Wrap-up

Informatica di base – a.a. 2022/2023

Silvio Peroni
0000-0003-0530-4305

Dipartimento di Filologia Classica e Italianistica, Università di Bologna, Bologna, Italia
silvio.peroni@unibo.it – @essepuntato – https://www.unibo.it/sitoweb/silvio.peroni/

DIPARTIMENTO DI FILOLOGIA CLASSICA E ITALIANISTICA

Aggiornare pagine Web (es. Wikipedia)

Metodi HTTP

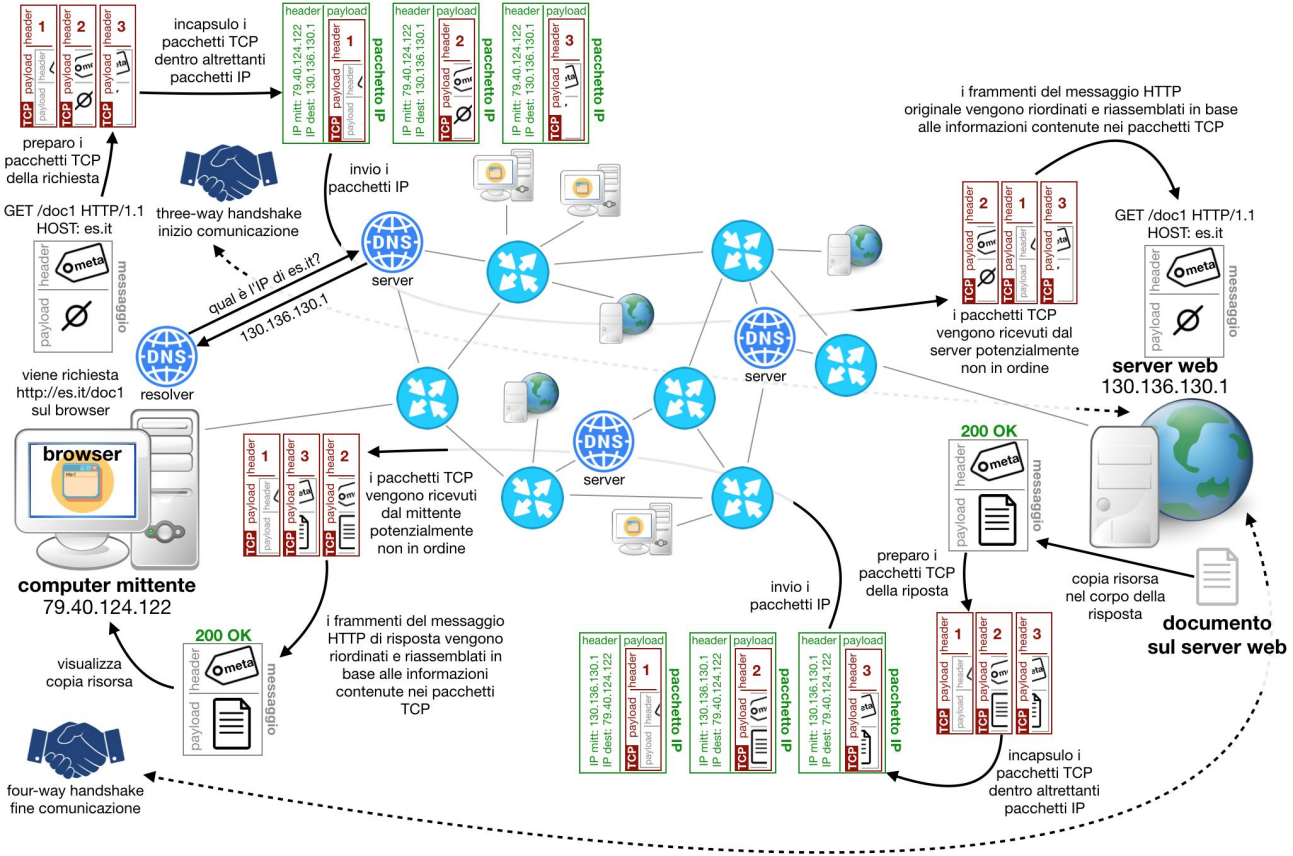
Il metodo più usato nel Web è **GET**, che permette di **richiedere informazioni** sulla risorsa definita dall'URL specificato – per esempio come conseguenza di un click su un collegamento ipertestuale

PUT permette di **creare** sul server web contattato la risorsa specificata dall'URL utilizzato nella richiesta, e di associarvi informazioni incluse nella richiesta

DELETE permette di **rimuovere** dal server web contattato tutte le informazioni relative alla risorsa specificata nella richiesta tramite l'URL

POST permette di **specificare informazioni aggiuntive**, incluse nella richiesta, ad una risorsa esistente che già risiede sul server

Richiesta - Risposta HTTP



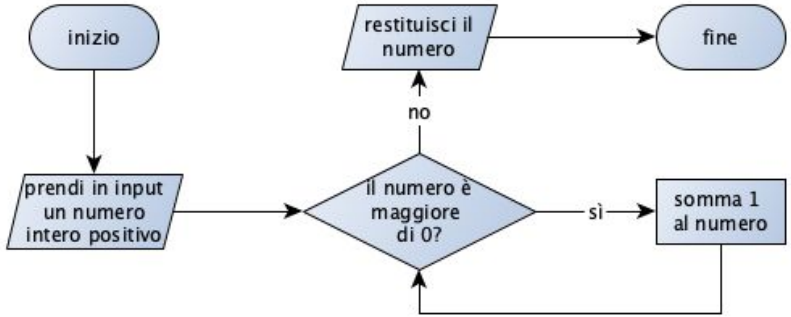
Algoritmi che non terminano e Turing

I problemi aperti della matematica

23 problemi aperti della matematica proposti da David Hilbert 1900

Includono (indirettamente) il **problema della terminazione**: capire se fosse possibile sviluppare un algoritmo che fosse in grado di rispondere se **un altro** algoritmo, specificato come input, terminasse la sua esecuzione o no

È possibile sviluppare un algoritmo che non termina mai la sua esecuzione?



Approssimazione della soluzione di Turing

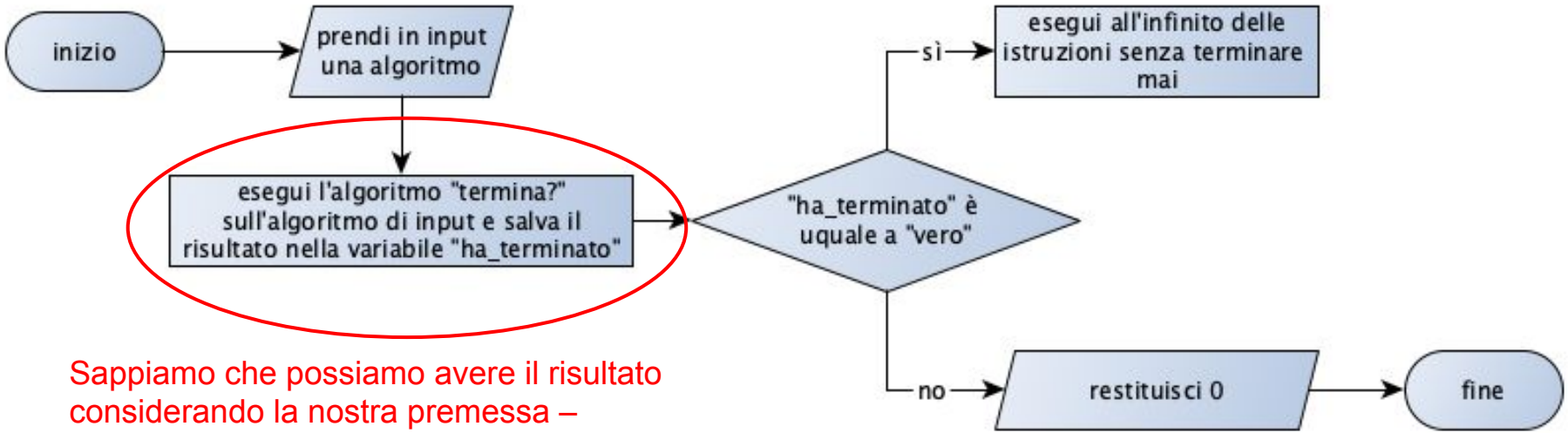
Premessa: supponiamo **sia possibile sviluppare** l'algoritmo "termina?", che prende in **input un certo algoritmo** e restituisce **"vero"** nel caso in cui l'algoritmo specificato come **input termina**, mentre restituisce **"falso" in caso contrario**

NB: è soltanto un algoritmo ipotetico, stiamo supponendo che possiamo svilupparlo in qualche modo, senza mostrare come farlo davvero

Usiamo questo algoritmo per crearne un altro

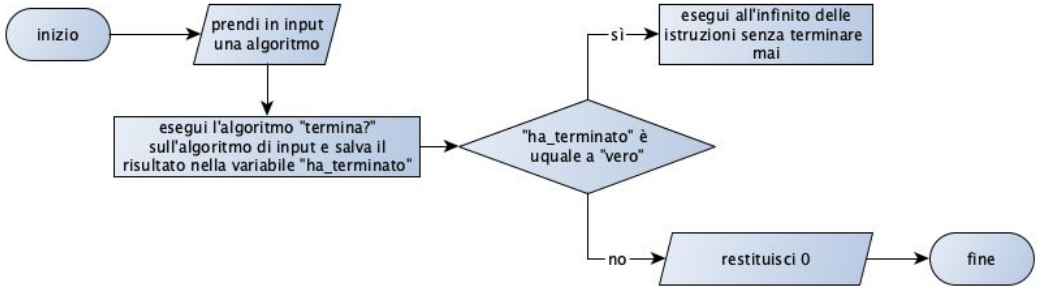
Nuovo algoritmo

Nuovo algoritmo: prendi in input un algoritmo e restituisci 0 se l'algoritmo in input termina, mentre non terminare in caso contrario



Sappiamo che possiamo avere il risultato considerando la nostra premessa – ovvero che "ha_terminato" esiste

E se usiamo il nuovo algoritmo come suo input?



Caso 1: se l'algoritmo "termina?" afferma che il nuovo algoritmo **termina**, conseguentemente (per come è definito) il nuovo algoritmo **non termina** l'esecuzione

Caso 2: se l'algoritmo "termina?" afferma che il nuovo algoritmo **non termina**, e conseguentemente (per come è definito) il nuovo algoritmo restituisce 0 e **termina** l'esecuzione

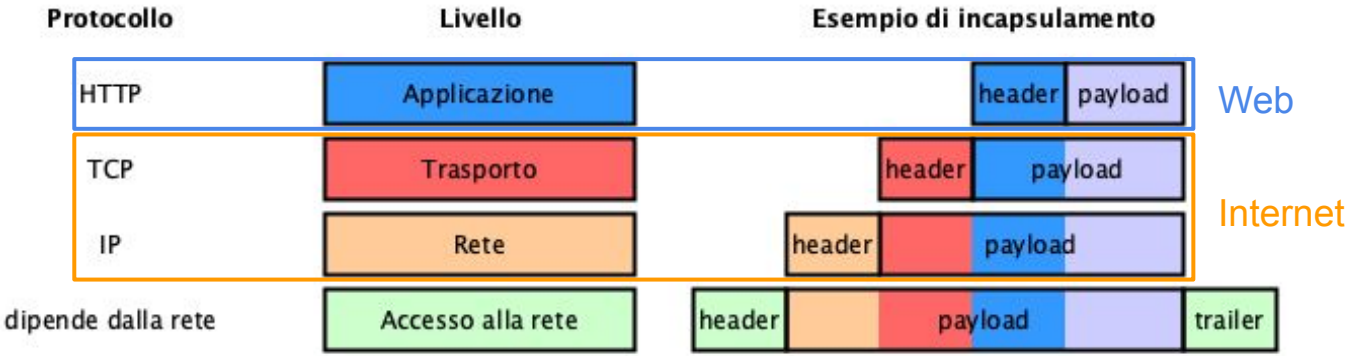
Paradosso: l'algoritmo che verifica se un altro termina **non può esistere**

Gerarchia protocolli, pacchetti e TCP

Gerarchia protocolli

Lo scambio di dati tra due computer collegati in rete è realizzata mediante l'uso dei pacchetti

Esiste una gerarchia di incapsulamento dei dati da spedire, definita dalla suite di protocolli Internet TCP/IP, organizzata in quattro livelli



Transmission Control Protocol (TCP)

Sia il riordino dei pacchetti sia lo spezzare il messaggio originale in più pacchetti conformemente con l'MTU della rete, è gestito dallo strato di trasporto

Protocollo principale: **Transmission Control Protocol (TCP)** proposto da Vinton Cerf e Robert Kahn nel 1974 per ARPANET

Cosa permette di fare

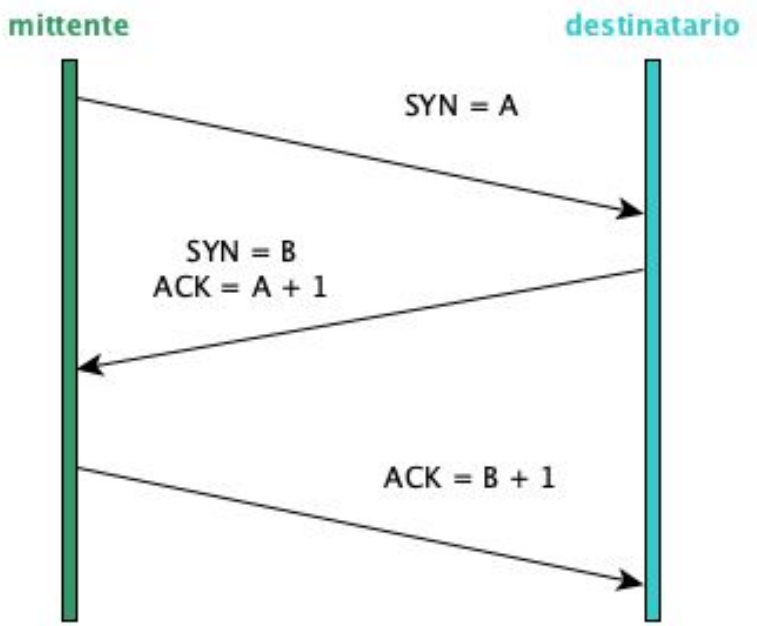
Il TCP permette la consegna **affidabile**, **ordinata**, e **esente da errori** di un flusso di byte tra due computer in comunicazione tra loro attraverso una rete basata sul protocollo IP, e gestisce tutti quei processi che garantiscono la **ritrasmissione** di un pacchetto nel caso in cui non sia stato recapitato al destinatario entro un certo tempo limite

L'**header** di un pacchetto TCP contiene informazioni relative alla comunicazione a livello trasporto tra i due computer mittente e destinatario (le **porte** usate per la comunicazione e un **numero di sequenza** che indica l'ordine dei vari pacchetti TCP), mentre il **payload** contiene le informazioni che devono essere scambiate tra i partecipanti alla comunicazione

Comunicazione connessa

Il mittente e il destinatario **si mettono d'accordo** di iniziare una comunicazione in modo esplicito prima di scambiarsi i dati, e dichiarano altrettanto esplicitamente quando questa comunicazione si può ritenere conclusa

Il processo di inizio della comunicazione, è regolato dal meccanismo del **three-way handshake**, mentre quello di chiusura è il **four-way handshake**

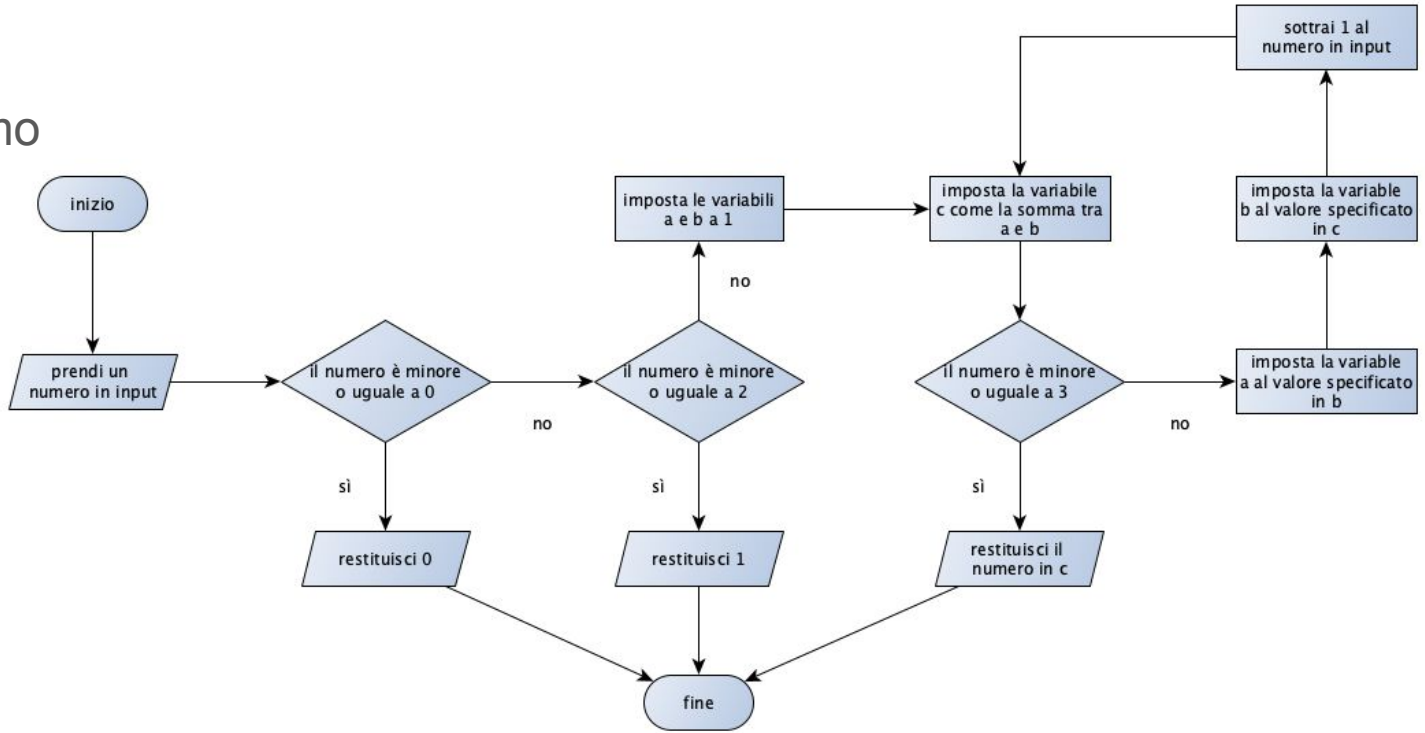


Esempio diagramma di flusso

Esempio 1

Cosa viene restituito dal seguente algoritmo se lo si esegue specificando il numero “4” come input?

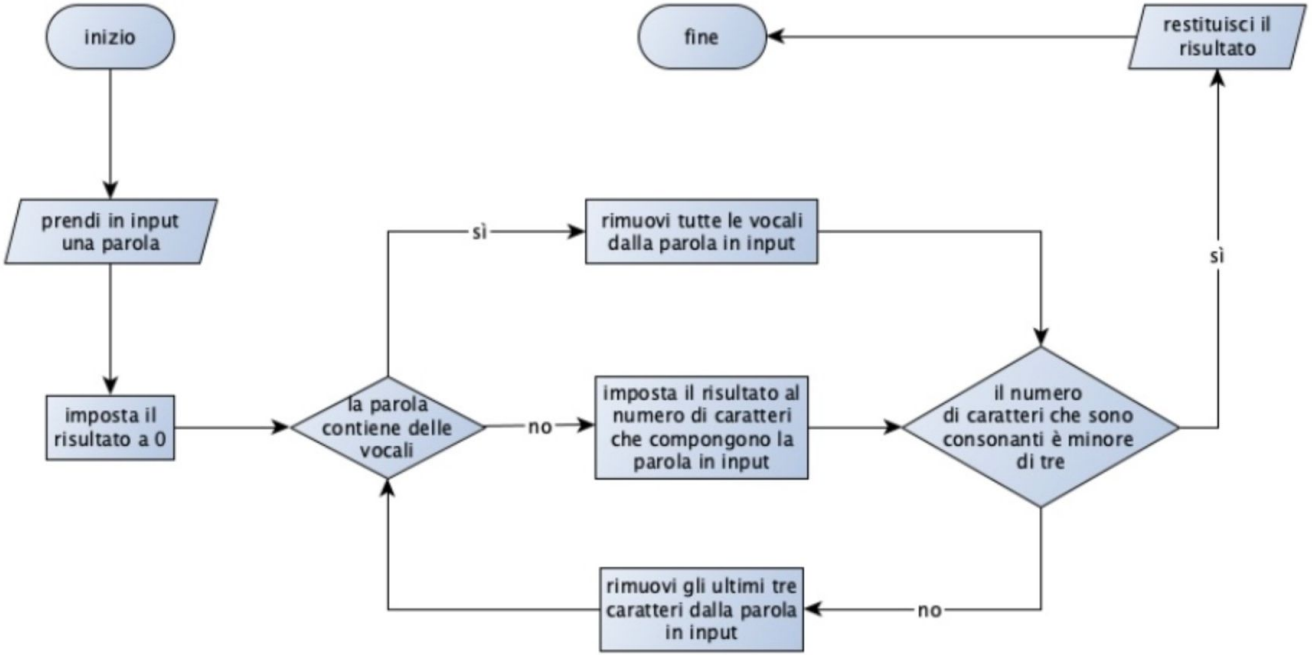
- ❏ 1
- ❏ 10
- ❏ 3
- ❏ 5



Esempio 2

Qual è il risultato che si ottiene eseguendo l'algoritmo descritto nel diagramma di flusso se la parola in input è "panda"

- ❏ 4
- ❏ 2
- ❏ 1
- ❏ 0



Wrap-up

Le tecnologie informatiche nelle scienze umane

Informatica di base – a.a. 2022/2023

Silvio Peroni
0000-0003-0530-4305

Dipartimento di Filologia Classica e Italianistica, Università di Bologna, Bologna, Italia
silvio.peroni@unibo.it – @essepuntato – https://www.unibo.it/sitoweb/silvio.peroni/

DIPARTIMENTO DI FILOLOGIA CLASSICA E ITALIANISTICA