

# **Esercitazione #1**

Antonio Brogi

Dipartimento di Informatica  
Università di Pisa

## Homework assignment #1

**Q1.** Supponiamo che un router A trasmetta un pacchetto su un collegamento con un router B, che la frequenza di trasmissione del collegamento sia 1 Mbps e che la velocità di propagazione sia  $2 \cdot 10^8$  m/s. Determinare –giustificando la risposta– quanto deve essere al più la lunghezza D del collegamento affinché i primi 1.000 bit del pacchetto siano arrivati a B dopo 2 millisecondi.

**E1.** Descrivere con uno pseudo-codice il comportamento di un server proxy quando riceve una richiesta GET di tipo “if-modified-since”. Assumere di avere a disposizione le operazioni:

```
connection TCPopen (IPAddress, int)    //per aprire una connessione
void TCPsend (connection, data)        //per spedire dati su una connessione
data TCPreceive (connection)           //per ricevere dati su una connessione
void TCPclose (connection)             //per chiudere una connessione
int TCPbind (int)                      //per rich. assegnaz. porta dove attendere rich. di conn.
void TCPunbind (int)                   //per liberare una porta
connection TCPaccept (int)             //per attendere richieste di connessione
```

**E2.** Descrivere con un automa a stati finiti il comportamento di un semplice server FTP che consente ai suoi clienti di eseguire solo i comandi LIST e QUIT dopo avere effettuato un accesso autenticato. Il server consente inoltre di inviare al più tre volte la password per ogni tentativo di login. Nella descrizione dell'automa utilizzare gli eventi

*USER u      PASS p      LIST      QUIT*

per indicare le richieste dei clienti.

**Q2.** Un host A vuole inviare un messaggio di email contenente come testo solo “OK”, indirizzato a un unico destinatario. Indicare – giustificando la risposta – quali messaggi SMTP A intercambia con il mailserver per inviare tale email.

**Q1.** Supponiamo che un router A trasmetta un pacchetto su un collegamento con un router B, che la frequenza di trasmissione del collegamento sia 1 Mbps e che la velocità di propagazione sia  $2 \cdot 10^8 \text{ m/s}$ . Determinare –giustificando la risposta– quanto deve essere al più la lunghezza D del collegamento affinché i primi 1.000 bit del pacchetto siano arrivati a B dopo 2 millisecondi.

$$\frac{10^3 \text{ b}}{10^6 \text{ b/s}} + \frac{D}{2 \cdot 10^8 \text{ m/s}} = 2 \cdot \frac{1}{10^3} \text{ s}$$

ovvero

$$D = 200 \text{ km}$$

**E1.** Descrivere con uno pseudo-codice il comportamento di un server proxy quando riceve una richiesta GET di tipo “if-modified-since”. Assumere di avere a disposizione le operazioni:

```
connection TCPopen (IPAddress, int)    //per aprire una connessione
void TCPsend (connection, data)         //per spedire dati su una connessione
data TCPreceive (connection)           //per ricevere dati su una connessione
void TCPclose (connection)             //per chiudere una connessione
int TCPbind (int)                      //per rich. assegnaz. porta dove attendere rich. di conn.
void TCPunbind (int)                   //per liberare una porta
connection TCPaccept (int)              //per attendere richieste di connessione
```

/\*

Assumiamo di disporre di:

```
getURL, getHost, getPort                //per estrarre URL, host, porta da richiesta HTTP
cacheLookup                             //per cercare URL in cache
refreshCache, updateCash                 //per aggiornare cache
valid(cached, request)                   //per confrontare data copia cache e richiesta
build304reply                           //per costruire risposta 304
isNotModified                           //per determinare tipo risposta HTTP
```

\*/

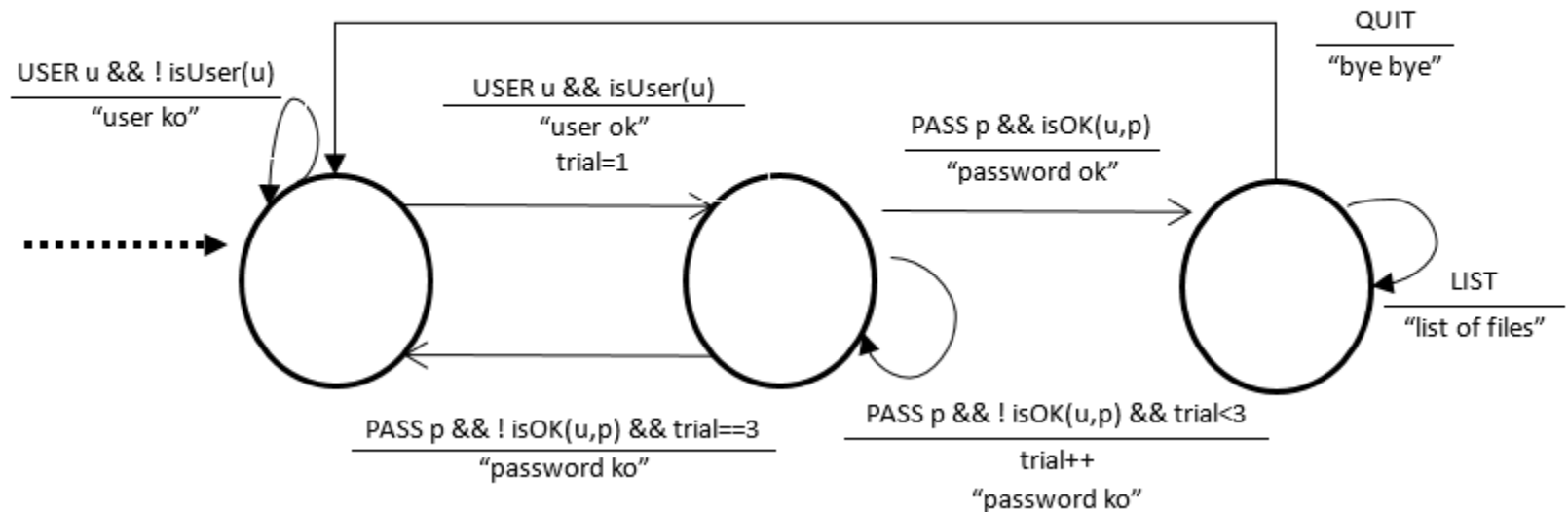
```
int p = TCPbind(8080);
while (true) do {
    c = TCPaccept(p);
    request = TCPreceive(c);
    URL = getURL(request);
    cached = cacheLookup(URL);
    if (cached != NULL && valid(cached, request))
        reply=build304reply(cached); ←
    else {
        s = TCPopen(getHost(request), getPort(request));
        TCPsend(s, request);
        reply = TCPreceive(s);
        TCPclose(s);
        if isNotModified(reply) refreshCash(URL);
        else updateCash(reply);
    }
    TCPsend(c, reply); TCPclose(c);
}
```

**E2.** Descrivere con un automa a stati finiti il comportamento di un semplice server FTP che consente ai suoi clienti di eseguire solo i comandi LIST e QUIT dopo avere effettuato un accesso autentificato. Il server consente inoltre di inviare al più tre volte la password per ogni tentativo di login. Nella descrizione dell'automa utilizzare gli eventi

*USER u    PASS p    LIST    QUIT*

per indicare le richieste dei clienti.

Indichiamo con  $\text{isUser}(u)$  la funzione che verifica se  $u$  è un utente del server e con  $\text{isOK}(u,p)$  la funzione che verifica se  $p$  è la password di  $u$ . Per semplicità indichiamo con "descrizione messaggio" ogni messaggio inviato dal server al cliente.



**Q2.** Un host A vuole inviare un messaggio di email contenente come testo solo “OK”, indirizzato a un unico destinatario. Indicare – giustificando la risposta – quali messaggi SMTP A intercambia con il mailserver per inviare tale email.

In totale 19\* messaggi SMTP:  
3 per apertura connessione  
+ 4 per busta  
+ 6 per intestazioni  
+ 1 per linea vuota  
+ 3 per corpo  
+ 2 per chiusura connessione.

\*14 se non vengono inviati i messaggi opzionali  
contenenti le intestazioni e la linea vuota

