

Zadanie na zajęciach

Autor rozwiązań: Maciej Sikora

Link do wersji online z poniższego krótkiego sprawozdania, na tym linku działają video.

<https://www.notion.so/Zadanie-na-zaj-ciach-17c5bfb1d44b45c98bc236d5164bd9da>

Task 1

Java Server

```
System.out.println("JAVA UDP SERVER");
DatagramSocket socket = null;
int portNumber = 9008;
socket = new DatagramSocket(portNumber);

while(true) {
    byte[] receiveBuffer = new byte[1024];
    Arrays.fill(receiveBuffer, (byte)0);
    DatagramPacket receivePacket = new DatagramPacket(receiveBuffer, receiveBuffer.length);
    socket.receive(receivePacket);
    String msg = new String(receivePacket.getData());
    System.out.println("received msg: " + msg);

    byte[] sendBuffer = "SERVER-SEND".getBytes();
    DatagramPacket sendPacket = new DatagramPacket(sendBuffer, sendBuffer.length, receivePacket.getAddress(), receivePacket.getPort());
    System.out.println("sending...");
    socket.send(sendPacket);
}
```

Dodana została sekcja związana z wysyłaniem wiadomości zwrotnej. Na szczególną uwagę zasługuje

```
DatagramPacket sendPacket = new DatagramPacket(sendBuffer, sendBuffer.length,
receivePacket.getAddress(), receivePacket.getPort());
```

W tym fragmencie widać, że informacje o nadawcy wiadomości do serwera (czyli adresata wiadomości zwrotnej) wyciągamy stricte z wiadomości którą otrzymaliśmy.

Java Client

```
System.out.println("JAVA UDP CLIENT");
DatagramSocket socket = null;
int portNumber = 9008;

socket = new DatagramSocket();
InetAddress address = InetAddress.getByName("localhost");
byte[] sendBuffer = "CLIENT-SEND".getBytes();
while(true){
    DatagramPacket sendPacket = new DatagramPacket(sendBuffer, sendBuffer.length, address, portNumber);
    socket.send(sendPacket);

    byte[] receiveBuffer = new byte[1024];
    Arrays.fill(receiveBuffer, (byte)0);
    DatagramPacket receivePacket = new DatagramPacket(receiveBuffer, receiveBuffer.length);
    socket.receive(receivePacket);
    System.out.println("client received msg: " + new String(receivePacket.getData()));
}
```

Tutaj jedyne dodajemy fragment z otrzymywaniem "potwierdzenia" od serwera, że otrzymał on pakiet od nas i nic więcej (tj nic o poprawności itd.)

Poniżej krótkie video prezentujące działanie zadania.

https://s3-us-west-2.amazonaws.com/secure.notion-static.com/30a77d33-c57f-4291-a36e-48ee2a1b482d/2021-03-01_16-58-24.mkv

Task 2

Java server

```
socket = new DatagramSocket(portNumber);
while(true) {
    byte[] receiveBuffer = new byte[1024];
    Arrays.fill(receiveBuffer, (byte)0);
    DatagramPacket receivePacket = new DatagramPacket(receiveBuffer, receiveBuffer.length);
    socket.receive(receivePacket);
    String msg = new String(receivePacket.getData(), "UTF-8");
    System.out.println("received msg: " + msg);
}
```

Python client

```
import socket

serverIP = "127.0.0.1"
serverPort = 9008
msg = "Żółć niedźwiedzia/żółta kaczka - PYTHON-CLIENT"

print('PYTHON UDP CLIENT')
client = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
client.sendto(bytes(msg.encode('utf-8')), (serverIP, serverPort))
```

Na powyższych fragmentach kodów obie aplikacje mają zastosowane kodowanie utf-8 do swoich wiadomości.

Ponownie krótkie video.

https://s3-us-west-2.amazonaws.com/secure.notion-static.com/cc7c4de9-ef04-4196-87b2-ce894ed7e4d7/2021-03-01_17-13-11.mkv

Task 3

Java Server

```
package task3;

import java.net.DatagramPacket;
import java.net.DatagramSocket;
import java.nio.ByteBuffer;
import java.nio.ByteOrder;
import java.util.Arrays;

public class JavaUdpServer {

    public static void main(String args[])
    {
        System.out.println("JAVA UDP SERVER");
        DatagramSocket socket = null;
        int portNumber = 9008;

        try{
            socket = new DatagramSocket(portNumber);

            while(true) {
                byte[] receiveBuffer = new byte[1024];
                Arrays.fill(receiveBuffer, (byte)0);
                DatagramPacket receivePacket = new DatagramPacket(receiveBuffer, receiveBuffer.length);
                socket.receive(receivePacket);
                int msg = ByteBuffer.wrap(receivePacket.getData()).order(ByteOrder.LITTLE_ENDIAN).getInt();
                System.out.println("received msg: " + msg);
            }
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

```

        byte[] sendBuffer = ByteBuffer.allocate(4).order(ByteOrder.LITTLE_ENDIAN).putInt(msg+1).array();
        DatagramPacket sendPacket = new DatagramPacket(sendBuffer, sendBuffer.length, receivePacket.getAddress(), receivePacket.getPort());
        System.out.println("sending " + (msg+1));
        socket.send(sendPacket);
    }
}
catch(Exception e){
    e.printStackTrace();
}
finally {
    if (socket != null) {
        socket.close();
    }
}
}
}

```

Python Client

```

import socket

serverIP = "127.0.0.1"
serverPort = 9008

print('PYTHON UDP CLIENT')
client = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
msg_bytes = (300).to_bytes(4, byteorder='little')
print(f"sending {int.from_bytes(msg_bytes, byteorder='little')}")
client.sendto(msg_bytes, (serverIP, serverPort))
buff = []

buff, addr = client.recvfrom(1024)
print(f"received {int.from_bytes(buff, byteorder='little')}")

```

Poniżej video

https://s3-us-west-2.amazonaws.com/secure.notion-static.com/9b2929dd-a724-4659-8c08-4908aa3b9e53/2021-03-01_17-36-47.mkv