

Курбатов Ярослав, РПО МКН

Задание 1

Ruby MRI 3.0.0

Запуск: `ruby task-1/rip/main.rb`

Каждый хост работает (принимает и отправляет пакеты) в отдельном потоке (точнее, даже в двух). В stdout выводится информация о состоянии таблицы на каждой итерации + финальная таблица для каждого хоста.

Конфигурация задана в `task-1/rip/configuraiton.yml`

Пример (для конфига, который описан в `configuration.yml`):

```
ctrl@prasilip@ruary:~/ipbo/2023-spring/networks/ipbo-masters-comput/homeworks/11$ ruby task-1/rip/main.rb
host: 127.0.0.2, iteration: 1
dest: 127.0.0.2, next_hop: 127.0.0.2, metric: 0
dest: 127.0.0.1, next_hop: 127.0.0.1, metric: 1

host: 127.0.0.3, iteration: 1
dest: 127.0.0.3, next_hop: 127.0.0.3, metric: 0
dest: 127.0.0.6, next_hop: 127.0.0.6, metric: 1

host: 127.0.0.2, iteration: 2
dest: 127.0.0.2, next_hop: 127.0.0.2, metric: 0
dest: 127.0.0.1, next_hop: 127.0.0.1, metric: 1
dest: 127.0.0.3, next_hop: 127.0.0.3, metric: 1
dest: 127.0.0.6, next_hop: 127.0.0.3, metric: 2

host: 127.0.0.4, iteration: 1
dest: 127.0.0.4, next_hop: 127.0.0.4, metric: 0
dest: 127.0.0.3, next_hop: 127.0.0.3, metric: 1
dest: 127.0.0.6, next_hop: 127.0.0.3, metric: 2

host: 127.0.0.6, iteration: 1
dest: 127.0.0.6, next_hop: 127.0.0.6, metric: 0
dest: 127.0.0.3, next_hop: 127.0.0.3, metric: 1

host: 127.0.0.3, iteration: 2
dest: 127.0.0.3, next_hop: 127.0.0.3, metric: 0
dest: 127.0.0.6, next_hop: 127.0.0.6, metric: 1
host: 127.0.0.1, iteration: 1
dest: 127.0.0.1, next_hop: 127.0.0.1, metric: 0
dest: 127.0.0.2, next_hop: 127.0.0.2, metric: 1
dest: 127.0.0.3, next_hop: 127.0.0.2, metric: 2
dest: 127.0.0.6, next_hop: 127.0.0.2, metric: 3

dest: 127.0.0.2, next_hop: 127.0.0.2, metric: 1
dest: 127.0.0.1, next_hop: 127.0.0.2, metric: 2

host: 127.0.0.5, iteration: 1
dest: 127.0.0.5, next_hop: 127.0.0.5, metric: 0
dest: 127.0.0.4, next_hop: 127.0.0.4, metric: 1
dest: 127.0.0.3, next_hop: 127.0.0.4, metric: 2
dest: 127.0.0.6, next_hop: 127.0.0.4, metric: 3

host: 127.0.0.3, iteration: 3
dest: 127.0.0.3, next_hop: 127.0.0.3, metric: 0
dest: 127.0.0.6, next_hop: 127.0.0.6, metric: 1
dest: 127.0.0.2, next_hop: 127.0.0.2, metric: 1
dest: 127.0.0.1, next_hop: 127.0.0.2, metric: 2
dest: 127.0.0.4, next_hop: 127.0.0.4, metric: 1

host: 127.0.0.4, iteration: 2
dest: 127.0.0.4, next_hop: 127.0.0.4, metric: 0
dest: 127.0.0.3, next_hop: 127.0.0.3, metric: 1
dest: 127.0.0.6, next_hop: 127.0.0.3, metric: 2
dest: 127.0.0.5, next_hop: 127.0.0.5, metric: 1

host: 127.0.0.2, iteration: 3
dest: 127.0.0.2, next_hop: 127.0.0.2, metric: 0
dest: 127.0.0.1, next_hop: 127.0.0.1, metric: 1
dest: 127.0.0.3, next_hop: 127.0.0.3, metric: 1
dest: 127.0.0.6, next_hop: 127.0.0.3, metric: 2

host: 127.0.0.3, iteration: 4
host: 127.0.0.5, iteration: 2
dest: 127.0.0.3, next_hop: 127.0.0.3, metric: 0
dest: 127.0.0.4, next_hop: 127.0.0.4, metric: 1
dest: 127.0.0.6, next_hop: 127.0.0.3, metric: 0
```

...

```

dest: 127.0.0.4, next_hop: 127.0.0.4, metric: 0
dest: 127.0.0.3, next_hop: 127.0.0.3, metric: 1
dest: 127.0.0.6, next_hop: 127.0.0.3, metric: 2
dest: 127.0.0.5, next_hop: 127.0.0.5, metric: 1
dest: 127.0.0.2, next_hop: 127.0.0.3, metric: 2
dest: 127.0.0.1, next_hop: 127.0.0.1, metric: 3

host: 127.0.0.6, iterations: 4
dest: 127.0.0.6, next_hop: 127.0.0.6, metric: 0
dest: 127.0.0.3, next_hop: 127.0.0.3, metric: 1
dest: 127.0.0.2, next_hop: 127.0.0.3, metric: 2
dest: 127.0.0.1, next_hop: 127.0.0.3, metric: 3
dest: 127.0.0.4, next_hop: 127.0.0.3, metric: 2
dest: 127.0.0.5, next_hop: 127.0.0.3, metric: 3

host: 127.0.0.5, iterations: 4
dest: 127.0.0.5, next_hop: 127.0.0.5, metric: 0
dest: 127.0.0.4, next_hop: 127.0.0.4, metric: 1
dest: 127.0.0.3, next_hop: 127.0.0.4, metric: 2
dest: 127.0.0.6, next_hop: 127.0.0.4, metric: 3
dest: 127.0.0.2, next_hop: 127.0.0.4, metric: 3
dest: 127.0.0.1, next_hop: 127.0.0.4, metric: 4

... FINAL STATE
host 127.0.0.1
dest: 127.0.0.1, next_hop: 127.0.0.1, metric: 0
dest: 127.0.0.2, next_hop: 127.0.0.2, metric: 1
dest: 127.0.0.3, next_hop: 127.0.0.2, metric: 2
dest: 127.0.0.6, next_hop: 127.0.0.2, metric: 3
dest: 127.0.0.4, next_hop: 127.0.0.2, metric: 3
dest: 127.0.0.5, next_hop: 127.0.0.2, metric: 4

host 127.0.0.2
dest: 127.0.0.2, next_hop: 127.0.0.2, metric: 0
dest: 127.0.0.1, next_hop: 127.0.0.1, metric: 1
dest: 127.0.0.3, next_hop: 127.0.0.3, metric: 1
dest: 127.0.0.6, next_hop: 127.0.0.3, metric: 2
dest: 127.0.0.4, next_hop: 127.0.0.3, metric: 2
dest: 127.0.0.5, next_hop: 127.0.0.3, metric: 3

host 127.0.0.3
dest: 127.0.0.3, next_hop: 127.0.0.3, metric: 0
dest: 127.0.0.6, next_hop: 127.0.0.4, metric: 1
dest: 127.0.0.2, next_hop: 127.0.0.2, metric: 1
dest: 127.0.0.1, next_hop: 127.0.0.2, metric: 2
dest: 127.0.0.4, next_hop: 127.0.0.4, metric: 1
dest: 127.0.0.5, next_hop: 127.0.0.4, metric: 2

host 127.0.0.4
dest: 127.0.0.4, next_hop: 127.0.0.4, metric: 0
dest: 127.0.0.3, next_hop: 127.0.0.3, metric: 1
dest: 127.0.0.6, next_hop: 127.0.0.5, metric: 2
dest: 127.0.0.5, next_hop: 127.0.0.5, metric: 1
dest: 127.0.0.2, next_hop: 127.0.0.3, metric: 2
dest: 127.0.0.1, next_hop: 127.0.0.3, metric: 3

host 127.0.0.5
dest: 127.0.0.5, next_hop: 127.0.0.5, metric: 0
dest: 127.0.0.4, next_hop: 127.0.0.4, metric: 1
dest: 127.0.0.3, next_hop: 127.0.0.4, metric: 2
dest: 127.0.0.6, next_hop: 127.0.0.4, metric: 3
dest: 127.0.0.2, next_hop: 127.0.0.4, metric: 3
dest: 127.0.0.1, next_hop: 127.0.0.4, metric: 4

host 127.0.0.6
dest: 127.0.0.6, next_hop: 127.0.0.6, metric: 0
dest: 127.0.0.3, next_hop: 127.0.0.3, metric: 1
dest: 127.0.0.2, next_hop: 127.0.0.3, metric: 2
dest: 127.0.0.1, next_hop: 127.0.0.3, metric: 3
dest: 127.0.0.4, next_hop: 127.0.0.3, metric: 2
dest: 127.0.0.5, next_hop: 127.0.0.3, metric: 3

OK

```

пример вывода см. В task-1/rip/log.txt

Задание 2

Язык: Python 3.10.0

Зависимости (для GUI):

kivy=2.1.0

kivymd=1.1.0

(можно установить через python “python -m pip install <lib-name-with-version>”

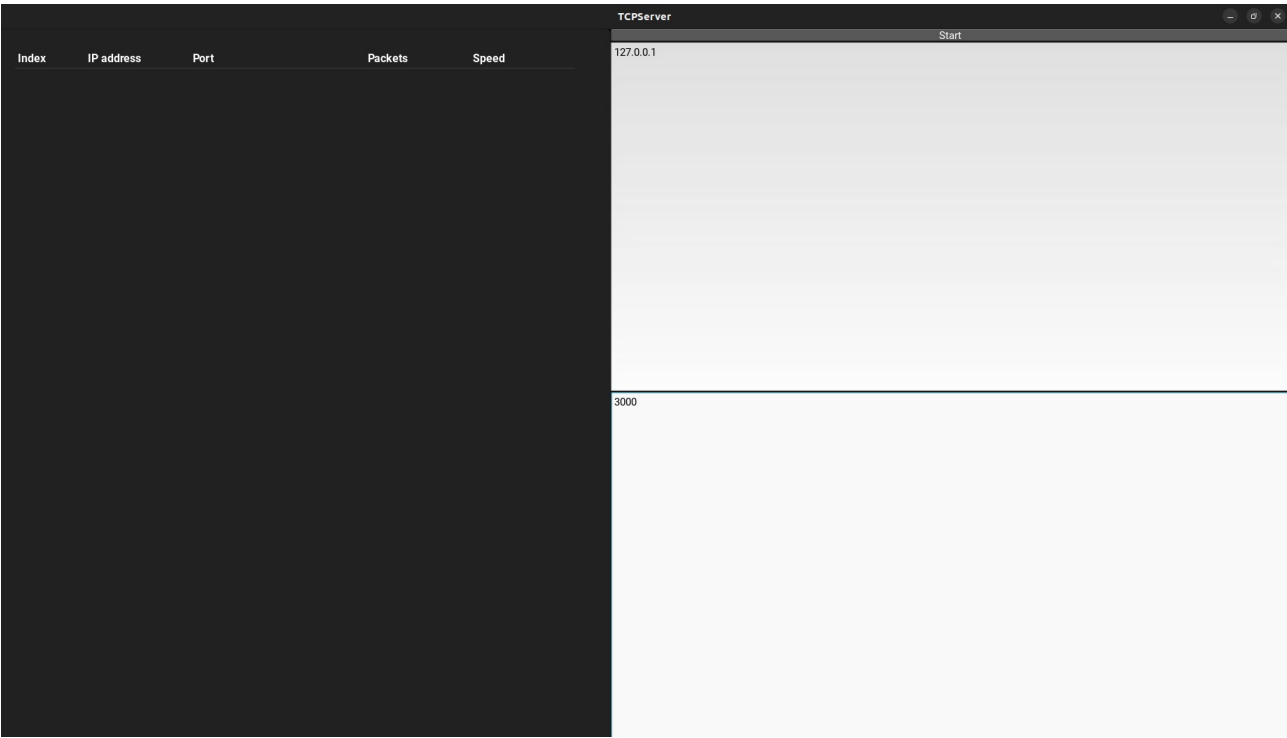
Запуск клиента (и для TCP, и для UDP): python task-2/client.py

Запуск TCP-сервера: python task-2/server/tcp.py

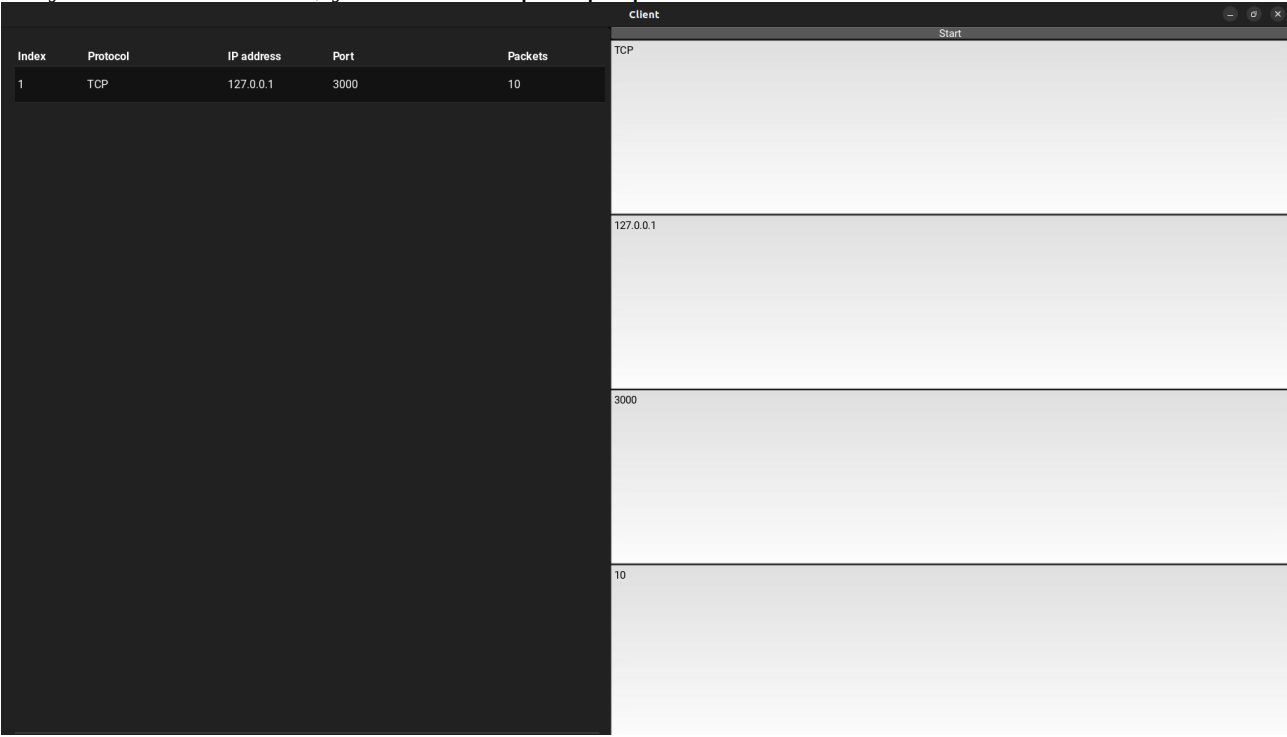
Запуск UDP-сервера: python task-2/server/udp.py

Для начала посмотри на работу TCP клиента/сервера.

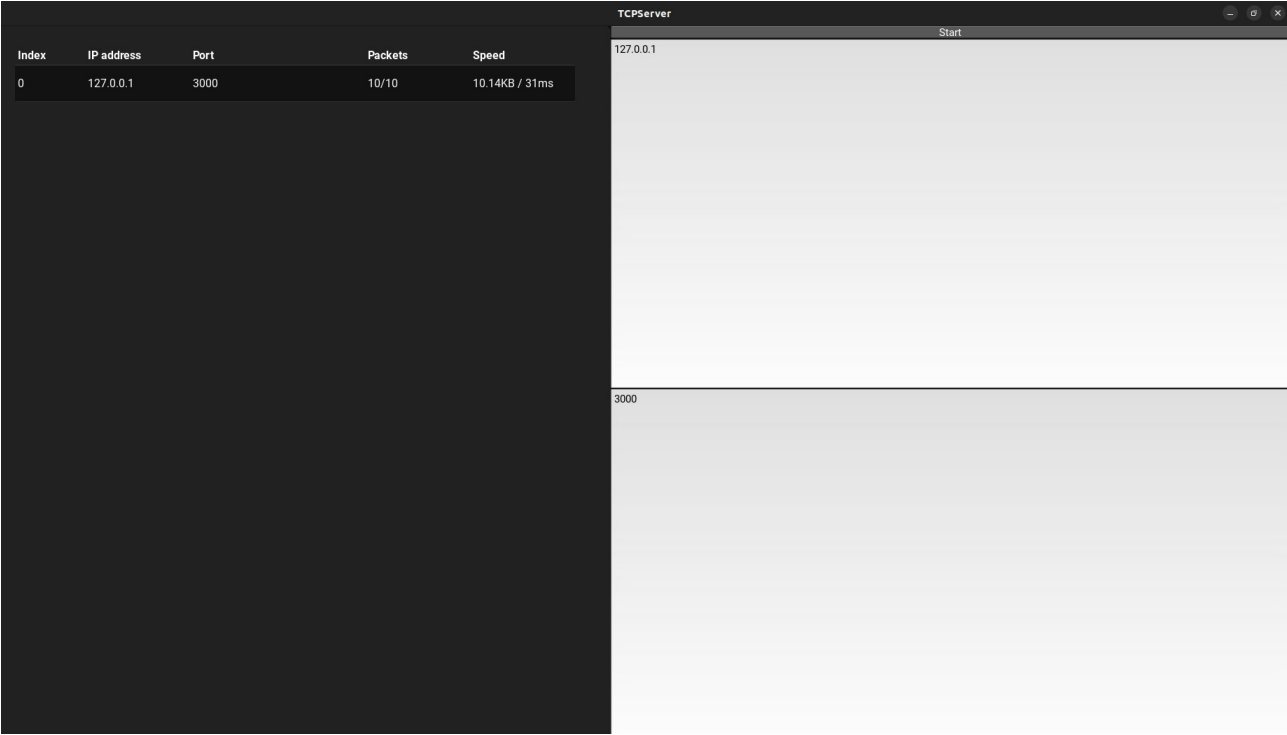
Запустим TCP сервер на порту 127.0.0.1:3000:



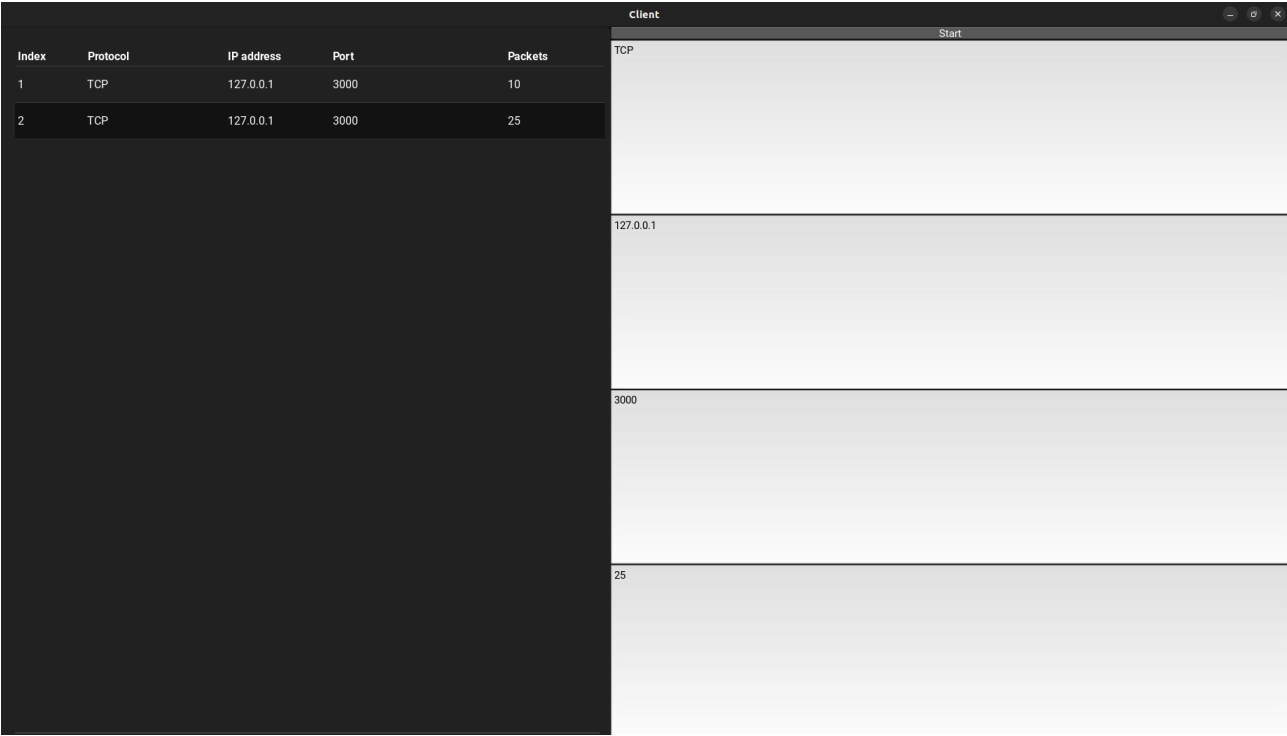
Запустим TCP-клиента, указав IP и порт сервера и кол-во пакетов 10:



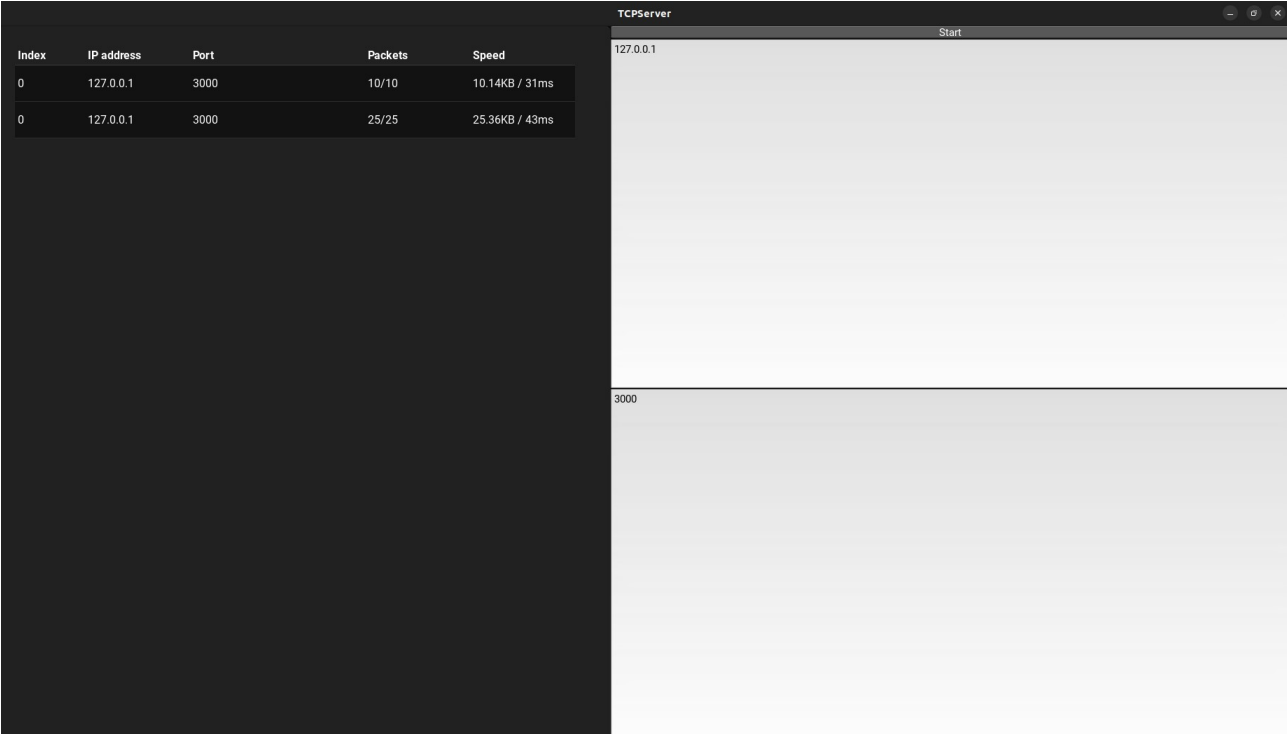
Результат отображается в GUI сервера:



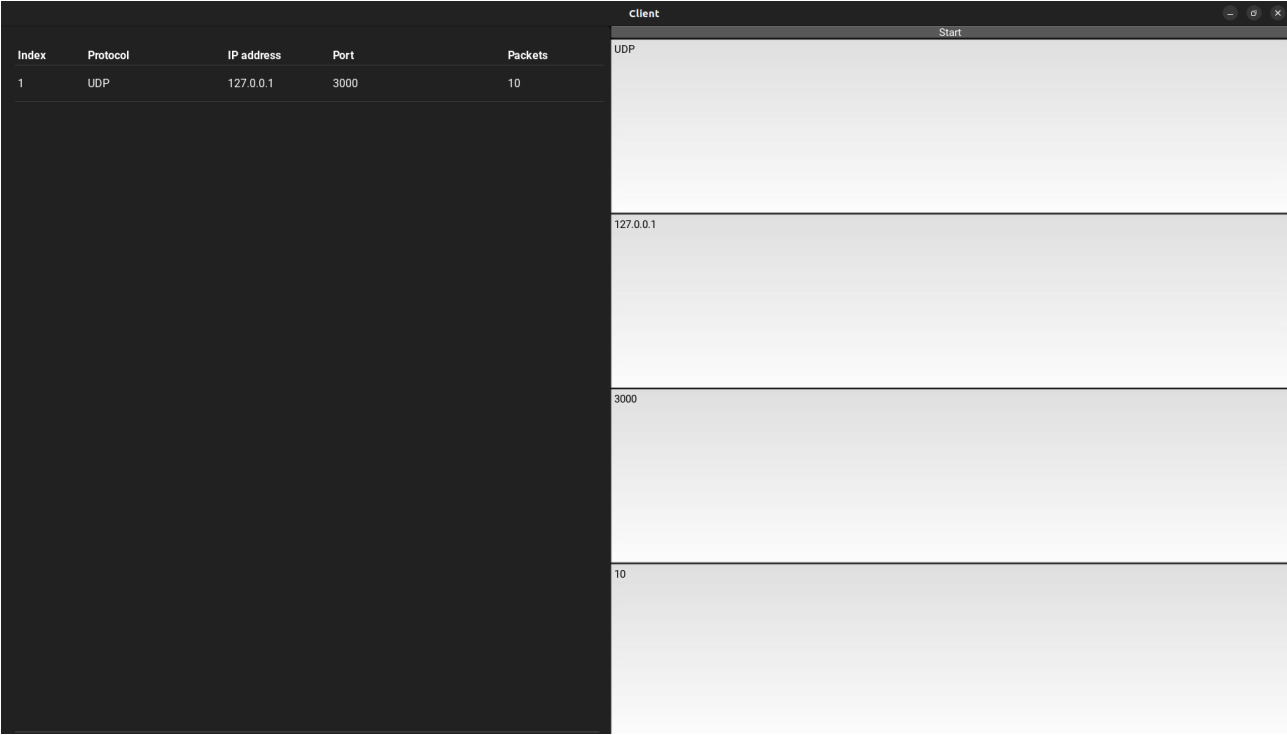
Еще раз запустим клиента, указав 25 пакетов:



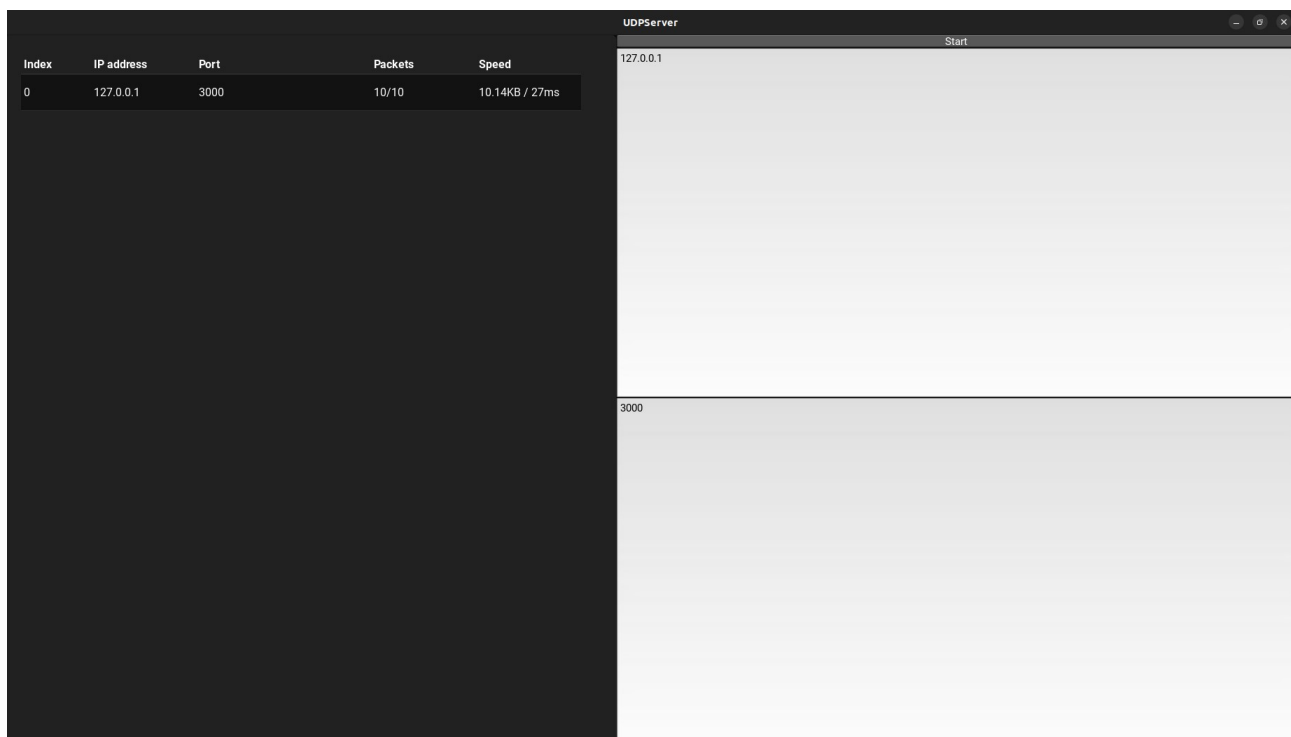
Отчет сервера:



Запустим UDP-сервер (task-2/server/udp.py), запустим клиента, указав в качестве протокола UDP и кол-во пакетов – 10:



проверим результат:



Задание 3

Язык: Python 3.10.0

Зависимости (для GUI):

kivy=2.1.0

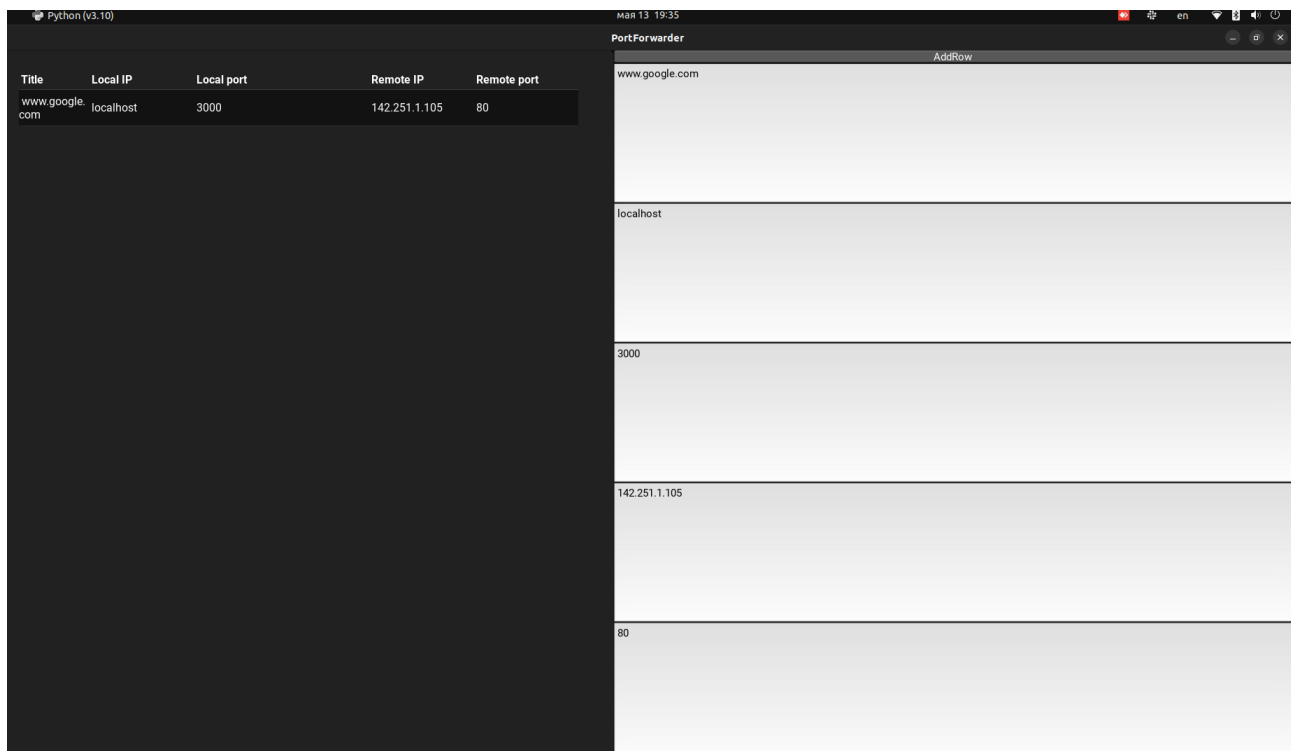
kivymd=1.1.0

(можно установить через python “python -m pip install <lib-name-with-version>”

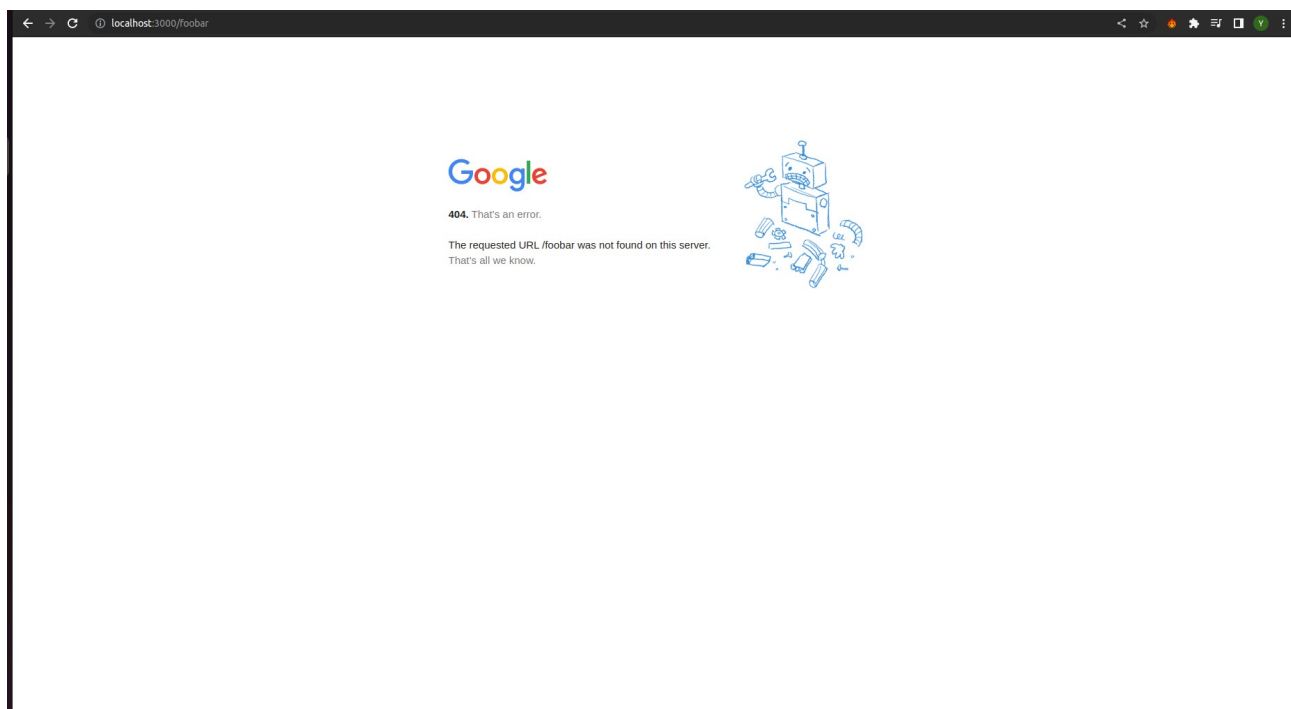
Запуск:

```
python task-3/ui.py
```

Добавим запись для port forwarding с (localhost, 3000) → (142.251.1.105):80 (IP адрес www.google.com)



Перейдем по localhost:3000 (гугл в любом случае ответит ошибкой, потому что в запросе на localhost:3000 передается заголовок Host: localhost, GWS такое не пропустит очевидно)



Попробуем добавить запись для проксирования (localhost, 3002) → (localhost, 4000)

Title	Local IP	Local port	Remote IP	Remote port
www.google.com	127.0.0.1	3000	64.233.165.105	80
www.yandex.ru	127.0.0.1	3001	77.88.55.60	80
localhost	127.0.0.1	3002	127.0.0.1	4000

AddRow
localhost
127.0.0.1
3002
127.0.0.1
4000

и поднимем локальный TCP-echo-сервер:

```

C
iralpraxis@primary:~/Documents/spbu/2023-spring/networks/spbu-masters-compnet/homeworks/12$ ncat -l -p 4000 --keep-open --exec "/bin/cat"

```

и проверим через telnet, что порт 3002 проксирует на порт 4000:


```
vitalpraxls@primary:~/Documents/spbu/2023-spring/networks/spbu-nasters-compnet/homeworks/12$ telnet localhost 3002
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
echoecho
echoecho
123
123
asd
asd
asd
asd
124
124
152412kjsdofjnelowskjrf
152412kjsdofjnelowskjrf
█
```