

# Complessità

## Calcolabilità

21/11

Ora ci interrogheremo sull'efficienza degli algoritmi ( $T(n)$ ) nella risoluzione di problemi (decisione di linguaggi).

L'efficienza viene misurata in termini di risorse:

- tempo e spazio (memoria)
- randomness
- numero processori per calcolo parallelo

Dunque le risposte alle domande quanto spazio e quanto tempo trovano risposta all'interno della **teoria della complessità**.

Vengono inoltre analizzati problemi del tipo:

- $P=NP \rightarrow$  trovare una soluzione è tanto veloce quanto verificarne una?
- $P=PSPACE \rightarrow$  se puoi risolvere un problema in poco spazio, puoi risolverlo in poco tempo?
- $P=BPP \rightarrow$  ogni algoritmo efficiente può essere derandomizzato

Iniziamo con il definire la complessità di tempo (caso deterministico)

### DEF Complessità di tempo

Sia  $M \in \mathcal{M}$  decisione, la **complessità di tempo** di  $M$  è  $T: \mathbb{N} \rightarrow \mathbb{N}$  t.c.  $T(n) = \max_{\substack{x \in \Sigma^* \\ |x|=n}} \{\# \text{passi richiesti da } M(x)\}$

**OSS** siamo interessati a come questa funzione si comporta in termini della lunghezza dell'input