

**25/01/2021**

## **Laboratorio Python Socket Programming (3 punti)**

Il codice sotto riportato è relativo ad una versione semplificata di un'applicazione che fornisce le fasce di rischio (Giallo, Arancione e Rosso) di alcune regioni italiane come disposto dall'ultimo DPCM relativo alle norme anti-COVID19. L'utente che accede all'applicazione specifica la regione di cui desidera avere informazioni e riceve l'informazione richiesta, quando disponibile. Si noti che ad ogni richiesta del client, il server verifica la presenza di DPCM più aggiornati e tenta di aggiornare (fino a massimo 3 tentativi) il proprio database tramite la funzione "update\_info".

Si completino le seguenti richieste:

1) Completare il codice del client e del server assumendo che (1.5 punti):

- La comunicazione avvenga su IPv4+TCP
- Il server sia in ascolto all'indirizzo 192.168.0.23 sulla porta 2021
- La dimensione del buffer in ricezione sia 1024

2) Scrivere nello spazio sottostante nel SERVER, il codice mancante per avvisare il client dell'errore nell'aggiornamento dei dati (1.5 punto)

### **Script Client:**

```
from socket import *
```

```
clSocket = socket(AF_INET, SOCK_STREAM)
```

```
clSocket.connect(('192.168.0.23', 2021))
```

```
while True:
```

```
    regione = input("Inserisci la Regione: ")
```

```
    clSocket.send(regione.encode('utf-8'))
```

```
    resp = clSocket.recv(1024)
```

```
    resp = resp.decode('utf-8')
```

```
    if resp == 'OD':
```

```
        stop = input("Errore nell'aggiornamento del database. Le informazioni potrebbero non essere aggiornate. Interrompere? (SI/NO):")
```

```
        clSocket.send(stop.encode('utf-8'))
```

```
        if stop == 'NO':
```

```
            resp = clSocket.recv(1024)
```

```
            resp = resp.decode('utf-8')
```

```
            print("Il colore della regione richiesta è: ", resp)
```

```
        else:
```

```
            print('Arrivederci')
```

```
            break
```

```
    elif resp == 'ND':
```

```
        print("Regione non disponibile")
```

```
    else:
```

```
        print("Il colore della regione richiesta è: ", resp)
```

```
keep = input("Continuare con un'altra regione? (SI/NO): ")
```

```
clSocket.send(keep.encode('utf-8'))
```

```
if keep == "NO":
```

```
    print('Arrivederci')
```

```
        break
    clSocket.close()
```

### Script Server:

```
from socket import *
from app_library import update

def update_info(dpcm, regione):
    dpcm[regione], http_code = update(dpcm, regione)

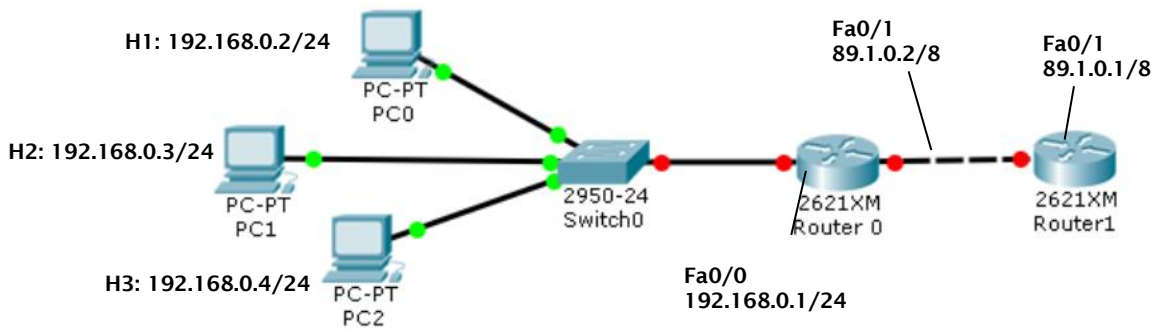
    return dpcm, http_code

dpcm = {"Lombardia": 'Rosso', "Veneto": 'Arancione',
        "Lazio": 'Arancione', "Toscana": 'Giallo',
        "Campania": 'Giallo', "Sicilia": 'Rosso'}
socket = socket(AF_INET, SOCK_STREAM)
socket.bind(('', 2021))
socket.listen(1)
while True:
    clSocket, clAddr = socket.accept()
    keep = True
    stop = 'NO'
    while keep:
        regione = clSocket.recv(1024)
        regione = regione.decode('utf-8')
        if regione in dpcm:
            attempt = 0
            code = 0
            while (code!=200 and attempt < 4):
                dpcm, code = update_info(dpcm, regione)
                attempt += 1
            if code!=200:
                clSocket.send('OD'.encode('utf - 8'))
                stop = clSocket.recv(1024)
                if stop.decode('utf - 8') == 'NO':
                    clSocket.send(dpcm[regione].encode('utf - 8'))
                else:
                    keep = False
            else:
                clSocket.send(dpcm[regione].encode('utf - 8'))
        else:
            clSocket.send('ND'.encode('utf-8'))
    if stop == 'NO':
        r = clSocket.recv(1024)
        if r.decode('utf - 8') == 'NO':
            keep = False
```

clSocket.close()

## **Laboratorio Packet Tracer (3 punti)**

Si consideri la rete in figura



Attenzione:

- Gli Indirizzi IP e gateway sono già stati configurati per gli host H1, H2 e H3
  - Le interfacce del router Router 0 non sono state ancora configurate.
  - L'interfaccia Fa0/1 del router Router 1 è già stata configurata.
- 1) Configurare gli indirizzi e attivare le due interfacce del router Router 0. (1 punto)
  - 2) Configurare routing statico sul router Router 1. (1 punto)
  - 3) Salvare la configurazione corrente del Router 0 in modo che sia disponibile al prossimo riavvio del router. (0,5 punti)
  - 4) Che indirizzi di livello 3 hanno i pacchetti che vengono inviati sulla rete tra Router 0 e Router 1? (0,5 punti)

### **Soluzione**

- 1) Router0>enable  
Router0#configure terminal  
Router0(config)#interface Fa0/0  
Router0(config-if)#ip 192.168.0.1 255.255.255.0  
Router0(config-if)#no shutdown  
Router1(config-if)#exit  
Router1(config)#interface Fa0/1  
Router1(config-if)#ip 89.1.0.2 255.0.0.0  
Router1(config-if)#no shutdown  
Router1(config-if)#exit  
Router1(config)#exit
- 2) Router1:  
ip route 0.0.0.0 0.0.0.0 89.1.0.2
- 3) Router0#copy running-config startup-config
- 4) L'indirizzo ip del sorgente del pacchetto (NAT non e' abilitato)