

Конспект по заданию №13

КОМАНДЫ

найти маску:

```
ip_net.netmask
```

найти адрес сети:

```
ip_net.network_address
```

записать IP-адрес:

```
ip_add = ip_address("152.152.152.152")
```

записать адрес сети:

```
ipnet = ip_network("IPадрес/маска", 0)
```

количество адресов в сети:

```
ip_net.num_addresses
```

1) IP-адрес

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 180.23.32.0 и маской сети 255.255.240.0. Сколько в этой сети IP-адресов, для которых сумма единиц в двоичной записи IP-адреса чётна?

В ответе укажите только число.

```
from ipaddress import *
ip_net=ip_network("180.23.32.0/255.255.240.0",0)
k=0
for ip_add in ip_net:
    if bin(int(ip_add)).count("1")%2==0:
        k+=1
print(k)
```

в двоичной записи IP-адреса суммарное количество единиц в правых двух байтах больше суммарного количества единиц в левых двух байтах.

```
bin(int(ip_add))[-16:].count("1") > bin(int(ip_add))[:-16].count("1")
```

правые > левые

АДРЕС	кол-во "1" маски	маска
145.128.0.0/9	9	255.128.0.0
145.192.0.0/10	10	255.192.0.0
145.192.0.0/11	11	255.224.0.0
145.192.0.0/12	12	255.240.0.0

```
from ipaddress import *
ip_add=ip_address("145.192.94.230")
for mask in range(33):
    ip_net=ip_network(f"145.192.94.230/{mask}",0)
    if ip_add in ip_net:
        print(ip_net, ip_net.netmask)
```

3) Наибольший или наименьший IP-адрес

Широковещательным адресом называется специализированный адрес, в котором на месте нулей в маске стоят единицы.

Сеть задана IP-адресом одного из входящих в неё узлов 218.194.82.148 и сетевой маской 255.255.255.192.

Найдите наибольший IP-адрес в данной сети, который может быть назначен компьютеру.

В ответе укажите найденный IP-адрес без разделителей.

```
from ipaddress import *
ip_net=ip_network("218.194.82.148/255.255.255.192",0)
print(ip_net[-2])
```

`ip_net [-2]` элемент - наибольший

`ip_net [1]` элемент - наименьший