# Limiti di un Elaboratore

## Lemma

*There are problems that cannot be solved by any model of computation. Such problems are called* ***undecidable*** *(indecidibili).*

## Example

- **Input**:
  1. Dato un (generico) programma *Prog*
  2. Dati dei valori **val** (potenziali input per *Prog*)
- **Output**: Sì oppure No
- **Vincoli**:
  1. *Prog* deve *terminare* (in tempo finito) sull'input **val**

# Alfabeti e Stringhe

Chiameremo **alfabeto** un insieme *finito* e *non vuoto* di simboli:

- $\Sigma_1 = \{\texttt{a}, \texttt{b}, \texttt{c}\}$ è un alfabeto
- $\Sigma_2 = \{\texttt{0}, \texttt{1}\}$ è un alfabeto
- $\Sigma_3 = \{1, 2, 3, \ldots\}$ non è un alfabeto (poiché infinito)

Chiameremo **stringa** su un alfabeto $\Sigma$ ogni *sequenza finita* di simboli presi da $\Sigma$:

- `aaa`, `abbbabc`, `ccab` sono stringhe su $\Sigma_1$
- `001` è una stringa su $\Sigma_2$ ma non su $\Sigma_1$
- `aacd` non è una stringa né su $\Sigma_1$, né su $\Sigma_2$

# La stringa vuota

Chiameremo **stringa vuota** la speciale stringa $\varepsilon$ che gode delle seguenti proprietà:

1. la sua lunghezza è nulla (uguale a zero)
2. presa una (qualunque) stringa $w$ su un (qualunque) alfabeto $\Sigma$, allora $w\varepsilon = \varepsilon w = w$

## Example

- $abbc\varepsilon = \varepsilon abbc = abbc$
- $0100\varepsilon = \varepsilon 0100 = 0100$
- $aa\varepsilon bb = aa\varepsilon bb = aabb$

# Linguaggio Universale

Chiameremo **linguaggio universale** su un dato alfabeto $\Sigma$ l'unione della *stringa vuota* con *tutte le stringhe* (di lunghezza finita) che si possono costruire partendo da $\Sigma$.

## Example

- $\Sigma_1^* = \{\varepsilon, a, b, c, aa, ab, ac, ba, bb, bc, ca, cb, cc, aaa, \ldots\}$
- $\Sigma_2^* = \{\varepsilon, 0, 1, 00, 01, 10, 11, 000, 001, 010, 011, 100, 101, \ldots\}$

## Lemma

*La concatenazione di due stringhe da $\Sigma^*$ è ancora una stringa in $\Sigma^*$*

## Example

- concatenando *aaa* e *bbb* otteniamo *aaabbb* e *bbbaaa* (tutte in $\Sigma_1^*$)
- concatenando 011 e 10 otteniamo 01110 e 10011 (tutte in $\Sigma_2^*$)

# Linguaggi

**Definizione**

Chiameremo **linguaggio** su un alfabeto $\Sigma$ qualsiasi insieme di stringhe prese da $\Sigma^*$.

Esempi di linguaggi:

- Sia $\Sigma = \{a, b, c, \ldots, z\}$ un alfabeto. Allora, $Stagioni = \{primavera, estate, autunno, inverno\}$ è un linguaggio (finito) su $\Sigma$
- Sia $\Sigma = \{1, 2, 3, \ldots, 9, 0\}$ un alfabeto. Allora, $Primi = \{2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, \ldots\}$ è un linguaggio (infinito) su $\Sigma$

# Il problema dell'**Appartenenza**

**Definizione**

Tutti i problemi **decisionali** dell'informatica possono essere ricondotti al problema **dell'appartenenza di una stringa ad un linguaggio**.

**Input**:

1. Dato un alfabeto $\Sigma$
2. Fissato un linguaggio $L$ di stringhe appartenenti a $\Sigma^*$
3. Data una stringa $w$ sull'alfabeto $\Sigma$

**Output**: Sì oppure No

**Vincoli**:

1. la stringa $w$ deve appartenere al linguaggio $L$

# Come possono essere *"descritti"* i linguaggi?

- Per **enumerazione** delle stringhe. Es. $\{aa, aba, bb, bab\}$
- Con **notazione matematica**. Es.
  $\{a^n b^n : n > 0\} = \{ab, aabb, aaabbb, aaaabbbb, \ldots\}$
- Tramite **Grammatiche Formali**
- ...

**Definizione**

Una **Grammatica** è una *notazione formale* con cui esprimere in modo rigoroso la **SINTASSI di un linguaggio**.

## Esempio (1)

- Dato l'alfabeto $\Sigma = \{\texttt{a}, \texttt{b}, \ldots, \texttt{z}\}$
- Dato un insieme di elementi aggiuntivi chiamati non-terminali $NT = \{\mathbb{FRASE}, \mathbb{SOGGETTO}, \mathbb{PREDICATO}, \mathbb{OGGETTO}\}$

**Example (Grammatica $\mathcal{G}$)**

$\mathbb{FRASE} \rightarrow \mathbb{SOGGETTO}\ \ \mathbb{PREDICATO}\ \ \mathbb{OGGETTO}$
$\mathbb{SOGGETTO} \rightarrow$ `marco` | `lucia`
$\mathbb{PREDICATO} \rightarrow$ `mangia` | `pulisce`
$\mathbb{OGGETTO} \rightarrow$ `la mela` | `il banco`

La grammatica $\mathcal{G}$ definisce il seguente linguaggio: $\mathcal{L}(\mathcal{G}) = \{$ `marco mangia la mela`, `marco mangia il banco`, `marco pulisce la mela`, `marco pulisce il banco`, `lucia mangia la mela`, `lucia mangia il banco`, `lucia pulisce la mela`, `lucia pulisce il banco`$\}$

# Esempio (2)

## Example (Grammatica $\mathcal{G}_{ab}$)

$\mathbb{P} \rightarrow \mathbb{A}\ \mathbb{Q}$

$\mathbb{Q} \rightarrow \mathbb{B} \mid \mathbb{P}\ \mathbb{B}$

$\mathbb{A} \rightarrow \mathtt{a}$

$\mathbb{B} \rightarrow \mathtt{b}$

Qual'è il linguaggio descritto (o definito) dalla grammatica $\mathcal{G}_{ab}$?

- $\mathbb{P} \rightarrow \mathbb{A}\ \mathbb{Q} \rightarrow \mathbb{A}\ \mathbb{B} \rightarrow \mathtt{a}\ \mathtt{b}$
- $\mathbb{P} \rightarrow \mathbb{A}\ \mathbb{Q} \rightarrow \mathbb{A}\ \mathbb{P}\ \mathbb{B} \rightarrow \mathbb{A}\ \mathbb{A}\ \mathbb{Q}\ \mathbb{B} \rightarrow \mathbb{A}\ \mathbb{A}\ \mathbb{B}\ \mathbb{B} \rightarrow \mathtt{a}\ \mathtt{a}\ \mathtt{b}\ \mathtt{b}$
- $\mathbb{P} \rightarrow \ldots \rightarrow \mathtt{a}\ \mathtt{a}\ \mathtt{a}\ \mathtt{b}\ \mathtt{b}\ \mathtt{b}$
- $\ldots$
- $\mathbb{P} \rightarrow \ldots \rightarrow \underbrace{\mathtt{a}\,\mathtt{a}\ldots\mathtt{a}}_{n}\ \underbrace{\mathtt{b}\,\mathtt{b}\ldots\mathtt{b}}_{n}$

# Esempio (2)

## Example (Grammatica $\mathcal{G}_{ab}$)

$\mathbb{P} \rightarrow \mathbb{A}\ \mathbb{Q}$
$\mathbb{Q} \rightarrow \mathbb{B} \mid \mathbb{P}\ \mathbb{B}$
$\mathbb{A} \rightarrow \mathtt{a}$
$\mathbb{B} \rightarrow \mathtt{b}$

Qual'è il linguaggio descritto (o definito) dalla grammatica $\mathcal{G}_{ab}$?

- $\mathbb{P} \rightarrow \mathbb{A}\ \mathbb{Q} \rightarrow \mathbb{A}\ \mathbb{B} \rightarrow \mathtt{a}\ \mathtt{b}$
- $\mathbb{P} \rightarrow \mathbb{A}\ \mathbb{Q} \rightarrow \mathbb{A}\ \mathbb{P}\ \mathbb{B} \rightarrow \mathbb{A}\ \mathbb{A}\ \mathbb{Q}\ \mathbb{B} \rightarrow \mathbb{A}\ \mathbb{A}\ \mathbb{B}\ \mathbb{B} \rightarrow \mathtt{a}\ \mathtt{a}\ \mathtt{b}\ \mathtt{b}$
- $\mathbb{P} \rightarrow \ldots \rightarrow \mathtt{a}\ \mathtt{a}\ \mathtt{a}\ \mathtt{b}\ \mathtt{b}\ \mathtt{b}$
- $\ldots$
- $\mathbb{P} \rightarrow \ldots \rightarrow \underbrace{\mathtt{a}\ \mathtt{a} \ldots \mathtt{a}}_{n}\ \underbrace{\mathtt{b}\ \mathtt{b} \ldots \mathtt{b}}_{n}$

# Esempio (2)

## Example (Grammatica $\mathcal{G}_{ab}$)

$\mathbb{P} \rightarrow \mathbb{A}\ \mathbb{Q}$
$\mathbb{Q} \rightarrow \mathbb{B} \mid \mathbb{P}\ \mathbb{B}$
$\mathbb{A} \rightarrow \mathtt{a}$
$\mathbb{B} \rightarrow \mathtt{b}$

Qual'è il linguaggio descritto (o definito) dalla grammatica $\mathcal{G}_{ab}$?

- $\mathbb{P} \rightarrow \mathbb{A}\ \mathbb{Q} \rightarrow \mathbb{A}\ \mathbb{B} \rightarrow \mathtt{a}\ \mathtt{b}$
- $\mathbb{P} \rightarrow \mathbb{A}\ \mathbb{Q} \rightarrow \mathbb{A}\ \mathbb{P}\ \mathbb{B} \rightarrow \mathbb{A}\ \mathbb{A}\ \mathbb{Q}\ \mathbb{B} \rightarrow \mathbb{A}\ \mathbb{A}\ \mathbb{B}\ \mathbb{B} \rightarrow \mathtt{a}\ \mathtt{a}\ \mathtt{b}\ \mathtt{b}$
- $\mathbb{P} \rightarrow \ldots \rightarrow \mathtt{a}\ \mathtt{a}\ \mathtt{a}\ \mathtt{b}\ \mathtt{b}\ \mathtt{b}$
- $\ldots$
- $\mathbb{P} \rightarrow \ldots \rightarrow \underbrace{\mathtt{a}\ \mathtt{a} \ldots \mathtt{a}}_{n}\ \underbrace{\mathtt{b}\ \mathtt{b} \ldots \mathtt{b}}_{n}$

# Esempio (2)

## Example (Grammatica $\mathcal{G}_{ab}$)

$\mathbb{P} \rightarrow \mathbb{A}\ \mathbb{Q}$
$\mathbb{Q} \rightarrow \mathbb{B} \mid \mathbb{P}\ \mathbb{B}$
$\mathbb{A} \rightarrow \mathtt{a}$
$\mathbb{B} \rightarrow \mathtt{b}$

Qual'è il linguaggio descritto (o definito) dalla grammatica $\mathcal{G}_{ab}$?

- $\mathbb{P} \rightarrow \mathbb{A}\ \mathbb{Q} \rightarrow \mathbb{A}\ \mathbb{B} \rightarrow \mathtt{a}\ \mathtt{b}$
- $\mathbb{P} \rightarrow \mathbb{A}\ \mathbb{Q} \rightarrow \mathbb{A}\ \mathbb{P}\ \mathbb{B} \rightarrow \mathbb{A}\ \mathbb{A}\ \mathbb{Q}\ \mathbb{B} \rightarrow \mathbb{A}\ \mathbb{A}\ \mathbb{B}\ \mathbb{B} \rightarrow \mathtt{a}\ \mathtt{a}\ \mathtt{b}\ \mathtt{b}$
- $\mathbb{P} \rightarrow \ldots \rightarrow \mathtt{a}\ \mathtt{a}\ \mathtt{a}\ \mathtt{b}\ \mathtt{b}\ \mathtt{b}$
- ...
- $\mathbb{P} \rightarrow \ldots \rightarrow \underbrace{\mathtt{a}\,\mathtt{a}\ldots\mathtt{a}}_{n}\ \underbrace{\mathtt{b}\,\mathtt{b}\ldots\mathtt{b}}_{n}$

# Esempio (2)

## Example (Grammatica $\mathcal{G}_{ab}$)

$\mathbb{P} \rightarrow \mathbb{A}\ \mathbb{Q}$
$\mathbb{Q} \rightarrow \mathbb{B} \mid \mathbb{P}\ \mathbb{B}$
$\mathbb{A} \rightarrow \mathtt{a}$
$\mathbb{B} \rightarrow \mathtt{b}$

Qual'è il linguaggio descritto (o definito) dalla grammatica $\mathcal{G}_{ab}$?

- $\mathbb{P} \rightarrow \mathbb{A}\ \mathbb{Q} \rightarrow \mathbb{A}\ \mathbb{B} \rightarrow \mathtt{a}\ \mathtt{b}$
- $\mathbb{P} \rightarrow \mathbb{A}\ \mathbb{Q} \rightarrow \mathbb{A}\ \mathbb{P}\ \mathbb{B} \rightarrow \mathbb{A}\ \mathbb{A}\ \mathbb{Q}\ \mathbb{B} \rightarrow \mathbb{A}\ \mathbb{A}\ \mathbb{B}\ \mathbb{B} \rightarrow \mathtt{a}\ \mathtt{a}\ \mathtt{b}\ \mathtt{b}$
- $\mathbb{P} \rightarrow \ldots \rightarrow \mathtt{a}\ \mathtt{a}\ \mathtt{a}\ \mathtt{b}\ \mathtt{b}\ \mathtt{b}$
- $\ldots$
- $\mathbb{P} \rightarrow \ldots \rightarrow \underbrace{\mathtt{a}\ \mathtt{a} \ldots \mathtt{a}}_{n}\ \underbrace{\mathtt{b}\ \mathtt{b} \ldots \mathtt{b}}_{n}$

# Esempio (2)

## Example (Grammatica $\mathcal{G}_{ab}$)

$\mathbb{P} \rightarrow \mathbb{A}\ \mathbb{Q}$

$\mathbb{Q} \rightarrow \mathbb{B} \mid \mathbb{P}\ \mathbb{B}$

$\mathbb{A} \rightarrow \mathtt{a}$

$\mathbb{B} \rightarrow \mathtt{b}$

Qual'è il linguaggio descritto (o definito) dalla grammatica $\mathcal{G}_{ab}$?

- $\mathbb{P} \rightarrow \mathbb{A}\ \mathbb{Q} \rightarrow \mathbb{A}\ \mathbb{B} \rightarrow \mathtt{a}\ \mathtt{b}$
- $\mathbb{P} \rightarrow \mathbb{A}\ \mathbb{Q} \rightarrow \mathbb{A}\ \mathbb{P}\ \mathbb{B} \rightarrow \mathbb{A}\ \mathbb{A}\ \mathbb{Q}\ \mathbb{B} \rightarrow \mathbb{A}\ \mathbb{A}\ \mathbb{B}\ \mathbb{B} \rightarrow \mathtt{a}\ \mathtt{a}\ \mathtt{b}\ \mathtt{b}$
- $\mathbb{P} \rightarrow \ldots \rightarrow \mathtt{a}\ \mathtt{a}\ \mathtt{a}\ \mathtt{b}\ \mathtt{b}\ \mathtt{b}$
- $\ldots$
- $\mathbb{P} \rightarrow \ldots \rightarrow \underbrace{\mathtt{a}\,\mathtt{a}\ldots\mathtt{a}}_{n}\ \underbrace{\mathtt{b}\,\mathtt{b}\ldots\mathtt{b}}_{n}$

Grammatiche *context-free*
Chiameremo **context-free** quelle grammatiche dove tutte le regole sono del tipo

$$\mathbb{N} \rightarrow \alpha_1 \ldots \alpha_n$$

In particolare,

- $\mathbb{N}$ è un *non-terminale*
- ciascun $\alpha$ può essere un *non-terminale* oppure un *simbolo dell'alfabeto*

# Equivalenza di Grammatiche *context-free*

**Input**:

1. Data una Grammatica context-free $\mathcal{G}_1$
2. Data una Grammatica context-free $\mathcal{G}_2$

**Output**: Sì oppure No

**Vincoli**:

1. Il linguaggio "descritto" da $\mathcal{G}_1$ deve essere equivalente al linguaggio "descritto" da $\mathcal{G}_2$

**Lemma**

*Tale problema è indecidibile.*

# Appartenenza ad una Grammatica *context-free*

**Input**:

1. Data una Grammatica context-free $\mathcal{G}$
2. Data una stringa **w**

**Output**: Sì oppure No

**Vincoli**:

1. La stringa **w** deve appartenere al linguaggio descritto da $\mathcal{G}$

**Lemma**

*Tale problema è decidibile. . . ed è pure "semplice"!*

# Linguaggi di programmazione

- Un **linguaggio di programmazione** è un linguaggio formale la cui sintassi è (usualmente) descritta mediante una grammatica context-free
- Un **programma** in un dato linguaggio di programmazione è una stringa (sequenza finita di simboli su un certo alfabeto) che appartiene a quel dato linguaggio
- La **verifica di correttezza sintattica** di un programma è un problema decidibile e relativamente semplice (vedi slide precedente)
- Sapere se un **programma** va in loop su qualche input è un problema indecidibile (vedi prima slide di questa sezione)
- Un programma, pur essendo semplicemente una sequenza di simboli, "dice" al computer le azioni da eseguire così come un **manuale di istruzione** "dice" a noi come far funzionare una lavatrice!