

1. Задача проверки валидности IP-адреса:

- Создайте класс "IP-адрес", который имеет переменные-члены для каждого октета IP-адреса.
- Реализуйте перегрузку оператора ввода (`>>`), чтобы пользователь мог вводить IP-адрес с клавиатуры.
- Напишите метод-член класса, который проверяет, является ли введенный IP-адрес валидным (соответствует формату xxx.xxx.xxx.xxx и значения каждого октета находятся в диапазоне от 0 до 255).

2. Задача проверки принадлежности IP-адреса к подсети:

- Создайте класс "Маска сети", который имеет переменные-члены для каждого октета маски.
- Реализуйте перегрузку оператора ввода (`>>`), чтобы пользователь мог вводить маску сети с клавиатуры.
- Добавьте метод-член класса "Маска сети", который проверяет, принадлежит ли заданный IP-адрес данной маске сети. Например, метод может проверять, совпадают ли первые N битов IP-адреса с первыми N битами маски сети.

3. Задача получения сетевого адреса:

- Добавьте метод-член класса "IP-адрес", который возвращает сетевой адрес, используя заданную маску сети. Например, если IP-адрес - 192.168.1.100, а маска сети - 255.255.255.0, сетевой адрес будет 192.168.1.0.

4. Задача сравнения IP-адресов:

- Реализуйте перегрузку оператора сравнения (`==`) для класса "IP-адрес", чтобы можно было проверять равенство двух IP-адресов.
- Реализуйте перегрузку оператора сравнения (`<`) для класса "IP-адрес", чтобы можно было сравнивать IP-адреса и определять, является ли один IP-адрес меньшим или большим по сравнению с другим.

5. Задача подсчета количества возможных узлов в сети:

- Добавьте метод-член класса "Маска сети", который вычисляет количество возможных узлов в сети, используя заданную маску. Например, для маски 255.255.255.0 количество возможных узлов будет 254.