

PSI 22Z Zad 1.x

Uruchomienie kontenerów

- Uruchamianie serwera `./server_py.sh` lub `./server_c.sh`
- Uruchamianie klienta `./client_py.sh` lub `./client_c.sh`. Dodatkowo do klienta wymagany jest argument `server` "c" albo "py", czyli np `./client_py.sh server=c`

Zadanie 1.1

Python

server

```
$ ./server_py.sh
UDP server up and listening
Waiting for data...
Message from Client: b'Hello UDP Server'
Client IP Address: ('172.21.21.3', 44075)
Waiting for data...
Message from Client: b'Hello UDP Server'
Client IP Address: ('172.21.21.3', 44075)
Waiting for data...
Message from Client: b'Hello UDP Server'
Client IP Address: ('172.21.21.3', 44075)
Waiting for data...
```

client

```
$ ./client_py.sh server=py
Message from Server "Hello UDP Client"
Message from Server "Hello UDP Client"
Message from Server "Hello UDP Client"
```

C

server

```
$ ./server_c.sh
UDP server up and listening
Waiting for data...Received packet from 172.21.21.3:32866
Data: Hello UDP Server
Waiting for data...Received packet from 172.21.21.3:32866
Data: Hello UDP Server
Waiting for data...Received packet from 172.21.21.3:32866
Data: Hello UDP Server
Waiting for data...
```

client

```
$ ./client_c.sh server=c
Message received from the server: "Hello UDP Client"
Message received from the server: "Hello UDP Client"
Message received from the server: "Hello UDP Client"
```

Działa również wysyłka "międzyplatformowa", czyli kombinacje client_py + server_c oraz client_c + server_py.

Zadanie 1.2

Przygotowaliśmy program, który wysyła datagramy UDP o przyrastającej wielkości. Maksymalny rozmiar danych, który udało się nam wysłać to **65507** bajtów.

- Rozmiar ten wynika z tego, że datagram jest wysyłany w jednym pakiecie IPv4, którego maksymalny rozmiar wynosi $2^{16} - 1 = 65535$ bajtów
- Zgodnie ze specyfikacją pakietu IPv4 minimalna wielkość nagłówków w pakiecie IPv4 to 20 bajtów
- Nagłówki datagramu UDP zajmują kolejne 8 bajtów

Stąd maksymalny rozmiar danych wynosi: $65535B - 20B - 8B = 65507B$