв, обозначающих фрагменты, в порядке, соответствующем IP-адресу.



А



Б



В



Г

**Ответ: 162.168.32.64 – ВБГА**

## Решение задач тип 2

### Подсчет количества адресов в сети

Адрес сети:  $N = 2^n$

Адрес устройств:  $N = 2^n - 2$

Автор Стрельникова Л.В.

Для узла с IP – адресом 111.81.200.27 адрес сети равен 111.81.192.0.  
Какое наименьшее количество адресов может быть в этой сети.

\*\*\*\*\*.\*\*\*\*\*.\*\*\*\*\*.\*\*\*\*\* - mask  
01101111.01010001.11001000.00011011 - ip  
01101111.01010001.11000000.00000000 - net

11111111.11111111.11\*\*0000.00000000 - mask  
01101111.01010001.11001000.00011011 - ip  
01101111.01010001.11000000.00000000 - net

11111111.11111111.1111 0000.00000000 - mask  
01101111.01010001.1100 1000.00011011 - ip  
01101111.01010001.1100 0000.00000000 - net

$$N = 2^{12} = 4096$$

Адрес сети  
 $N = 2^n$   
Адрес устройств  
 $N = 2^n - 2$

## Задача №2 тип 4

В терминологии сетей TCP/IP маской сети называется двоичное число, определяющее, какая часть IP-адреса узла сети относится к адресу сети, а какая — к адресу самого узла в этой сети. При этом в маске сначала (в старших разрядах) стоят единицы, а затем с некоторого места — нули. Обычно маска записывается по тем же правилам, что и IP-адрес, — в виде четырёх байтов, 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?

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

**11111111.11111111.11111110.00000000**

$$N = 2^n - 2$$

$$N = 2^9 - 2 = 510$$

1. Количество адресов -  $2^n$
2. Количество устройств -  $2^n - 2$

## Решение задача №2 тип 4

1. Количество адресов -  $2^n$
2. Количество устройств -  $2^n - 2$

```
from ipaddress import *  
net = ip_network('0.0.0.0/255.255.254.0')  
print(net.num_addresses-2)
```

**N = 510**

## Задача №3 тип 4

В терминологии сетей TCP/IP маской сети называется двоичное число, определяющее, какая часть IP-адреса узла сети относится к адресу сети, а какая — к адресу самого узла в этой сети. При этом в маске сначала (в старших разрядах) стоят единицы, а затем с некоторого места — нули. Обычно маска записывается по тем же правилам, что и IP-адрес, — в виде четырёх байтов, 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.

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

108.133.75.64/26 64

108.133.75.64/27 32



64

## Задача №3 тип 4

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.

**11111111.11111111.11111111.11000000 - mask**

**10011100.10000101.01001011.01011011 - ip**

**10011100.10000101.01001011.01000000 - net**

$$N = 2^6 = 64$$

## Задача №4 тип 2

№ 6843. В терминологии сетей TCP/IP маской сети называется двоичное число, определяющее, какая часть IP-адреса узла сети относится к адресу сети, а какая — к адресу самого узла в этой сети. При этом в маске сначала (в старших разрядах) стоят единицы, а затем с некоторого места — нули. Обычно маска записывается по тем же правилам, что и IP-адрес, — в виде четырёх байтов, причём каждый байт записывается в виде десятичного числа. Адрес сети получается в результате применения поразрядной конъюнкции к заданному IP-адресу узла и маске.

**Сеть задана IP адресом 211.48.136.64 и маской сети: 255.255.255.224. Сколько в этой сети IP адресов, которые в двоичной записи IP адреса оканчиваются двумя единицами? В ответе укажите только число.**

```
from ipaddress import *

k = 0
net = ip_network('211.48.136.64/255.255.255.224')
for ip in net:
    b = f'{ip:b}'
    if b[-2]+b[-1] == '11':
        k += 1
print(k)
```

## Решение задачи №4

```
# Тип4. Ищем число IP адресов
from ipaddress import *
k=0
net = ip_network('210.48.136.64/255.255.255.224',0)
for ip in net:
    b = f'{ip:b}'
    if b[-2] + b[-1] == '11':
        k += 1
print(k)
```

## Задание 13 ФИПИ

## Задача №5 тип 2

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 сумма единиц в двоичной записи IP-адреса чётна? В ответе укажите только число.

# Тип4. Ищем число IP адресов

```
from ipaddress import *
```

```
k=0
```

```
net = ip_network('192.168.32.160/255.255.255.240')
```

```
for ip in net:
```

```
    b = f'{ip:b}'
```

```
    if b.count("1") % 2 == 0: # Считаем сколько адресов подходят под условие
```

```
        k += 1
```

```
print(k)
```

**Ответ: 8**

## Задача №6 тип 2

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 **122.159.136.144** и маской сети **255.255.255.248**.

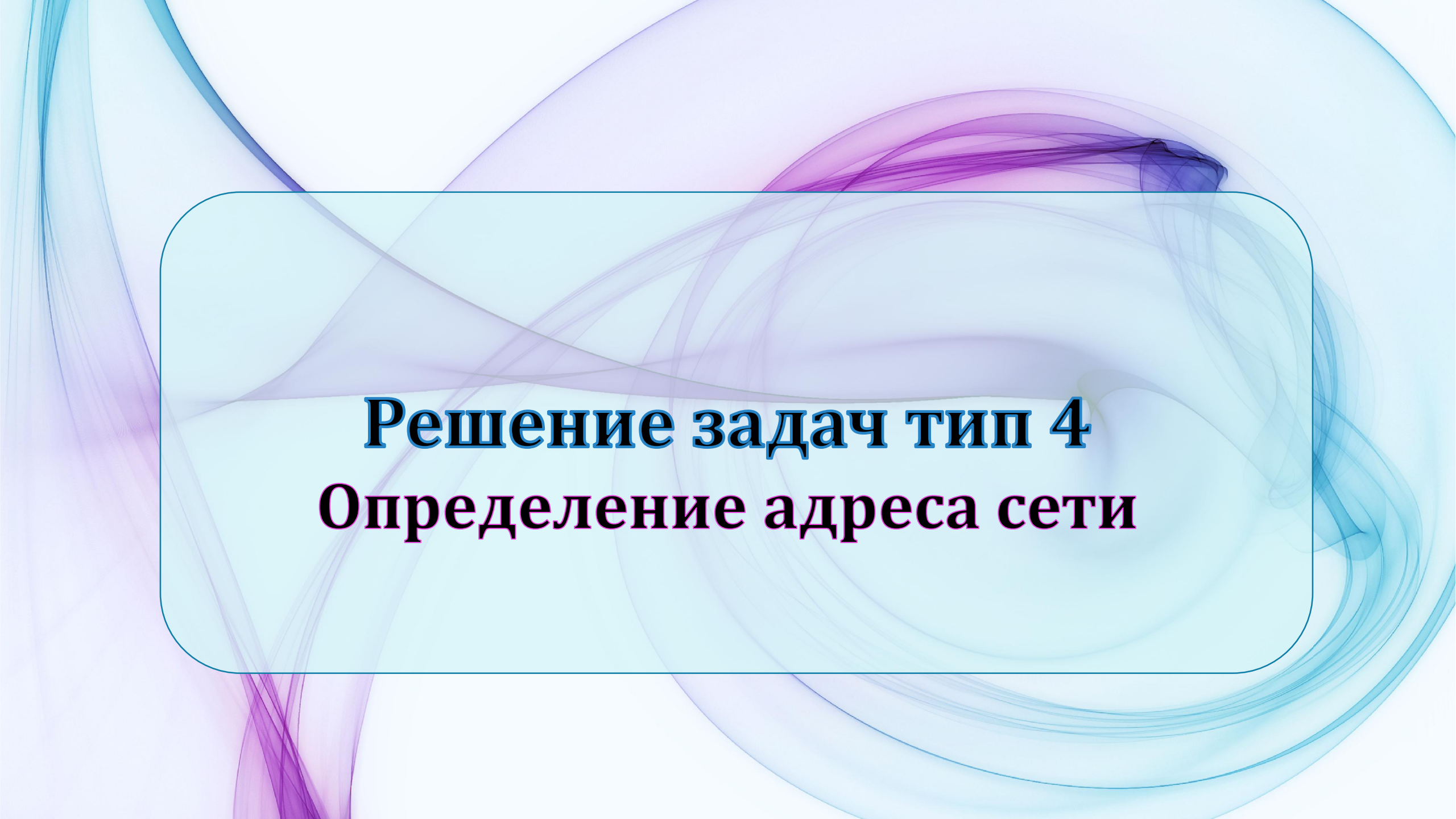
**Сколько в этой сети IP-адресов, для которых количество единиц в двоичной записи IP-адреса не кратно 4? В ответе укажите только число.**

## Решение задачи №6 тип 2

```
from ipaddress import*
# Заданный IP и маска сети
net = ip_network('122.159.136.144/255.255.255.248')
n = 0
for ip in net:
    x = bin(int(ip)).count('1')
    if x % 4 != 0:
        n += 1
print(n)
```

## Шаблон задач типа №2

```
# Тип4. Ищем число IP адресов
from ipaddress import *
k=0
net = ip_network('210.48.136.64/255.255.255.224',0)
for ip in net:
    b = f'{ip:b}'
    if условие:
        k += 1
print(k)
```



**Решение задач тип 4**  
**Определение адреса сети**

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

**IP-адрес: 145.92.137.88 Маска: 255.255.240.0**

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	145	255	137	128	240	88	92

Пример. Пусть искомый адрес сети 192.168.128.0 и дана таблица  
В этом случае правильный ответ будет HBAF.

**ВНЕА**

A	B	C	D	E	F	G	H
128	168	255	8	127	0	17	192

IP-адрес: 145.92.137.88    Маска: 255.255.240.0

**11111111.11111111.11110000.00000000    mask**

**10010001.01011100.10001001.01011000    IP**

---

**10010001.01011100.10000000.00000000    net**

**145.92.128.0    net**

<b>A</b>	<b>B</b>	<b>C</b>	<b>D</b>	<b>E</b>	<b>F</b>	<b>G</b>	<b>H</b>
<b>0</b>	<b>145</b>	<b>255</b>	<b>137</b>	<b>128</b>	<b>240</b>	<b>88</b>	<b>92</b>

**Ответ: BHEA**

# Определение адреса сети на Питоне

```
#Тип1. Ищем адрес сети  
from ipaddress import *  
net=ip_network(f'145.92.137.88/255.255.240.0',0)  
print(net)
```

145.92.128.0/20

Решение задач тип 5  
Определение третьего байта  
маски

# Задача №1

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.

Например, если IP-адрес узла равен 231.32.255.131, а маска равна 255.255.240.0, то адрес сети равен 231.32. 240.0.

Для узла с IP-адресом 224.128.112.142 адрес сети равен 224.128.64.0. Чему равен третий слева байт маски? Ответ запишите в виде десятичного числа.

Для узла с IP-адресом 224.128.112.142 адрес сети равен 224.128.64.0.  
Чему равен третий слева байт маски? Ответ запишите в виде десятичного числа.

IP 224.128.112.142

Net 224.128.64.0.

11111111.11111111.11000000.00000000 mask

11100000.10000000.01110000.10001110 IP

---

11100000.10000000.01000000.00000000 net

$$11000000 = 2^7 + 2^6 = 128 + 64 = 192$$

# Определение маски сети на Питоне

```
# Тип2. Ищем 3 байт маски
from ipaddress import *
for mask in range (33):
    net = ip_network(f'224.128.112.142/{mask}', 0)
    print(net, net.netmask)
```

Net 224.128.64.0.

224.128.64.0/18 255.255.192.0

## Задача №2

240

Для узла с IP-адресом 145.192.94.230 адрес сети равен 145.192.80.0. Чему равен третий слева байт маски? Ответ запишите в виде десятичного числа.

```
*****.*****.*****.***** - mask  
10010001.11000000.01011110.11100110 - ip  
10010001.11000000.01010000.00000000 - net
```

```
11111111.11111111.11110000.00000000 - mask  
10010001.11000000.01011110.11100110 - ip  
10010001.11000000.01010000.00000000 - net
```

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.

Одинаковые числа – в одной сети эти 2 ip-адреса

Одна маска для одной сети

```
01110000.01110101.01101011.01000110 - ip1
01110000.01110101.01111001.01010000 - ip2
```

### Задача №3

```
11111111.11111111.11100000.00000000 - mask
01110000.01110101.01101011.01000110 - ip1
01110000.01110101.01111001.01010000 - ip2
```

*Note: A bracket above the third octet of the mask (11100000) is labeled with the number 224.*

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, находящиеся в одной сети, имеют IP-адреса 112.117.107.70 и 112.117.121.80. Чему равно наибольшее значение третьего слева байт маски сети? Ответ запишите в виде десятичного числа.

```
from ipaddress import *
```

```
for mask in range(33):
```

```
    net1 = ip_network(f'112.117.107.70/{mask}', 0)
```

```
    net2 = ip_network(f'112.117.121.80/{mask}', 0)
```

```
    if net1 == net2:
```

```
        print(net)
```

```
112.117.96.0/19 255.255.224.0
```

## Задача №4

## Решение задачи №4

```
from ipaddress import *
for mask in range (33):
    net1 = ip_network(f'112.117.107.70/{mask}', 0)
    net2 = ip_network(f'112.117.121.80/{mask}', 0)
    if net1==net2:
        print(net1, net1.netmask)
```

```
0.0.0.0/0 0.0.0.0
0.0.0.0/1 128.0.0.0
64.0.0.0/2 192.0.0.0
96.0.0.0/3 224.0.0.0
112.0.0.0/4 240.0.0.0
112.0.0.0/5 248.0.0.0
112.0.0.0/6 252.0.0.0
112.0.0.0/7 254.0.0.0
112.0.0.0/8 255.0.0.0
112.0.0.0/9 255.128.0.0
112.64.0.0/10 255.192.0.0
112.96.0.0/11 255.224.0.0
112.112.0.0/12 255.240.0.0
112.112.0.0/13 255.248.0.0
112.116.0.0/14 255.252.0.0
112.116.0.0/15 255.254.0.0
112.117.0.0/16 255.255.0.0
112.117.0.0/17 255.255.128.0
112.117.64.0/18 255.255.192.0
112.117.96.0/19 255.255.224.0
```

112.117.96.0/19 255.255.224.0

## Задача №5

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.

## Решение задачи №5

```
from ipaddress import *
for mask in range (33):
    net=ip_network(f'175.122.80.13/{mask}',0)
    if net.num_addresses>=60:
        print(net, net.netmask)
```

**175.122.80.0/26**

175.122.80.0/20  
175.122.80.0/21  
175.122.80.0/22  
175.122.80.0/23  
175.122.80.0/24  
175.122.80.0/25  
175.122.80.0/26

Решение задач тип 6  
Определение числа 0 и 1 в  
маске сети

## Задача №1 тип 6

Для узла с IP-адресом 220.127.169.27 адрес сети равен 220.127.160.0. найдите наименьшее возможное количество единиц в двоичной записи маски подсети.

**Ответ 19**

```
***** . ***** . ***** . ***** - mask
```

```
11011100.01111111.10101001.00011001 - ip
```

```
11011100.01111111.10100000.00000000 - net
```

```
11111111.11111111. [REDACTED].00000000 - mask
```

```
11011100.01111111.10101001.00011001 - ip
```

```
11011100.01111111.10100000.00000000 - net
```

# Решение задачи №1 тип 6 на Питоне

Для узла с IP-адресом 220.127.169.27 адрес сети равен 220.127.160.0. найдите наименьшее возможное количество единиц в двоичной записи маски подсети.

```
from ipaddress import *
for i in range(32):
    net=ip_network("220.127.169.27/"+ str(i),0)
    sub = str(net).split("/")
    if sub[0] == "220.127.160.0":# нужная сеть
        print(net.prefixlen)
        break
```

Ответ 19

## Задача №2 тип 6

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.

```
# Тип 6. Ищем число нулей в маске подсети
from ipaddress import *
for mask in range (33):
    net = ip_network(f'241.185.253.57/{mask}', 0)
    print(net, 32-mask)
```

241.185.252.0/22 10

241.185.252.0/23 9

241.185.252.0/23 9

Ответ: 9

**P-13.** Для узла с IP-адресом 15.51.208.15 адрес сети равен 15.51.192.0. Найдите наименьшее возможное количество единиц в двоичной записи маски подсети.

## Задача №3 тип 6

11111111	11111111	11000000	00000000	- mask
00001111	00110011	11010000	00001111	- ip
00001111	00110011	11000000	00000000	- net

```
# Тип5. Ищем число единиц в маске подсети
from ipaddress import *
for mask in range(33):
    net = ip_network(f'15.51.208.15/{mask}', 0)
    print(net, net.netmask)
```

```
15.51.0.0/16 255.255.0.0
15.51.128.0/17 255.255.128.0
15.51.192.0/18 255.255.192.0
15.51.192.0/19 255.255.224.0
15.51.208.0/20 255.255.240.0
```

Ответ: 18.



**P-13.** Для узла с IP-адресом 15.51.208.15 адрес сети равен 15.51.192.0. Найдите наименьшее возможное количество единиц в двоичной записи маски подсети.

```
from ipaddress import *
for i in range(32):
    net = ip_network("15.51.208.15/" + str(i), 0)
    sub = str(net).split("/")
    if sub[0] == "15.51.192.0": # нужная сеть
        print(net.prefixlen)
        break
```

Задача №3 тип 6

Ответ: 18.



Решение А. Носкин):

Задача №3 тип 6

```
from ipaddress import *
for i in range(32):
    net = ip_network("15.51.208.15/" + str(i), 0)
    sub = str(net).split("/")
    if sub[0] == "15.51.192.0": # нужная сеть
        x10 = int(net.netmask) # маска в 10СС
        x2 = bin(x10)[2:] # маска в 2СС
        print(x2.count('1'))
        break
```

Ответ: **18.**



## Задача №4 тип 6

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.

```
print( '.'.join(f'{x:>08b}' for x in [157, 127, 182, 76]))  
print( '.'.join(f'{x:>08b}' for x in [157, 127, 190, 80]))
```

10011101.01111111.10110110.01001100

10011101.01111111.10111110.01010000

**157.127.182.76/30 157.127.190.80/30**

## Решение задачи №4 тип 6

# **Тип2.** Число 1 в масках подсетей

```
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)
```

```
157.127.182.76/30 157.127.190.80/30
```

```
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
```

Для узла с IP-адресом 116.123.64.53 адрес сети равен 116.123.64.0. Для скольких различных значений маски это возможно?

Ответ - 9

```
* . * . ***** . ***** - mask
* . * . 01000000 . 00110101 - ip
* . * . 01000000 . 00000000 - net
```

```
* . * . 11***** . **000000 - mask
* . * . 01000000 . 00110101 - ip
* . * . 01000000 . 00000000 - net
```

## Задача №5 тип 6

```
***** . **
000000 . 00
100000 . 00
110000 . 00
111000 . 00
111100 . 00
111110 . 00
111111 . 00
111111 . 10
111111 . 11
```

## Задача №6 тип 6

Для узла с IP-адресом 76.155.48.2 адрес сети равен 76.155.48.0. Для скольких различных значений маски это возможно?

```
from ipaddress import *
for mask in range(33):
    net=ip_network(f'76.155.48.2/{mask}', 0)
    print(net)
```

```
76.155.48.0/20
76.155.48.0/21
76.155.48.0/22
76.155.48.0/23
76.155.48.0/24
76.155.48.0/25
76.155.48.0/26
76.155.48.0/27
76.155.48.0/28
76.155.48.0/29
76.155.48.0/30
```

**Ответ : 11**

# Самостоятельное решение задач из ЕГЭ

**1. Для узла с IP-адресом 224.128.114.142 адрес сети равен 224.128.96.0. Чему равен третий слева байт маски? Ответ запишите в виде десятичного числа.**

**2. Для узла с IP-адресом 227.138.127.144 адрес сети равен 227.138.64.0. Чему равен третий слева байт маски? Ответ запишите в виде десятичного числа.**

**3. Для узла с IP-адресом 119.167.50.77 адрес сети равен 119.167.48.0. Чему равно наименьшее возможное значение третьего слева байта маски? Ответ запишите в виде десятичного числа.**

**4. Для узла с IP-адресом 119.83.200.27 адрес сети равен 119.83.192.0. Каково наибольшее возможное количество единиц в разрядах маски?**

# Решение задач

```
from ipaddress import *  
for mask in range (33):  
    net = ip_network (f'224.128.114.142/{mask} ', 0)  
    print (net, net.netmask)
```

**224.128.96.0/19 255.255.224.0**

```
from ipaddress import *  
for mask in range (33):  
    net = ip_network (f'227.138.127.144/{mask} ', 0)  
    print (net, net.netmask)
```

**227.138.64.0/18 255.255.192.0**

## Решение задач

3. Для узла с IP-адресом 119.167.50.77 адрес сети равен 119.167.48.0. Чему равно наименьшее возможное значение третьего слева байта маски? Ответ запишите в виде десятичного числа.

```
# Тип2. Ищем 3 байт маски
from ipaddress import *
for mask in range (33):
    net = ip_network (f'119.167.50.77/{mask} ', 0)
    print (net, net.netmask)
```

```
119.167.48.0/20 255.255.240.0
119.167.48.0/21 255.255.248.0
119.167.48.0/22 255.255.252.0
```

## Решение задач

4. Для узла с IP-адресом 119.83.200.27 адрес сети равен 119.83.192.0. Каково наибольшее возможное количество единиц в разрядах маски?

```
# Тип5. Ищем число единиц в маске подсети
from ipaddress import *
for mask in range (33):
    net = ip_network(f'119.83.200.27/{mask}', 0)
    print(net)
```

```
119.83.192.0/18
119.83.192.0/19
119.83.192.0/20
```

# Литература

1. Сайт для 13 задания