

## Практика 12. Сетевой уровень (сдать до 15.05.2022)

### 1. RIP (9 баллов)

#### Задание А (6 баллов)

Реализуйте эмулятор работы протокола RIP в виде консольного приложения.

Ваша автономная сеть (АС) из маршрутизаторов может быть сконфигурирована на основе файла (например, .json или .xml) либо генерироваться случайным образом каждый раз при запуске.

Каждый маршрутизатор должен иметь свой уникальный IP адрес. Это приложение не предполагает передачу данных по сети, поэтому IP адреса, как и связи между маршрутизаторами, могут быть произвольными.

Программа должна корректно работать с произвольной АС.

В конце работы программы для каждого маршрутизатора должна быть выведена таблица маршрутизации. Пример таблицы:

```
Final state of router 198.71.243.61 table:
```

[Source IP]	[Destination IP]	[Next Hop]	[Metric]
198.71.243.61	122.136.243.149	42.162.54.248	4
198.71.243.61	157.105.66.180	42.162.54.248	2
198.71.243.61	229.28.61.15	42.162.54.248	3
198.71.243.61	42.162.54.248	42.162.54.248	1

Приведите скрин или лог работы программы.

#### Задание Б (1 балл)

Выведите на консоль промежуточные этапы работы протокола: по каждому маршрутизатору должна быть выведена его текущая таблица маршрутизации.

```
Simulation step 3 of router 42.162.54.248
```

[Source IP]	[Destination IP]	[Next Hop]	[Metric]
42.162.54.248	122.136.243.149	157.105.66.180	3
42.162.54.248	157.105.66.180	157.105.66.180	1
42.162.54.248	229.28.61.15	157.105.66.180	2
42.162.54.248	198.71.243.61	198.71.243.61	1

#### Задание В (2 балла)

Реализуйте имитацию работы маршрутизаторов в виде отдельных потоков на примере приложения, рассмотренного на занятии.

### 2. Скорость передачи (6 баллов)

Реализуйте программу, которая измеряет скорость передачи информации по протоколам TCP и UDP, а также выводит количество потерянных пакетов.

Программа состоит из двух частей: клиента и сервера. Клиент создает трафик случайным образом (т.е. генерируется случайная последовательность данных) и отправляет их на сервер. Сервер подсчитывает количество полученных данных и выводит результат. Время отправки указывается клиентом вместе с данными.

Ваше приложение должно иметь GUI. Приложите скрин работы программы.

#### 1. Измерение по протоколу TCP (3 балла)

Пример интерфейса

Получатель TCP

Введите IP: 127.0.0.1

Выберите порт для получения: 123

Скорость передачи: 1,935 B/S

Число полученных пакетов: 5 of 5

Получить

Отправитель TCP

Введите IP адрес получателя: 127.0.0.1

Выберите порт отправки: 123

Введите количество пакетов для отправки: 5

Отправить пакеты

## 2. Измерение по протоколу UDP (3 балла)

Получатель UDP

Введите IP для получения: 127.0.0.1

Выберите порт получения: 123

Число полученных пакетов: 5 of 5

Скорость соединения: 0,715 KB/S

Получить

Отправитель UDP

Введите IP для получения: 127.0.0.1

Выберите порт получения: 123

Введите число пакетов для отправки: 5

Отправить

## 3. Транслятор портов (6 баллов)

Разработать приложение – транслятор портов. Трансляция осуществляется в соответствии с набором правил трансляции, заданных в конфигурационном файле. Каждое правило должно указывать, с какого порта на какие IP адрес и порт транслировать. При изменении конфигурационного файла новые правила должны вступать в действие, но установленные соединения не должны разрываться. В программе должен быть реализован GUI.

Пример GUI:

Транслятор портов

Управление  
IP: 192.168.1.33

Получение IP  
Имя хоста: yandex.ru

Запустить транслятор

IP: 213.180.193.11

Сохранять порт

Название	Внутренний IP	Внутренний порт	Внешний IP	Внешний порт
Yandex	127.0.0.1	12345	93.158.134.11	80
Rambler	127.0.0.1	123	81.19.70.3	80
Mail	127.0.0.1	500	217.69.139.201	80

Приложите скрин работы программы.