

Cognome	BRAVO	
Nome	STUDENTE	
Matricola	SOLUZIONE	

Parte I – Programmazione Socket (3 punti)

Si vuole scrivere un'applicazione client/server UDP per conteggiare il numero di consonanti presenti in una stringa.

Il client chiede all'utente di inserire una stringa, il server risponde indicando il numero di consonanti presenti nella stringa (sia maiuscole che minuscole).

Hint: `y.count(x)` conta quante volte appare l'elemento `x` nella lista `y`.

UDP server

```
from socket import *

serverPort = 12000
serverSocket = socket(AF_INET, SOCK_DGRAM)
serverSocket.bind(('', serverPort))

print('Server pronto a ricevere!')

vocali = ['A', 'E', 'I', 'O', 'U']

while 1:
    messaggio, clientAddress = serverSocket.recvfrom(2048)
    messaggio = messaggio.decode('utf-8')
    num = len(messaggio)
    for voc in vocali:
        num = num - messaggio.count(voc)
    risposta = "Il messaggio contiene "+str(num)+" consonanti."
    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 (senza caratteri speciali):')
clientSocket.sendto(message.encode('utf-8'), (serverName, serverPort))

try:
    reply, serverAddress = clientSocket.recvfrom(2048)
    print(reply.decode('utf-8'))
except:
    print("Il server non ha risposto entro il timeout...")
finally:
    clientSocket.close()
```

Q1) 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) Con quale messaggio risponde il server se il client invia la stringa "SOcKet" ?

Indicare se la risposta del server è corretta per lo specifico caso in cui sia stata inviata la stringa "SOcKet" e, nel caso non lo fosse, spiegare brevemente come si potrebbe modificare il codice lato server per renderlo corretto. **(1.5 punti)**

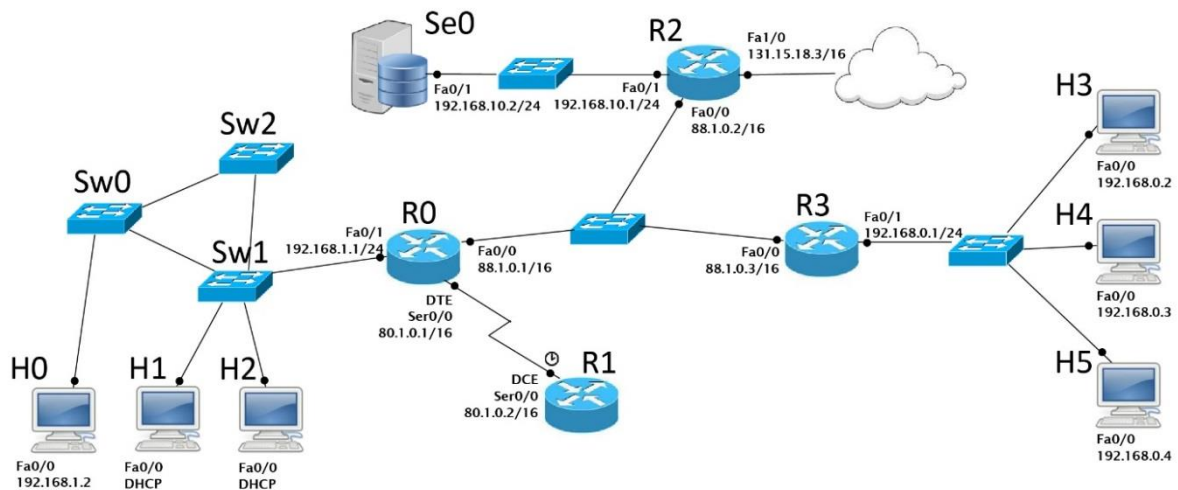
La risposta del server è: Il messaggio contiene 5 consonanti.

Non è conforme alle specifiche del programma perché il server non riconosce le vocali minuscole, considerandole delle consonanti.

Per rendere il server conforme è possibile estendere la lista *vocali* aggiungendo le vocali in minuscolo. Oppure, prima del conteggio, convertire in maiuscolo la stringa *messaggio* ricevuta dal client.

Parte II – Packet Tracer (3 punti)

Si consideri la rete in figura



Attenzione:

- Indirizzi IP e gateway sono già stati configurati per i 6 host.
- Le interfacce dei router R0, R1 e R3 sono già state configurate ed attivate come in figura.
- Le reti /24 sono reti private
- Indicare sempre prima del comando il prompt visualizzato dal sistema, prestando attenzione alla modalità di partenza in ciascuna richiesta

1) Configurare ed attivare le interfacce Fa0/0 e Fa0/1 del router R2 come in figura. (1 punto)

```
R2> enable
R2# configure terminal
R2(config)# interface Fa0/0
R2(config-if)# ip address 88.1.0.2 255.255.0.0
R2(config-if)# no shutdown
R2(config-if)# exit
R2(config)# interface Fa0/1
R2(config-if)# ip address 192.168.10.1 255.255.255.0
R2(config-if)# no shutdown
R2(config-if)# exit
```

2) Abilitare e configurare il protocollo RIPv2 sul router R0. (1 punto)

```
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 80.1.0.0
R0(config-router)# network 88.1.0.0
```

3) Configurare R0 come server DHCP in modo tale che agli host che si collegano alla sottorete (H1 e H2) venga assegnata automaticamente una configurazione IP. (1 punto)

```
R0(config)# ip dhcp pool pool_name
R0(dhcp-config)# default-router 192.168.1.1
R0(dhcp-config)# network 192.168.1.0 255.255.255.0
```