

| | |
|------------------|--|
| Cognome | |
| Nome | |
| Matricola | |

Parte I – Programmazione Socket (3 punti)

Si vuole scrivere un'applicazione client/server UDP che riconosca parole palindroma (cioè parole che si leggono uguali anche al contrario). Il client chiede all'utente di inserire una stringa, il server risponde indicando i) la lunghezza della stringa e ii) se si tratta di una stringa palindroma o meno.

UDP server

```
from socket import *

def riconosci_palindroma(stringa):
    indice = (len(stringa) - 1)
    nuova_stringa = ""
    while indice >= 0:
        nuova_stringa += stringa[indice]
        indice -= 1
    return nuova_stringa

serverPort = 12000
serverSocket = socket(AF_INET, SOCK_DGRAM)

serverSocket.bind(('', serverPort))

print('Server pronto a ricevere!')
while 1:
    messaggio, clientAddress = serverSocket.recvfrom(2048)
    messaggio = messaggio.decode('utf-8')

    print('La stringa passata ha '+str(len(messaggio))+ ' caratteri...')
    test = riconosci_palindroma(messaggio)
    if test == messaggio:
        risposta = 'OK'
    else:
        risposta = 'KO'
    serverSocket.sendto(risposta.encode('utf-8'), clientAddress)
```

UDP client

```
from socket import *
serverName = 'localhost'
serverPort = 12000
clientSocket = socket(AF_INET, SOCK_DGRAM)
clientSocket.settimeout(5)

message = input('Inserisci una parola:')
clientSocket.sendto(message.encode('utf-8'), (serverName, serverPort))
try:
    reply, serverAddress = clientSocket.recvfrom(2048)
    if reply.decode('utf-8') == 'OK':
        print('La stringa passata è palindroma!')
    else reply.decode('utf-8') == 'KO':
        print('Mi dispiace, la stringa inserita non è palindroma.')
except:
    print("Il server non ha risposto entro il timeout...")
finally:
    clientSocket.close()
```

Q1) (vedi codice) Completare gli script "UDP client" e "UDP server" date le seguenti specifiche:

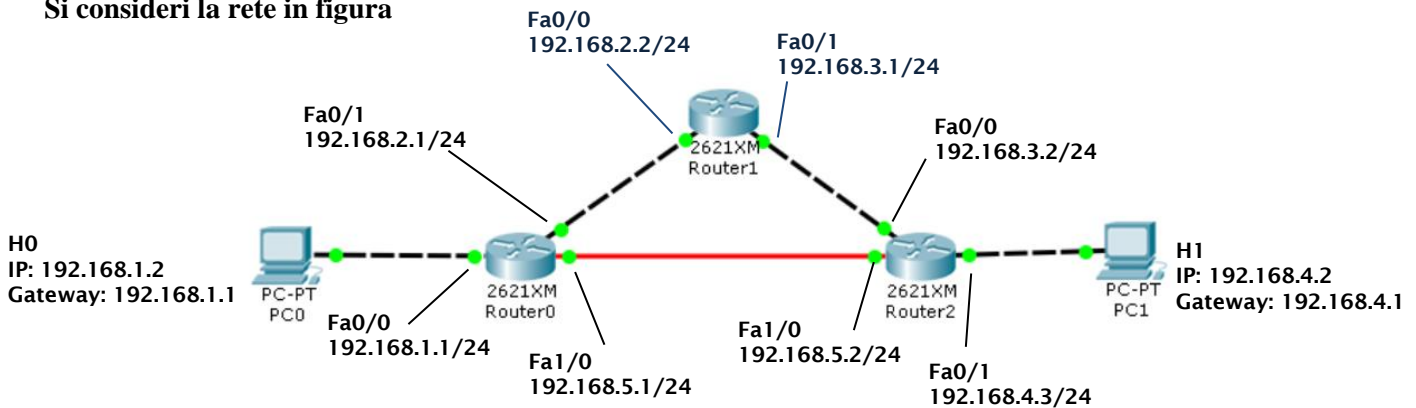
- Utilizzare indirizzi IPv4 **(0.5 punti)**
- Time-out in ricezione (lato client): 5 secondi. **(0.5 punti)**
- Lunghezza buffer di ricezione: 2048 byte **(0.5 punti)**

Q2) (vedi codice) Completare la parte mancante nel ciclo while di "UDP server" affinché il servizio funzioni correttamente **(1 punto)**

Q3) Con quale messaggio risponde il server se l'utente dell'applicazione client inserisce in input '12321'? **(0.5 punti)**

La risposta del server è: 'OK'

Si consideri la rete in figura



Attenzione:

- Gli Indirizzi IP e i gateway sono già stati configurati per gli host H0 e H1.
 - Le interfacce del router Router 1 non sono state ancora configurate mentre le interfacce del Router 0 e le interfacce del Router 2 sono state configurate.
- 1) Configurare gli indirizzi e attivare le due interfacce del router Router 1 e salvare la configurazione corrente in modo che sia disponibile al prossimo riavvio del router. (1 punto)
 - 2) RIP v2 e' stato configurato correttamente su Router 1 e su Router 2 mentre per Router 0 non ancora. Configurare RIP v2 sul Router 0. (1 punto)
 - 3) Ancora non si può verificare la connettività della rete tramite H0 a H1. Trova il problema e suggerisci le modifiche da apportare per garantire la connettività della rete. (1 punto)

1) Router1>enable

```
Router1#configure terminal
Router1(config)#interface Fa0/0
Router1(config-if)#ip 192.168.2.2 255.255.255.0
Router1(config-if)#no shutdown
Router1(config-if)#exit
Router1(config)#interface Fa0/1
Router1(config-if)#ip 192.168.3.1 255.255.255.0
Router1(config-if)#no shutdown
Router1(config-if)#exit
Router1(config)#exit
Router1#copy running-config startup-config
```

2) R0> enable

```
R0# configure terminal
R0(config)# router rip
R0(config-router)# version 2
R0(config-router)# network 192.168.1.0
R0(config-router)# network 192.168.2.0
R0(config-router)# network 192.168.5.0
```

3) Il gateway di default di H1 è su 192.168.4.1 mentre l'indirizzo dell'unico router connesso a H1 è 192.168.4.3. Impostare il default gateway su 192.168.4.3