

# MREŽNO PROGRAMIRANJE

## Vježba 4

### ZADATAK 1

Osnovni klijent-server program iz laboratorijske vježbe 2 prilagodite na način da radi preko UDP protokola. Kod i screenshot iz konzole priložite vježbi.

*FILE1: tcp\_server.py*

```
import socket

UDP_IP = "127.0.0.1"
UDP_PORT = 5005

sock = socket.socket(socket.AF_INET, socket.SOCK_DGRAM) # UDP
sock.bind((UDP_IP, UDP_PORT))

while True:
    data, addr = sock.recvfrom(1024) # buffer size is 1024 bytes
    print("received message: %s" % data)
```

*FILE2: tcp\_client.py*

```
import socket

UDP_IP = "127.0.0.1"
UDP_PORT = 5005
MESSAGE = b"Hello, World!"

print("UDP target IP: %s" % UDP_IP)
print("UDP target port: %s" % UDP_PORT)
print("message: %s" % MESSAGE)

sock = socket.socket(socket.AF_INET, socket.SOCK_DGRAM) # UD
sock.sendto(MESSAGE, (UDP_IP, UDP_PORT))
```

```

tcp_server.py — vjezba4
tcp_client.py U
tcp_server.py U x
tcp_server.py > ...
1 import socket
2
3 UDP_IP = "127.0.0.1"
4 UDP_PORT = 5005
5
6 sock = socket.socket(socket.AF_INET, socket.SOCK_DGRAM) # UDP
7 sock.bind((UDP_IP, UDP_PORT))
8
9 while True:
10     data, addr = sock.recvfrom(1024) # buffer size is 1024 byt
11     print("received message: %s" % data)

```

```

vjezba4 git:(master) x python3 tcp_server.py
received message: b'Hello, World!'

```

```

vjezba4 git:(master) x python3 tcp_client.py
UDP target IP: 127.0.0.1
UDP target port: 5005
message: b'Hello, World!'
vjezba4 git:(master) x

```

## ZADATAK 2 - ECHO CLIENT-SERVER

Osnovni klijent-server program iz laboratorijske vježbe 2 prilagodite na način da radi preko UDP protokola. Kod i screenshot iz konzole priložite vježbi.

**FILE1:** *echo\_server.py*

```

# -- coding: utf-8 --
# echo_server.py
import socket

host = socket.gethostname()
port = 12345
echo_server = socket.socket() # tcp socket

echo_server.bind((host, port))
echo_server.listen(5)

print("Cekam klijenta!")

```

```
conn, addr = echo_server.accept() # prihvaćanje konekcije kada
se klijent spoji

print("Spojen: ", addr)

while True:
    data = conn.recv(1024) # prihvaćanje podataka od klijenta
    if not data: # ako nema podataka, izađi
        break
    conn.sendall(data) # vrati primljene podatke klijentu

conn.close()
```

***FILE2: echo\_client.py***

```
# -- coding: utf-8 --
# echo_client.py
import socket

host = socket.gethostname()
port = 12345
client_socket = socket.socket() # tcp socket

client_socket.connect((host, port))

client_socket.sendall("Tekst koji se salje serveru".encode())

data = client_socket.recv(1024) # tekst koji je primljen od
servera

print(data)
client_socket.close()
```

The screenshot shows a VS Code editor with a file explorer on the left and a terminal at the bottom. The file explorer shows a project named 'VJEZBA4' with files like 'echo\_client.py', 'echo\_server.py', 'tcp\_client.py', and 'tcp\_server.py'. The main editor window displays the code for 'echo\_client.py'.

```

1 # -- coding: utf-8 --
2 # echo_client.py
3 import socket
4
5 host = socket.gethostname()
6 port = 12345
7 client_socket = socket.socket() # tcp socket
8
9 client_socket.connect((host, port))
10
11 client_socket.sendall("Tekst koji se salje serveru".encode())
12
13 data = client_socket.recv(1024) # tekst koji je primljen od s
14
15 print(data)
16 client_socket.close()

```

The terminal at the bottom shows the execution of the server and client programs. The server output is:

```

vjezba4 git:(master) x python3 echo_server.p
y
Cekam klijenta!
Spojen: ('192.168.1.101', 62620)
vjezba4 git:(master) x

```

The client output is:

```

vjezba4 git:(master) x python3 echo_client.p
y
b'Tekst koji se salje serveru'
vjezba4 git:(master) x

```

Ove programe prilagodite na sljedeći način:

- U svaki kod treba napraviti import datetime modula, te napraviti sljedeću liniju koda:
  - `print(datetime.datetime.now())`
- `local_machine_info.py` program iz prve vježbe treba importirati i pozvati funkciju koja će ispisati hostname i ip adresu.
  - `from local_machine_info import print_machine_info`
- U klijent programu, umjesto slanja teksta, napravite da se korisnik programa pita tekst za unos, te se onda taj isti tekst pošalje serveru koji ga vrati natrag i klijent ga ispiše. Koristite `raw_input` funkciju koja prima bilo kakav unos od strane korisnika i sprema ga u string.
- Ukoliko korisnik unese string 'vaše\_ime\_prezime' server treba vratiti klijentu obavijest da taj unos nije podržan.
- Također, program prilagodite na način da server cijelo vrijeme sluša na dolazne konekcije, a ne da izađe nakon što primi i zatvori konekciju.

*FILE1: echo\_server2.py*

```
# -- coding: utf-8 --
# echo_server.py
import socket
import datetime
from local_machine_info import print_machine_info

print_machine_info()
print(datetime.datetime.now())

host = socket.gethostname()
port = 12345
echo_server = socket.socket() # tcp socket

echo_server.bind((host, port))
echo_server.listen(5)

while True:
    conn, addr = echo_server.accept()
    print("Spojen: ", addr)
    data = conn.recv(1024) # prihvaćanje podataka od klijenta
    if data == b"matea_beslic":
        data2 = "Unos nije podrzan"
        conn.sendall(data2.encode())
        break
    else:
        conn.sendall(data) # vrati primljene podatke klijentu

conn.close()
```

**FILE2: echo\_client2.py**

```
# -- coding: utf-8 --
# echo_client.py
import socket
import datetime
from local_machine_info import print_machine_info

print_machine_info()
print(datetime.datetime.now())

host = socket.gethostname()
port = 12345
client_socket = socket.socket() # tcp socket

client_socket.connect((host, port))

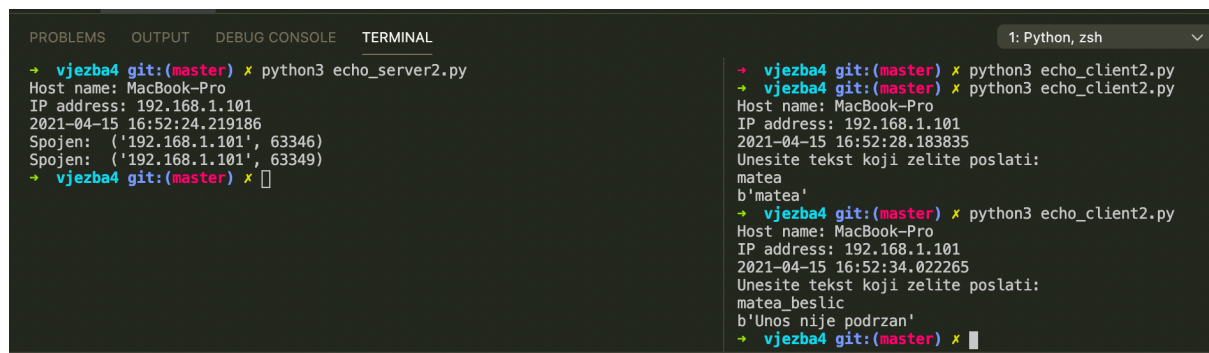
print("Unesite tekst koji zelite poslati: ")
data = input()

client_socket.sendall(data.encode())

data = client_socket.recv(1024) # tekst koji je primljen od
servera

print(data)

client_socket.close()
```



```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
+ vjezba4 git:(master) x python3 echo_server2.py
Host name: MacBook-Pro
IP address: 192.168.1.101
2021-04-15 16:52:24.219186
Spojen: ('192.168.1.101', 63346)
Spojen: ('192.168.1.101', 63349)
+ vjezba4 git:(master) x
+ vjezba4 git:(master) x python3 echo_client2.py
+ vjezba4 git:(master) x python3 echo_client2.py
Host name: MacBook-Pro
IP address: 192.168.1.101
2021-04-15 16:52:28.183835
Unesite tekst koji zelite poslati:
matea
b'matea'
+ vjezba4 git:(master) x python3 echo_client2.py
Host name: MacBook-Pro
IP address: 192.168.1.101
2021-04-15 16:52:34.022265
Unesite tekst koji zelite poslati:
matea_beslic
b'Unos nije podrzan'
+ vjezba4 git:(master) x
```