

Consegna 3

A_output(message): Il messaggio, prima di essere inviato, viene creato e inserito in lista. Quando viene eseguito il controllo sulla finestra e il buffer risulta occupato il messaggio verrà inviato(o meno). Una volta inviato il messaggio, si metterà il campo `attend_ack` su 1 e il blocco in lista aggiornato.

A_input(pkt):

Se il pacchetto è corrotto non fa nulla e attende il nuovo pacchetto, altrimenti A controlla di che pacchetto si tratta, potrebbe essere un ACK cumulativo oppure uno dei due ack che B gli sta mandando in seguito alla ricezione di un pacchetto corrotto. I due casi sono distinti da un controllo incentrato sui valori `acknum` e `seqnum` del pacchetto ricevuto da A:

Se nel pacchetto `seqnum` è maggiore di `acknum` ci si trova nel caso di un doppio ack, per sapere se è il primo oppure il secondo si esegue un ulteriore controllo: se nel pacchetto `acknum` e `seqnum` corrispondono all'ultimo `acknum` e `seqnum` ricevuti allora siamo nel caso in cui stiamo ricevendo il secondo ack del secondo ack che B sta inviando, in questo caso A rispedisce il pacchetto che nella sua lista ha gli stessi valori descritti nel secondo ack; Se non corrispondono allora aggiorniamo gli ultimi `acknum` e `seqnum` ricevuti così da avere i valori aggiornati per il prossimo pacchetto che arriva; Nel caso opposto si tratta dell'Ack cumulativo, questo significa che B ha ricevuto tutta la finestra inviata e ci sta comunicando di aver ricevuto tutti i messaggi correttamente, dunque A si occupa di aggiornare la sua lista principale e quindi elimina tutti i pacchetti che fanno parte della finestra inviata a B, ovvero quelli con il capo attesaAck uguale a 1;

A_init(): si inizializza la Lista dei messaggi e i vari campi

A_timerinterrupt(): Quando scatta, A reinvia una finestra di messaggi

B_init(): Qui vengono settati i campi ai valori iniziali,

B_input(pkt): una volta ricevuto un pacchetto, il primo controllo che fa B è sulla correttezza del pacchetto:

se il pacchetto è corrotto si inviano 2 pacchetti ack con `seqnum` e `acknum` corrispondenti ai dati del pacchetto corrotto ricevuto. Pacchetto non corrotto si controlla se il `seqnum` è quello atteso:

in caso positivo si manda il pacchetto al livello successivo con `tolayer5(msg)` e lo si elimina dal buffer; nel caso negativo invece si inserisce il pacchetto nel buffer e si ricerca se è arrivato qualcosa in ordine da inviare al piano di sopra. Se qualcosa c'è poi lo elimina dal buffer;

Dopo questo controllo se ha ricevuto e trasmette correttamente tot messaggi da riempire la finestra allora manda l'ack cumulativo.

