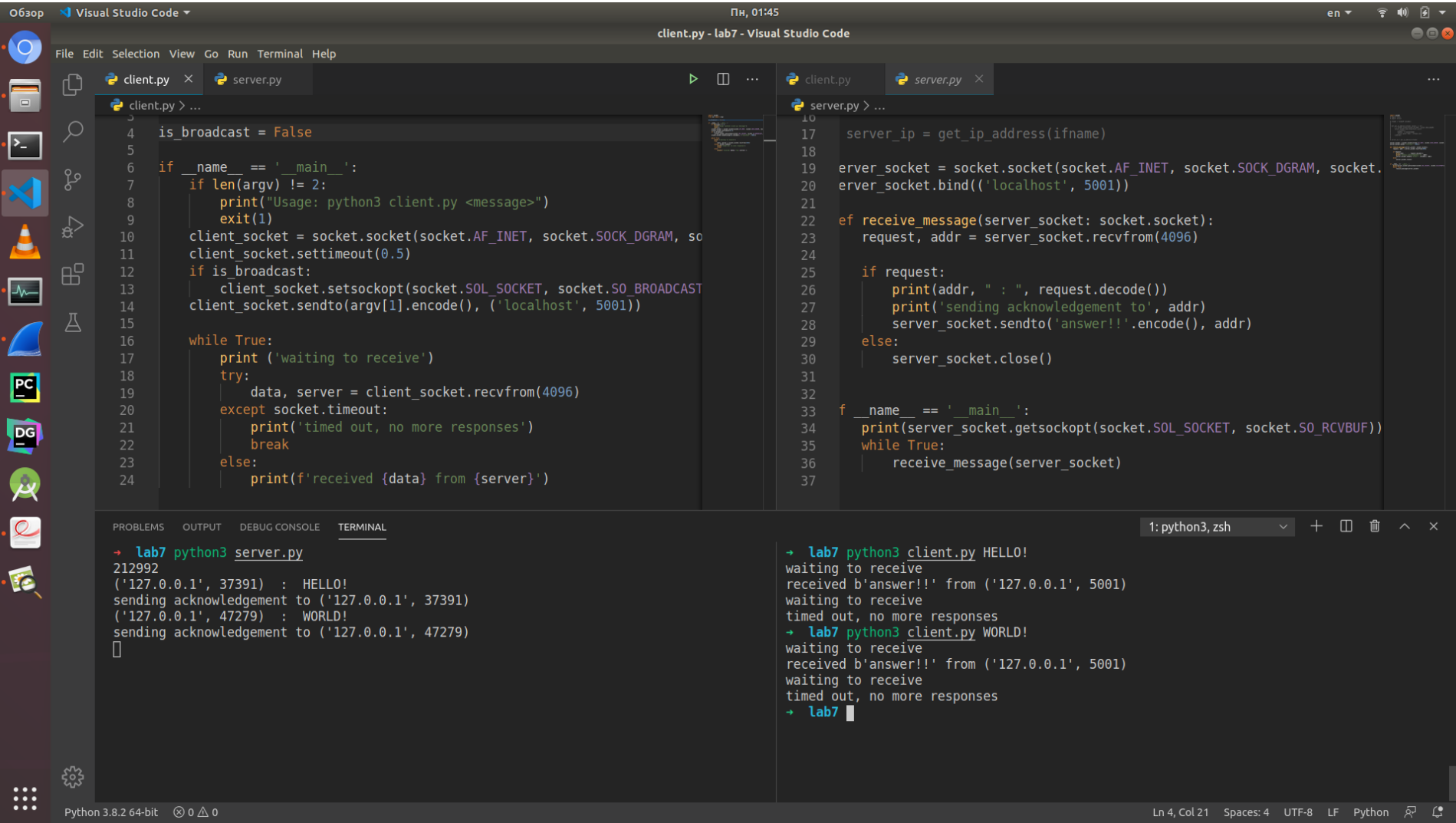


Отчет по лабораторной № 7

Задание 3б

Было написано небольшое консольное приложение на python. Клиент отправляет прослушивающему на 127.0.0.1:5001 серверу заданное сообщение и получает ответ.

Демонстрация работы в обычном режиме



212992 - размер максимальной дейтаграммы, которую можно отправить, в linux можно получить с помощью

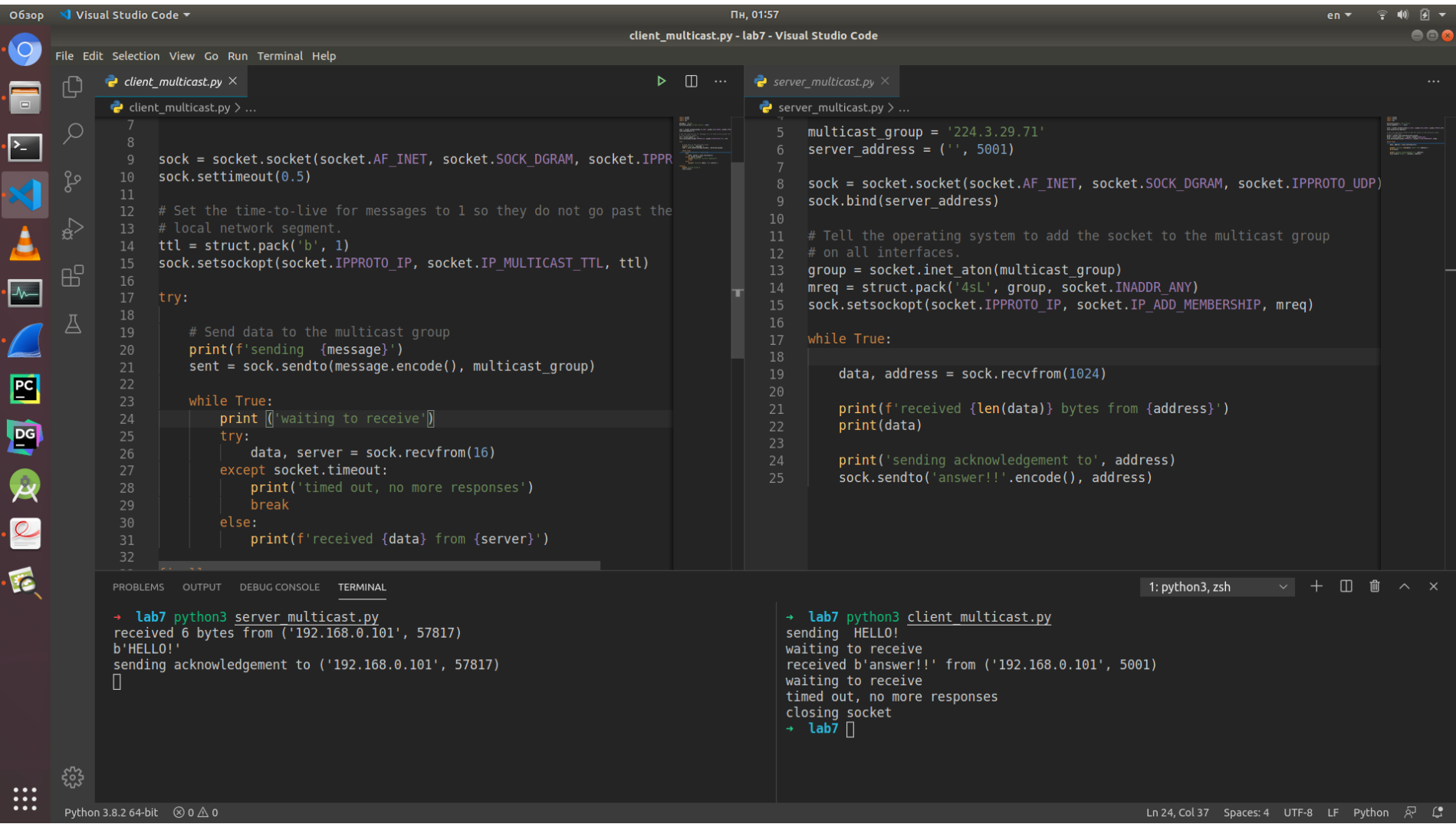
```
socket.getsockopt(socket.SOL_SOCKET, socket.SO_RCVBUF)
```

Отправленные пакеты можно посмотреть в дампе lab7.pcapng (5 и 6 пакеты - сообщение HELLO! и ответ на него, 13 и 14 пакеты - сообщение WORLD! и ответ на него)

Задание 4а

Аналогично и для задания 4а было написано консольное приложение для обмена MULTICAST сообщениями.

Демонстрация работы приложения



В файле lab7_multicast.pcapng можно увидеть как с помощью IGMP пакетов 7 и 9 сервер покидает группу 224.3.29.7 (тк сервер перезапускался) и затем вначале своей работы присоединяется к этой же группе для всех интерфейсов - пакеты 16 и 19. Это действие выполняется в следующем фрагменте кода:

```
group = socket.inet_aton('224.3.29.71')
mreq = struct.pack('4sL', group, socket.INADDR_ANY)
sock.setsockopt(socket.IPPROTO_IP, socket.IP_ADD_MEMBERSHIP, mreq)
```

У клиента в заголовке IP пакета выставляется TTL=1, чтобы гарантировать, что пакет гарантированно не выйдет во внешнюю сеть. Это выполняется в:

```
ttl = struct.pack('b', 1)
sock.setsockopt(socket.IPPROTO_IP, socket.IP_MULTICAST_TTL, ttl)
```

Исходный код на [гитхабе](#)