

# Calcolabilità e Complessità

Possiamo formulare domande :  $a \neq b$

chiedo se  $a=b$  :  $w \neq w$  se viene accettato allora la risposta è sì!

Esempio di domande a cui

non posso rispondere

(Non precisa nel 100%)

$J \neq S$

Programma Java  
ben terminato.

Si termina su S? Il nostro programma  
terminerà oppure no?

Definizione di Turing-Recognizable Forma debole

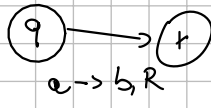
Decisione Positiva una macchina che dice Sì se appartiene al linguaggio oppure no e se  
"Loop"

Decisore Più affidabile : Sì o No non va in loop

Decisore Positivo

$L_{conv} = \{ J \neq S \mid \text{prog. Java } S = \text{Std\_Input}, J \text{ termina su } S \}$

$J \neq S \in L_{conv}?$



Una macchina del genere sarebbe  
Lenta, difficile da progettare e non  
avrebbe lo stato : NO!

Dato un programma e un dato

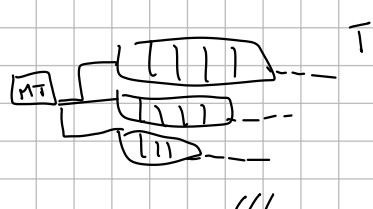
All'inizio dell'informatica ci si chiedeva se esistesse un altro decisore per  $L_{conv}$  un meta-decisore

Varianti delle MT

Multitasta oppure un'istruzione aggiuntiva che permette di fermarsi invece di andare a destra  
o sinistra

Multitasta con k numero di tape  $\delta : Q \times \Gamma^k \rightarrow Q \times \Gamma^k \times \{L, R, S\}^k$

$\delta = (q_i, a_1, \dots, a_k)$



Ogni macchina MULTITape può fare gli stessi calcoli  
di una semplice MT possiamo concatenare il contenuto  
dei 3 nastri con dei divisoni (#) un passaggio di  
MT costa quanto tutta il passaggio su **S** 2 volte  
(Lettura + Scrittura)  
della memoria

Teorema

Un linguaggio è riconoscibile da una MT con 1 tape  $\Leftrightarrow$  è riconoscibile da MT con k-tape

Una 'decisione' è una relazione fra una MT ed un linguaggio

M chiede  $L \subseteq \Sigma^*$   $\forall w \in \Sigma$  se  $w \in L$  accetta w  
se  $w \notin L$  non accetta