



Mo	Tu	We	Th	Fr	Sa	Su
----	----	----	----	----	----	----

No. RETI DI TELE.

Date 17/12/19

Come è fatto lo scritto: (2 ore, 2 ore a mezza)

2 esercizi iniziali (8-9 PUNTI)

- ↳ SORGENTI (TOKEN/LENKI)
- ↳ GO BACK / SR o IP

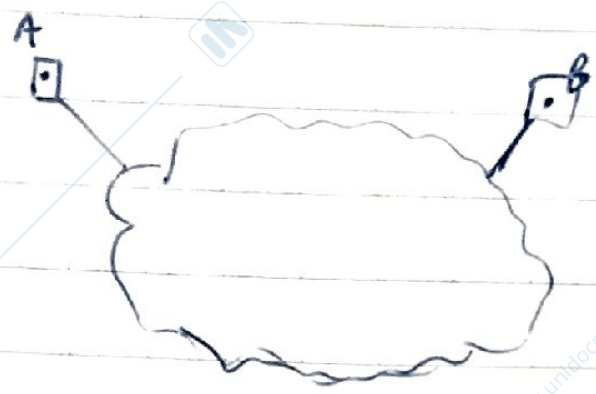
NON TCP → MM

1/2 ESERCIZI + PICCOLI  
(3-4 PUNTI)

DOMANDE DI TEORIA (NON SOLO PURA  
TEORIA)  
(8 PUNTI)

Mo	1	We	Th	Fr	Sa	Su
----	---	----	----	----	----	----

# TCP



Vogliamo  
 mettere  
 in comunicazione  
 i 2 processi  
 A e B  
 e tra i  
 terminali  
 c'è una  
 rete

multi  
 cammini  
 possibili

Riprenderemo i controlli di flusso = controllo

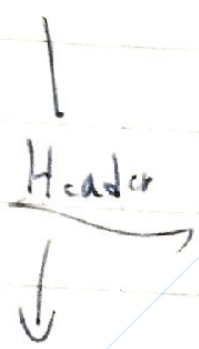
sulle sorgenti  
 affinché  
 non "attropi"  
 la destinazione

Controlli di congestione = controllo sulle  
 sorgenti affinché  
 non "attropi"  
 la rete

☀ ☁ ☂ 3

Mo	Tu	We	Th	Fr	Sa	Su
	X					

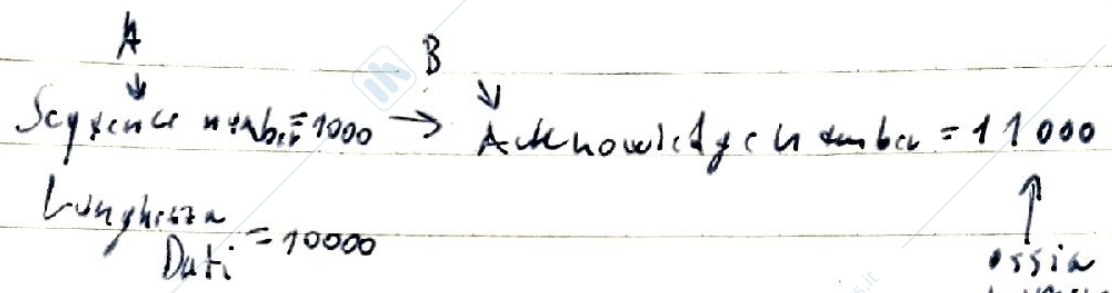
TCP → + affidabile



il numero presentato in caso

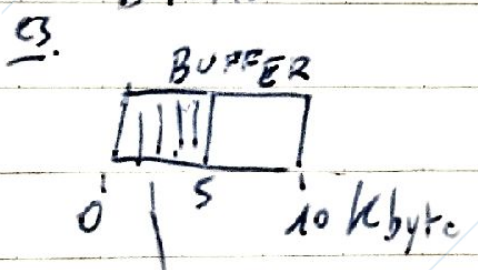
Sequence number: numeriamo il primo byte del pacchetto

Acknowledgment number: numero per il ricevente



↑  
ossia numero del secondo pacchetto che voglio che arrivi

Supponiamo che ci sia un buffer



il ricevitore dice nel tempo come se riempie il suo buffer

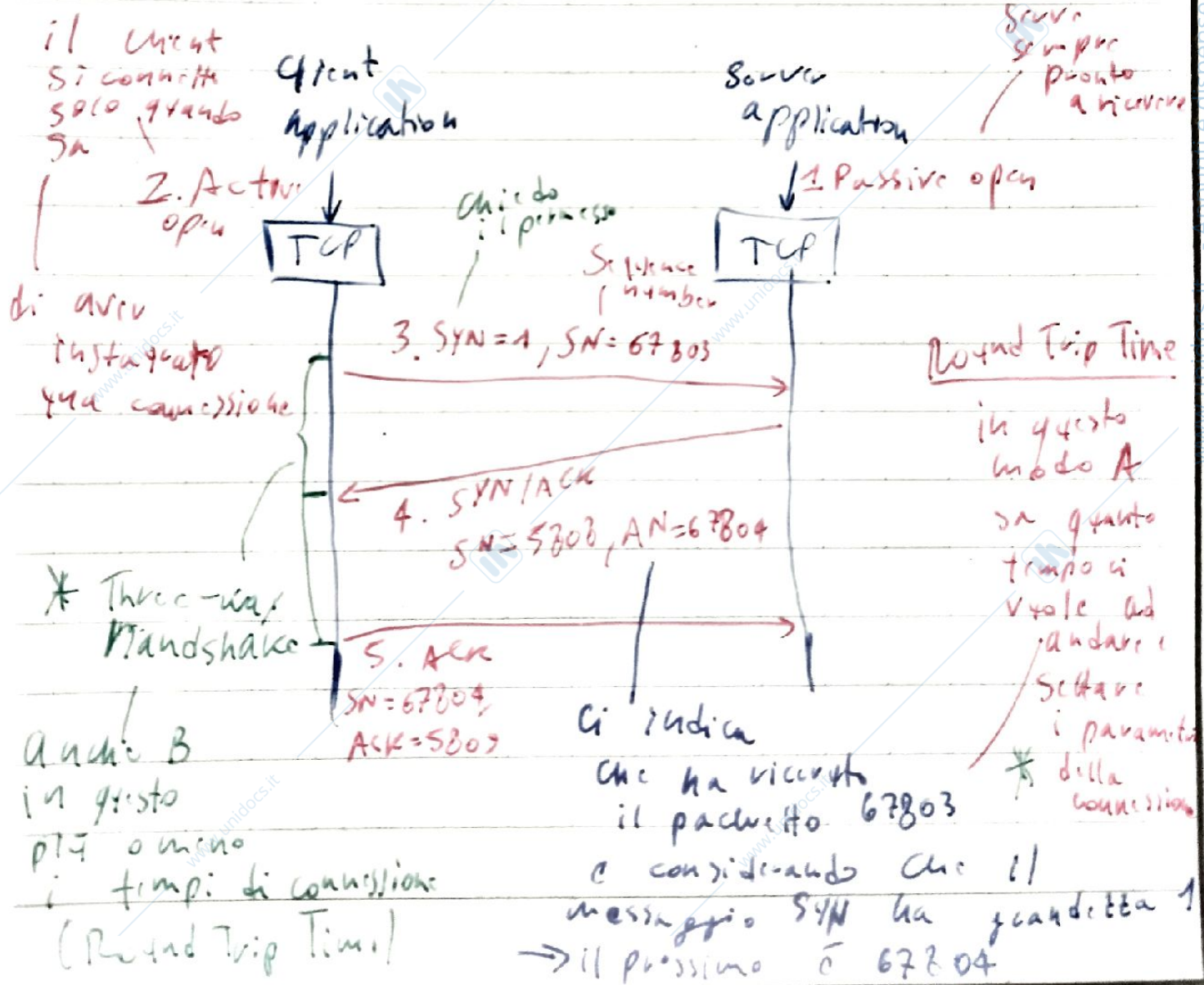
spazio occupato (vuoto nel tempo)



# TCP è Connection Oriented

↓  
prima del trasferimento di un flusso di dati occorre prima fare una connessione (SETUP)

## SETUP della Connessione



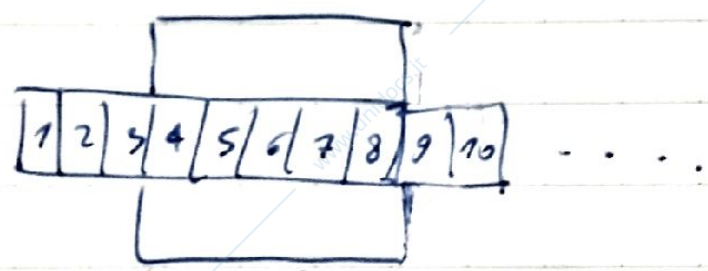
Quando uno dei 2 vuole chiudere la connessione (non voluta da entrambi) si usa Two-way handshake

↓  
in questo caso  
Three way handshake

Dopo aver instaurato la connessione abbiamo bisogno di gestire i buffer.

il concetto è lo stesso della doppia finestra a livello 2

differente vapp.

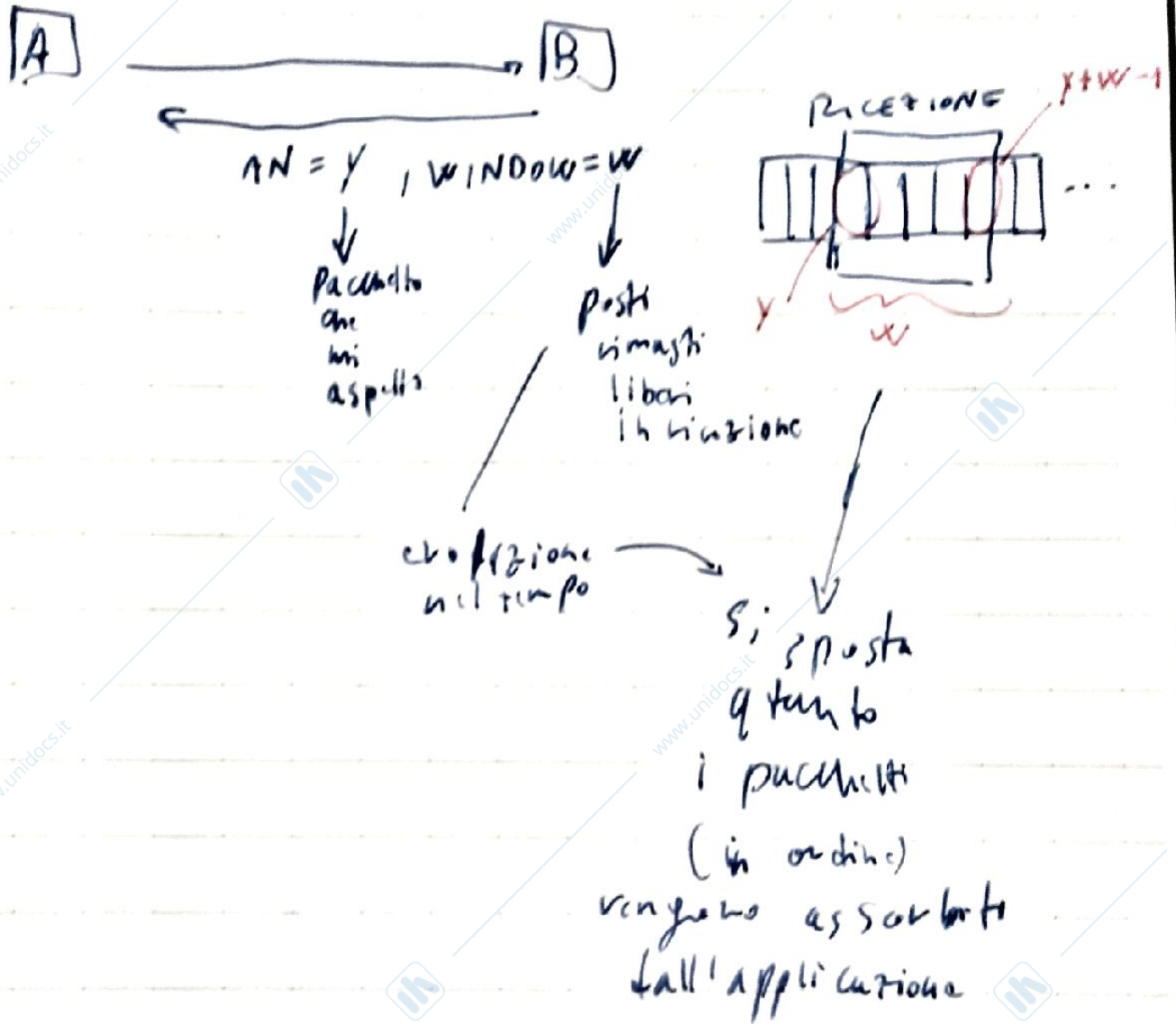


← finestra  
(da i byte 4 a 8  
posso mandare senza aspettare il broadcast)

A destinazione

stessa cosa per indicare quanti byte posso aspettare per ricevere anche non in sequenza

Se tubo va bene le 2 finestre scorrono contemporaneamente nel tempo. 1 tempo la linea di byte.



Non vediam  
Ungol e Clark

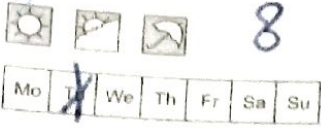
Casi di errori

Noi facciamo TCP-TAHoe (GBN)  
 TCP-reno (SR)

↙  
 l'unica differenza  
 rispetto a GBN e SR  
 di livello 2

è che TCP Non usa il  
 NACK partiva dopo che si era  
 invece ← ricevuto il pacchetto n+1, ma non  
 con mandato l'n, riscontrando implicitamente  
 l'ACK(n+1) fino a n-1  
 mandato NACK(n)

A livello 4 non siamo se un unico  
 cammino (in rete = tanti percorsi) per cui se  
 non mi arriva un pacchetto non significa che si  
 sia perso, ma magari solamente arriva fuori  
 sequenza  
 ↳ ritrasmissione avvengono solo per  
 scadenza del timer oqt



Cioè quello che  
mi aspetto.

No. RETI DI TELE.

Date 17 12 10

ripeto continuamente l'AN, di h (quello che non  
ho ricevuto) fino a quando non scade il  
time out e ritrasmette.

↓  
ci si mette troppo tempo

→ quindi si fa che dopo aver  
ricevuto (oltre al 1°) per 3  
volte AN si ritrasmette

↓  
3 AN = h vale come NACK CN

FAST RETRANSMIT

Mo	Tu	We	Th	Fr	Sa	Su
	X					

es (TCP ST Bcup) / Selective Repeat  
 (ASSUMIAMO SR)

MSS = 300 Byte

Time out = 8T<sub>e</sub> (tempi di trasmissione)  
 4 quadratini

W<sub>B</sub> = 1200

# SEGMENTI TRASMISSIONE SENZA RISCONTRO =  $\frac{W_B}{MSS} = 4$

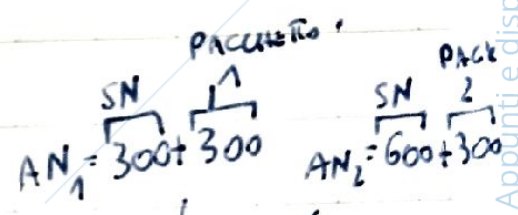
SN<sub>A</sub> = 300

Sequence number

4 posti in finestra di trasmissione

Disegno su file

Perdo il 3° pacchetto → non parte l'AN



dopo aver ricevuto l'ACK di 1 e 2  
 (notare che il riscontro arriva prima che finisca la finestra)

la finestra si sposta di 2  
 la finestra non sblocca

**BASIC TCP**

aspetto 8T<sub>e</sub> (8 quadratini) e ritrasmetto

 10

Mo	Tu	We	Th	Fr	Sa	Su
----	----	----	----	----	----	----

No. RETI DI TELE.Date 17.12.19

Oltre al 900 (3° pacchetto) vanno in timeout anche il 1200, anche 1500 e 1800 perché non è ancora arrivato (acknowledg. cumulativo = 2100)

Ricordiamo che l'AN da mandare dopo 1800 e 2100 (ossia il pacchetto che mi uspieto dopo 1800  $\rightarrow 1800+300$ )

### TCP con FINI RETRANSMIT

dopo 3 AN = 900 (in attesa del 3°)

ritrasmetto — la finestra si è bloccata a 1800

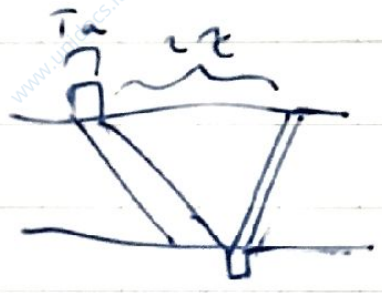
ritrasmetto anche 1200 poiché non mi è ancora arrivato l'AN cumulativo = 2100) e mi va scade il timeout di 1200

Abbiamo visto che non usiamo NACA  
 poiché RT (Method Trip Time) non è fisso  
 come a livello 2 ma è molto variabile.

a livello  
 4 possiamo  
 solo  
 fare una  
 stima  
 del RT

quindi  
 quella che  
 sembra  
 una perdita  
 ma qui  
 è solo la  
 attività fuori  
 sequenza

A livello 2 inoltre a vicenda visto che  
 vale  
 $T_0 > 2 \tau + T_a$



→ Il Time Out  $T_0$  dipende dalla stima  
 di RT

Stima di RTT

Ugiano yha STIMA MOBILE

es.

Abbiamo

$$RTT_1 = 5s$$

$$RTT_2 = 10s$$

$$RTT_3 = 6s$$

} dati ottenuti da 3 pacchetti che ho mandato



$$RTT = 7$$

Poss dobbiamo tenere conto del carico

(Carico + o meno busy)

↓  
 pian piano arriva + carico

↳ l'ultimo RTT lo faccio pesare di più

$$RTT_{av} = (1-\alpha)RTT_{av} + \alpha RTT_{last} \quad \begin{matrix} 0 \leq \alpha \leq 1 \\ \text{Casi particolari} \end{matrix}$$

C'è anche  $RTT_{dev} \rightarrow$  variabilità di RTT

$$RTT_{dev} = (1-\beta)RTT_{dev} + \beta (RTT_{last} - RTT_{av})$$

No. 1271 DI TELE.Date 17.12.19

ES 3 campioni slide  
 $\alpha = 0,4$   $\beta = 0,4$   $h = 2$

$$RT_{11} = 5, \quad RT_{12} = 10, \quad RT_{13} = 6$$

STEP	$RT_{av}$	$RT_{dev}$	$T_0$
0	$RT_{av}(0) = 0$	$RT_{dev}(0) = 1,5$	$T_0(0) = 3$ ← initializzato
1	$RT_{av}(1) = RT_{11} = RT_{12}$	$= \frac{RT_{11} + RT_{12}}{2} = 2,5$	$T_0 = 5 + 2 \cdot 1,5 = 10$
2	$RT_{av}(2) = 7 = (1 - 0,4) \cdot 5 + 0,4 \cdot 10$	$RT_{dev}(2) = 2,7$	$T_0 = 12,4$
3	$RT_{av}(3) = (1 - 0,4) \cdot 7 + 0,4 \cdot 6$	$RT_{dev}(3) = 1,38$	$T_0 = 9,36$

SALTIAMO IL CONTROLLO DI CONGESTIONE